Proceedings of the
Joint ARW & OAGM Workshop 2019

May 9–10, 2019
University of Applied Sciences
Upper Austria, Campus Steyr
Steyr, Austria

OAGM - Austrian Association of Pattern Recognition
GMAR - Gesellschaft für Mess-, Automatisierungs- und Robotertechnik

TU Graz
Graz University of Technology
OAGM
OESTERREICHISCHE COMPUTER GESELLSCHAFT
AUSTRIAN COMPUTER SOCIETY
GMAR
Die österreichische Gesellschaft für Mess-, Automatisierungs- und Robotertechnik

Andreas Pichler, Peter M. Roth, Robert Sablatnig, Gernot Stübl,
and Markus Vincze (eds.)

Proceedings of the

Joint ARW & OAGM Workshop 2019

May 9–10, 2019

University of Applied Sciences Upper Austria, Campus Steyr

Steyr, Austria

Austrian Association of Pattern Recognition (OAGM)

Gesellschaft für Mess-, Automatisierungs- und Robotertechnik (GMAR)

Editors

Andreas Pichler, Peter M. Roth, Robert Sablatnig, Gernot Stübl, and Markus Vincze

Layout

Austrian Association of Pattern Recognition
http://aapr.at/

PROFACTOR GmbH
www.profactor.at

Supported by

Cover

Verlag der Technischen Universität Graz

ISBN 978-3-85125-663-5
DOI 10.3217/978-3-85125-663-5

Contents

Preface

The ARW and OAGM Workshop on "Vision and Robotics" is organized this year by Profactor in the beautiful city of Steyr. The main intention is to bring together researchers, students, professionals, and practitioners from the fields of Computer Vision and Robotics to present and actively discuss latest research and developments. While in the past there has been a perceivable gap between these two research directions, one may note that the borders get more and more blurred. During the preparation of the workshop this observation was underpinned by interesting contributions addressing both scientific communities. From over 50 submissions, an international program committee selected 48 for the presentation at the workshop. Based on the decisions, we could assemble an interesting and lively program with 30 talks and 18 posters both in plenum and parallel sessions. During the joint workshop, outstanding contributions will be awarded with prices sponsored by OAGM and IEEE RAS. We are also very proud that four invited international established researchers support the event: Jiří Matas (Czech Technical University of Prague), Manfred Tscheligi (Austrian Institute of Technology & University of Alin Albu-Schäffer (German Aerospace Center).

We wish an interesting and fruitful event,

Andreas Pichler, Markus Vincze (ARW Chairs)

Gernot Stübl, Robert Sablatnig, Peter M. Roth (OAGM Chairs)

Steyr, May 2019

ARW Workshop Chairs

Markus Vincze (Vienna University of Technology)

Andreas Pichler (PROFACTOR GmbH)

OAGM Workshop Chairs

Robert Sablatnig (Vienna University of Technology)

Peter M. Roth (Graz University of Technology)

Gernot Stübl (PROFACTOR GmbH)

ARW Program Committee

Markus Bader (Vienna University of Technology)
Mathias Brandstötter (JOANNEUM RESEARCH)
Bernhard Dieber (JOANNEUM RESEARCH)
Alexander Hofmann (University of Applied Sciences Technikum Wien)
Martin Humenberger (Naver Labs Europe)
Brigitte Krenn (The Austrian Research Institute for Artificial Intelligence)
Gernot Kronreif (Austrian Center for Medical Innovation and Technology)
Wilfried Kubinger (University of Applied Sciences Technikum Wien)
Wilfried Lepuschitz (Practical Robotics Institute Austria)
Andreas Müller (Johannes Kepler University Linz)
Andreas Pichler (PROFACTOR GmbH)
Friedrich Praus (University of Applied Sciences Technikum Wien)
Bernhard Rinner (Klagenfurt University)
Lukas Silberbauer (taurob OG)
Gerald Steinbauer (Graz University of Technology)
Markus Vincze (Vienna University of Technology)
Christian Wögerer (PROFACTOR GmbH)

OAGM Program Committee

Helmut Ahammer(Medical University of Graz)
Csaba Beleznai (Austrian Institut of Technology)
Horst Bischof (Graz University of Technology)
Michael Bleyer (Microsoft)
Kristian Bredies (University of Graz)
Katja Buehler (VRVis)
Wilhelm Burger (Upper Austria University of Applied Sciences)
Friedrich Fraundorfer (Graz University of Technology)
Johannes Fuernkranz (TU Darmstadt)
Harald Ganster (JOANNEUM RESEARCH)
Margrit Gelautz (Vienna University of Technology)
Alexander Grabner (Graz University of Technology) Martin Hirzer (Graz University of Technology)
Michael Hoedlmoser (emotion3D) Martin Kampel (Vienna University of Technology)
Florian Kleber (Vienna University of Technology)
Arjan Kuijper (TU Darmstadt)
Christoph Lampert (IST Austria)
Mathias Lux (Klagenfurt University)
Hubert Mara (IWR Heidelberg University)
Gerhard Paar (JOANNEUM RESEARCH)
Justus Piater (University of Innsbruck)
Peter M. Roth (Graz University of Technology)
Robert Sablatnig (Vienna University of Technology)
Josef Scharinger (Johannes Kepler University)
Otmar Scherzer (University of Vienna)
Gernot Stübl (PROFACTOR GmbH)
Stefan Thumfart (RISC Software GmbH, Research Unit Medical Informatics)
Andreas Uhl (University of Salzburg)
Martin Welk (UMIT – Private University of Health Sciences, Medical Informatics and Technology)
Sebastian Zambal (PROFACTOR GmbH)

OAGM Award 2018

The

OCG Best Paper Award 2018

was awarded to the paper

Globally Consistent Dense Real-Time 3D Reconstruction from RGBD Data

by

Rafael Weilharter, Fabian Schenk, and Friedrich Fraundorfer.

ARW Awards 2018

The

IEEE RAS Austria Best Research Paper Award

was awarded to the paper

Analysis of Feature Tracking Methods for Vision-Based Vibration Damping of Flexible Link Robots

by

Florian Pucher, Hubert Gattringer, and Andreas Müller.

The

Best Student Paper

sponsored by the ABB-Group was awarded to the paper

Development of a 3D-Printed Bionic Hand with Muscle- and Force Control

by

Florian Dannereder, Paul Herwig Pachschwöll, Mohamed Aburaia, Erich Markl, Maximilian Lackner, and Corinna Engelhardt-Nowitzki.

The

Best Student Poster

sponsored by the ABB-Group was awarded to the paper

Extension of the Action Verb Corpus for Supervised Learning

by

Matthias Hirschmanner, Stephanie Gross, Brigitte Krenn, Friedrich Neubarth, Martin Trapp, Michael Zillich, Markus Vincze.

Index of authors

Keynote Talks

Visual Tracking of Fast Moving Objects

Jiri Matas

Center for Machine Perception
Czech Technical University, Prague
E-mail: surname@cmp.felk.cvut.cz

Abstract
Visual tracking is a classical computer vision problem with many applications. I will first overview the diverse setting in which it has been studied: single view vs. multi-camera, color or intensity vs. RGBD, short-term vs. long-term, with vs. without a prior model of the tracked object. Next, I will discuss the state-of-the-art and the influence of the "CNN revolution" on the field. In the second part of the talk I will present a method for tracking of objects that move fast with respect to camera, at a speed that makes them appear as blurred streaks. Tracking of such object requires inversion of the image formation that involves a blurring and matting process. We will show that the presented method recovers the inter- and intra-frame trajectory in an interesting class of cases.

Automation Experience: An Experience Centered View Into Automated Contexts

Sebastian Egger-Lampl

Austrian Institute of Technology & University of Salzburg
E-mail: sebastian.egger-lampl(at)ait.ac.at

Abstract
User Experience is an omnipresent qualitative issue in almost all application contexts. More and more this is and has to be tackled as key ingredient of technological developments. The presentation will deliver a comprehensive view into the diverse facets of user experience, both from research as well as an industrial perspective. What automated contexts inhabited by the co-existence of autonomous artifacts could gain from a wider experience perspective? This includes the discussion of (future) relevant experience qualities (e.g. an interaction material view on interaction) as well the relevance of a wider over time-considerations of experience with automated systems. An example of quantitative analysis of human-to-human interaction and its transferability to human-machine interaction paradigms will depict the potential of novel experience approaches for human-machine collaboration. Furthermore, the presentation will also introduce a wider view on technology including acceptance as another success parameter.

.

Computer Vision for Complex Activities

Horst Bischof

Graz University of Technology
Institute of Computer Graphics and Vision
E-mail: bischof@icg.tugraz.at

Abstract
Understanding complex human activities is a requirement for efficient human-robot interaction as well as several other tasks in a production environment. This talk will highlight challenges that arise when analyzing complex human activities (eg., assembly tasks) with a computer vision system. We will demonstrate our recent work in that area and describe some of the major research challenges, including training these systems with minimal supervision, representations of complex activities etc.

Multi-contact Control and Locomotion with Torque Controlled Humanoids

Christian Ott

German Aerospace Center (DLR)
christian.ott<at>dlr.de

Abstract
During the last ten years several new humanoid robots have been presented, which allow for joint torque control as an alternative to classical position control. The low level torque interfaces in these robots are used for implementing force based control approaches, like whole-body impedance and compliance control. Recently, considerable progress has also been made in torque based walking on rough terrain. In this talk I will give an overview of these developments and I will present some recent results made with the robot TORO at DLR in the context of the EU project COMANOID in which humanoid robots have been evaluated in an aircraft manufacturing task. Finally, I will give an outlook on the next generation of physically compliant humanoids and their control approaches.

Austrian Robotics Workshop

Action Selection for Interactive Object Segmentation in Clutter

Timothy Patten, Michael Zillich and Markus Vincze

***Abstract*— Robots operating in human environments are often required to recognise, grasp and manipulate objects. Identifying the locations of objects amongst their complex surroundings is therefore an important capability. However, when environments are unstructured and cluttered, as is typical for indoor human environments, reliable and accurate object segmentation is not always possible because the scene representation is often incomplete or ambiguous. We overcome the limitations of static object segmentation by enabling a robot to directly interact with the scene with non-prehensile actions. Our method does not rely on object models to infer object existence. Rather, interaction induces scene motion and this provides an additional clue for associating observed parts to the same object. We use a probabilistic segmentation framework in order to identify segmentation uncertainty. This uncertainty is then used to guide a robot while it manipulates the scene. Our probabilistic segmentation approach recursively updates the segmentation given the motion cues and the segmentation is monitored during interaction, thus providing online feedback. Experiments performed with RGB-D data show that the additional source of information from motion enables more certain object segmentation that was otherwise ambiguous. We then show that our interaction approach based on segmentation uncertainty maintains higher quality segmentation than competing methods with increasing clutter.**

I. Introduction

Robots are becoming important for a diverse range of household applications, such as cleaning, tidying, gardening and personal care. These tasks require a robot to recognise, grasp and place items in their correct locations or use items appropriately. Distinguishing individual objects from other objects and the background is a first step for achieving these high-level actions. Making this distinction is a challenging problem, especially in human environments, because of the high variability of structure and the presence of clutter or occlusion. In these scenarios, it is difficult to accurately identify the portions of the data that belong to individual objects in a scene. Often the perceived environment is incomplete and this leads to an ambiguous interpretation. Typical segmentation approaches address these challenges by fine tuning parameters and exploiting structure (e.g., planar surfaces) or complex features. However, tuned parameters, structure and features for specific scenarios do not necessarily transfer to other scenarios. Consequently, the generality of these approaches is limited. For arbitrarily shaped objects or scenes with heavy occlusion, it is non-trivial to combine separated segments that in fact belong to a single object.

Fig. 1: Segmenting a cluttered scene with interaction. Left: Robot observing the scene. Middle: Robot inducing object motion. Right: Segmentation after interaction. Moved object (top left in blue) is correctly segmented.

In this paper we enable a robot to directly *interact* with the objects in a scene in order to resolve object ambiguity. The object motion induced by the robot is an additional source of information that is used in the segmentation process. Previous work on interactive segmentation apply only fixed motions; focus on segmenting an object from the background; or do not segment during interaction, only after each action is complete. Our work addresses these issues to enable interactive segmentation in real-world scenarios. First, we make use of a probabilistic segmentation approach to provide cues for where and how the robot should interact given the full state space of the manipulator, which might require complex motion plans. The scene is analysed and directly exposes the most uncertain regions where interaction is likely to improve object segmentation. Second, a combination of dense optical flow and sparse feature tracking is applied to monitor objects during interaction and to recover the motion of occluded objects (e.g., by the manipulator itself). This enables the identification of the individual motions of multiple objects with respect to each other and the background. Lastly, statistics about the pushed region of the scene are maintained during the interaction. The robot receives online feedback and updates its actions according to the segmentation quality at each time instance.

Experiments are performed with RGB-D data of cluttered scenes to first demonstrate the probabilistic segmentation process with moving objects. We benchmark against existing work of [22] and show that not only does interaction singulate objects and simplify object segmentation, but incorporating motion cues into the segmentation process leads to a significant performance boost. The second set of experiments compare the proposed active interaction strategy with competing methods. The results show that our approach maintains higher quality segmentation in comparison to the other methods as the amount of clutter increases.

*This work was supported by the European Community's Seventh Framework Programme under grant agreement No. 610532 SQUIRREL.

TP and MV are with the Vision for Robotics Laboratory, Automation and Control Institute, TU Wien, 1040 Vienna, Austria {patten, vincze}@acin.tuwien.ac.at

MZ is with Blue Danube Robotics, Vienna, Austria zillich@bluedanuberobotics.com

The remainder of this paper is organised as follows. Sec. II reviews related work. Sec. III provides an overview of the system. Sec. IV describes the probabilistic segmentation algorithm and Sec. V develops the active interactive segmentation approach. Sec. VI presents results for experiments with RGB-D data. Finally, the paper is concluded in Sec. VII with an outlook of future work.

II. Related Work

Interactive perception is the problem of interacting with the environment to improve perception quality and has been applied to many tasks such as recognition, modelling and pose estimation [2]. Interactive segmentation focuses on applying non-prehensile actions with a mobile manipulator to improve the separation of visual data into individual entities, i.e., segments. Pioneering work demonstrated the principle of interactive segmentation [10], [20], [16], which has been extended through numerous innovations. Many methods, however, are passive with respect to the segmentation task as actions are pre-planned according to expected object kinematics [15], [19] or any action expected to move an object hypothesis is chosen [17], [25], [23], [5], [11]. Accordingly, segmentation is only improved as a coincidence and actions are not chosen to directly segment objects in clutter.

Active approaches to interactive segmentation, on the other hand, specifically select actions that are expected to disambiguate a scene. A common technique is to apply actions that physically separate or "singulate" objects from a cluttered pile. This has been addressed with heuristics based on avoiding other local object clusters [4], spreading objects through orthogonal motions [12], the splitting plane between object hypotheses [14] or observed corners with local concavities [1], [13]. An alternative approach is presented by Eitel at al. [7], who use a neural network to learn favourable push actions. In contrast to these methods, we select push actions based on local segmentation uncertainty to most quickly resolve ambiguity. Resultingly, our method is driven by the observed data and does not rely on hand-crafted heuristics or training data.

Our approach to interactive segmentation exploits a probabilistic representation and an information-theoretic measure for action selection. This is similar in principle to the method of van Hoof et al. [26], who maintain a probability distribution over segmentation and select actions based on expected information gain. The main difference of our method is that we continuously monitor the interaction online and use the motion sequence instead of comparing the scene immediately before and after a push. An advantage of continuous tracking is that information about the trajectory information can be used to update the segmentation. Xu et al. [27] use information theory to select actions and also track segment hypotheses during interaction. However, this information directly fuses local regions into objects where as we apply Bayesian updates to the probabilistic edges of the graph. We maintain patch-wise relations over multiple interactions, which allows the system to recover from ambiguous observed motion that can sometimes result in over-segmentation.

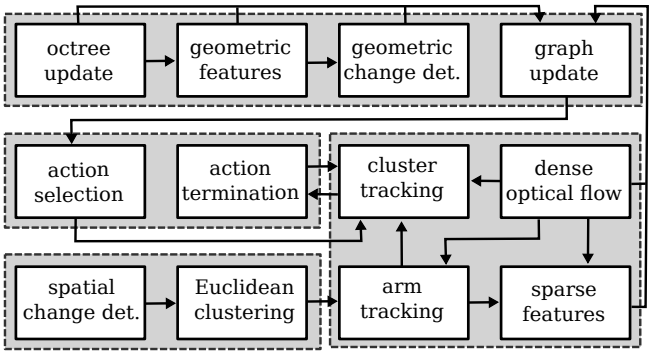


Fig. 2: System overview.

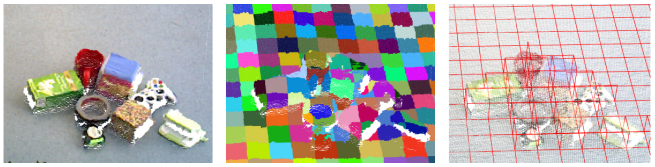


Fig. 3: Data structures for segmentation. Left: RGB-D input. Middle: Points binned in octree (coloured voxels). Right: Neighbourhood graph with edges connecting adjacent voxels.

III. System Overview

An overview of our system is illustrated in Fig. 2. The input point clouds are generated from an RGB-D sensor. The point clouds are stored in an octree, which is used to construct a neighbourhood graph. The edges in the graph represent the similarity between components of the scene and probability values are determined based on the similarity of geometric features. The edge probabilities are updated when objects are moved by exploiting dense optical flow and sparse feature tracking as well as geometric change detection as outlined in Sec. IV. The movement of objects is induced by the robot by selecting actions to best resolve segmentation ambiguity that is determined from the uncertainty in the neighbourhood graph as described in Sec. V. Online, during the action, the robot arm is identified and tracked to remove it from the segmentation and to account for occlusion. The scene, represented as clusters of nodes in the graph, is also monitored. The observed motion provides feedback that determines when the action is terminated. Finally, the process repeats by planning the next action with the updated neighbourhood graph.

IV. Probabilistic Object Segmentation

This section outlines the probabilistic object segmentation framework. Point clouds are stored in an octree and a neighbourhood graph is constructed. The probability of nodes belonging to the same object are computed according to the similarity of geometric features. When the scene is observed to move, optical flow and change detection are used to update edge probabilities.

A. Neighbourhood Graph

Each RGB-D input z_t at time t is stored in an octree as shown in Fig. 3 (middle). The neighbourhood graph $G =$

$(\mathscr{V},\mathscr{E})$ is initialised by generating graph nodes $i \in \mathscr{V}$ for every voxel. Edges $e(i,j) \in \mathscr{E}$ are generated between adjacent nodes i and j. An example of a neighbourhood graph is shown in Fig. 3 (right).

Each node i stores a collection of points as well as the centroid $\mathbf{p}^i \subseteq \mathbb{R}^3$, average normal vector $\mathbf{n}^i \subseteq \mathbb{R}^3$ and curvature $c^i \subseteq \mathbb{R}$ from the most recent RGB-D input. Edges in the graph encode the similarity between connected parts. The existence of an edge encodes the fact that the two parts are adjacent. In addition, a weight is assigned to each edge that measures their geometric similarity according to

$$g_t^{ij} = f_p(\mathbf{p}_t^i, \mathbf{p}_t^j) f_n(\mathbf{n}_t^i, \mathbf{n}_t^j) f_c(c_t^i, c_t^j). \quad (1)$$

The weight is the product of the independent probabilities given by the geometric properties of the nodes. We model the probabilities with the exponential functions

$$f_p(\mathbf{p}_t^i, \mathbf{p}_t^j) = \exp\left(-\frac{||\mathbf{p}_t^i - \mathbf{p}_t^j||^2}{2\sigma_p^2}\right), \quad (2)$$

$$f_n(\mathbf{n}_t^i, \mathbf{n}_t^j) = \exp\left(-\frac{\left(\cos^{-1}(\mathbf{n}_t^i \mathbf{n}_t^j)\right)^2}{2\sigma_n^2}\right), \quad (3)$$

$$f_c(c_t^i, c_t^j) = \exp\left(-\frac{(c_t^i - c_t^j)^2}{2\sigma_c^2}\right), \quad (4)$$

for the Euclidean distance between point centroids (2), angular difference between normals (3) and scalar difference between curvatures (4). These features capture spatial compactness and smoothness. The constants σ_p, σ_n and σ_c scale the values. In our experiments, we set σ_p to 0.85 times the voxel resolution, σ_n to 0.45π and σ_c to 0.5.

Edge relations between small patches cannot comprehensively identify similarity, especially for objects with sharp edges or irregular shape. Therefore, we combine learned features to overcome problems of locality. To achieve this, the method of [22] is performed on the original input. This procedure generates a list of geometrically consistent patches $\mathscr{P}$ from a pre-segmentation stage and computes pairwise relations based on their features. Patches are grouped by performing a graph cut. This information is mapped to the neighbourhood graph G to compute the final edge weight

$$w_t^{ij} = \alpha g_t^{ij} + (1-\alpha) l_t^{ij}, \quad (5)$$

where $l_t^{ij} = 0.9$ if i and j belong to the same patch in $\mathscr{P}$ or 0.1 otherwise. In our experiments, we set $\alpha = 0.5$.

Single input methods generate the most likely segmentation from the observed scene. We instead improve upon the initial segmentation by accumulating more data overtime. The graph is preserved and the edge weights are updated by inducing and observing motion.

B. Dense Optical Flow and Sparse Feature Tracking

The motion of the scene is monitored using a combination of dense optical flow and sparse feature tracking. Dense optical flow is determined for an RGB image by tracking pixel gradients using the method of Farnebäck et al. [8]. This assigns a flow vector to each pixel that is then projected back to the point cloud. Regions occluded by the arm are tracked by computing a Lucas Kanade (LK) [18] feature for the corresponding pixels. When interaction is terminated and the manipulator is removed from the scene, the sparse features are matched to pixels in the final RGB image using the pyramidal implementation of the LK feature tracker [3]. Additionally, if dense optical flow fails at any point during interaction, LK features are computed from the previous frame and added to the set of sparse features. These are also matched to the scene when interaction stops. This addition of tracking sparse points assists the overall flow estimation because it is more robust to occlusions. It enables flow to be computed for portions of the scene that become temporarily occluded, for example, by the manipulator or other objects. An example of dense optical flow for an intermediate frame during interaction is shown in Fig. 4 (bottom row, third from left). In Fig. 4 (bottom row, right) the image pixels are coloured to distinguish the type of motion detection. Red indicates dense optical flow, yellow indicates tracked LK features and blue indicates the static scene (flow with magnitude less than the octree resolution).

After interaction, the octree is updated with the final point cloud and graph nodes are regenerated. All pixels that are tracked during motion are used to estimate the flow for the graph nodes. For each voxel represented by a graph node, the voxel motion vector $\mathbf{m}_t^i \subseteq \mathbb{R}^3$ is computed by averaging the flow vectors from each point that belonged to the voxel prior to interaction. The new location of the voxel is computed according to

$$\mathbf{p}_{\text{new}}^i = \mathbf{p}_{t-1}^i + \mathbf{m}_t^i. \quad (6)$$

The identity i at time t is determined as the voxel in $\mathscr{V}$ nearest to $\mathbf{p}_{\text{new}}^i$.

In some cases, flow may have been corrupted and the previous identity of a voxel cannot be found. These voxels initialise nodes with no history.

C. Change Detection

Further robustness is achieved by detecting change in the scene. In most cases, optical flow suffices to estimate object motion. However, under strong occlusion, where pixels never reappear, no history can be determined. Therefore, we also compute the change for voxels (and neighbours of these voxels) that do not have a valid flow vector assigned. Change detection is performed at the end of interaction by comparing the new values for the point centroid, normal and curvature with the previous values. The magnitude of change for node i at time t is given by

$$\Delta(i, z_t) = 1 - f_p(\mathbf{p}_t^i, \mathbf{p}_{t-1}^i) f_n(\mathbf{n}_t^i, \mathbf{n}_{t-1}^i) f_c(c_t^i, c_{t-1}^i), \quad (7)$$

where $0 \leq \Delta(i, z_t) \leq 1 \ \forall i,t$.

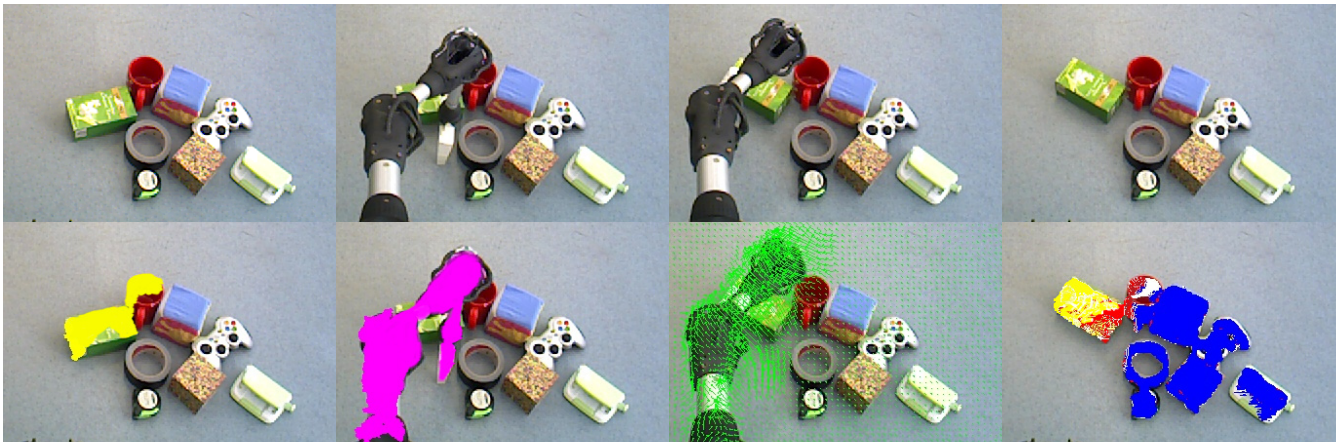


Fig. 4: Example of motion analysis. Top: Initial scene, introduction of arm, snapshot of arm moving an object and arm removed before next interaction. Bottom: Occluded regions on objects that initialise the set of LK features, arm identified with spatial change detection, dense optical flow field and motion detection (successful dense optical flow in red, static in blue, tracked LK features in yellow, invalid flow in white and background/ground unmodified).

D. Recursive Object Segmentation

The flow vectors computed in Sec. IV-B determine the previous identity of each graph node at the previous time instance (i.e., before the interaction). Correspondingly, neighbouring nodes that were also neighbours before the interaction will maintain their edge. This probability is updated by Bayes' rule where the new similarity probability is the likelihood and the previous probability is the prior.

Consider an edge $e(i,j)$ between two nodes i and j at time t. Let the continuous random variable x^{ij} denote the state between two nodes in the graph i and j. The probability of two nodes belonging to the same object given the RGB-D measurements is expressed by the conditional probability $p(x^{ij}|\mathbf{z}_{0:t})$ where $\mathbf{z}_{0:t} = \{\mathbf{z}_0, \ldots, \mathbf{z}_t\}$ is the history of measurements. If these nodes have valid motion vectors and if the edge exists at time $t-1$, then the nodes were also neighbours. In this case, $p(x^{ij}|\mathbf{z}_{0:t-1})$ is known, otherwise, the nodes were not neighbours and the similarity probability is unknown, therefore a uniform prior is assigned, i.e., $p(x^{ij}|\mathbf{z}_{0:t-1}) = 0.5$. The probability of any edge $e(i,j)$ between nodes i and j at time t given the *history* of measurements can be computed according to Bayes' rule

$$p(x^{ij}|\mathbf{z}_{0:t}) = \frac{p(x^{ij}|\mathbf{z}_{0:t-1})p(\mathbf{z}_t|x^{ij})}{p(\mathbf{z}_t|\mathbf{z}_{0:t-1})}, \tag{8}$$

where $p(x^{ij}|\mathbf{z}_{0:t-1})$ is the prior, $p(\mathbf{z}_t|x^{ij}) = w_t^{ij}$ is the measurement likelihood and the denominator is given by

$$p(\mathbf{z}_t|\mathbf{z}_{0:t-1}) = \int p(\mathbf{z}_t|x^{ij})p(x^{ij}|\mathbf{z}_{0:t-1})dx. \tag{9}$$

The posterior maintains the total probability after many point cloud inputs. Consequently, neighbouring nodes that move along similar trajectories will recursively update their similarity probability. On the other hand, nodes that become neighbours but have no history are initialised with the similarity probability of the most recent observation due to the uniform prior.

The Bayesian updates require the integral in (9) to be computed. To keep the updates tractable, we compute the denominator with

$$\begin{aligned} p(\mathbf{z}_t|\mathbf{z}_{0:t-1}) \quad \approx \quad & p(\mathbf{z}_t|x^{ij})p(x^{ij}|\mathbf{z}_{0:t-1}) \; + \\ & p(\mathbf{z}_t|\neg x^{ij})p(\neg x^{ij}|\mathbf{z}_{0:t-1}), \end{aligned} \tag{10}$$

where $p(\mathbf{z}_t|\neg x^{ij}) = 1 - p(\mathbf{z}_t|x^{ij})$ and $p(\neg x^{ij}|\mathbf{z}_{0:t-1}) = 1 - p(x^{ij}|\mathbf{z}_{0:t-1})$. This approximates the random variable to a discrete domain with two possible states x^{ij} and $\neg x^{ij}$.

Edges between nodes without a valid flow vector are updated using the change detection. The prior probability is computed as

$$p(x^{ij}|\mathbf{z}_{0:t-1}) = \min\left(0.5, \min(\Delta(i,z_t), \Delta(j,z_t))\right). \tag{11}$$

When the magnitude of the change for both nodes is greater than 0.5, the prior is set to the minimum change value and the resulting posterior $p(x^{ij}|\mathbf{z}_{0:t}) > p(\mathbf{z}_t|x^{ij})$. The increase of the posterior from the likelihood captures the fact that the nodes likely moved but the optical flow failed. If neither or both change magnitudes are less than 0.5, the prior is set to a uniform probability, $p(x^{ij}|\mathbf{z}_{0:t-1}) = 0.5$. Applying (8) gives $p(x^{ij}|\mathbf{z}_{0:t}) = p(\mathbf{z}_t|x^{ij})$. Here, the magnitude of the change is not sufficient to increase the similarity probability.

V. Active Interactive Object Segmentation

This section describes the active interaction approach for improving object segmentation. The uncertainty from the probabilistic segmentation is exploited to generate contact points and directions where interaction is considered to be most useful. During interaction the scene is monitored. The robot gathers evidence in real time to determine if the consequence of the planned action can assist the segmentation. When sufficient data is collected, the action is terminated so that the process can repeat.

A. Action Determination

Determining the best push action first generates a set of candidate pushes, then evaluates the quality of each push and finally selects the push with the highest quality. Unlike other methods that sample the action space, e.g., [26], our set of candidate push actions is generated by the geometry of the scene and it captures all possible stable actions. Evaluating the actions exploits the uncertainty from the probabilistic segmentation. The uncertainty provides very informative cues about which regions of the scene are most ambiguous and therefore which regions would most benefit from interaction.

1) Action Candidate Generation: Neighbouring nodes are clustered if their edges have high similarity probability as well as if the nodes have similar normal vectors. Merging nodes with high similarity attempts to group nodes into reasonable object hypotheses. Enforcing nodes to have similar normal vectors results in clusters with smooth surfaces, which are more easily pushed than irregular surfaces.

The contact point for each cluster is computed as the median of all points belonging to the nodes in the cluster. Each candidate generates two potential push vectors in the orthogonal directions to the normal of the surface represented by the cluster. The initial set of push candidates is refined by removing pushes that are perpendicular to the support plane (e.g., floor or table). These push directions imply pushing into or away from the fixed surface. Furthermore, candidates are removed if they are not accessible by the manipulator. This occurs if a collision free path to the contact point cannot be found or if the contact point is beyond the reach of the manipulator. The set of candidate push actions is denoted $\mathscr{A}$.

2) Action Evaluation: Every candidate action $\mathbf{a} \in \mathscr{A}$ is evaluated by computing a score

$$s(\mathbf{a}) = \frac{\bar{H}}{d_c + d_\theta + d_m + N_c}. \tag{12}$$

This assigns a high score to candidates with high entropy (uncertainty) $\bar{H}$ and divides the values by the distance to the circumference of the clutter pile (d_c), the length of the chord from the contact point to the clutter pile circumference (d_θ), the distance to the end effector (d_m) and the number of points in the point cloud in the immediate vicinity of the cluster along the push direction (N_c). The entropy $\bar{H}$ represents the uncertainty of the surrounding segmentation of the cluster and it is computed by averaging the entropy of the edges from every node within the cluster to every node not in the cluster. The distance from the push contact point to the circumference of the clutter pile d_c and the length of the chord from the contact point to the clutter pile circumference d_θ favour actions that are more likely to singulate an object. Intuitively, an object is pushed further away from other objects if it is near to the circumference of the clutter pile and the push direction is short, i.e., towards the circumference and away from the centre of the clutter. The distance to the end effector d_m favours clusters that are nearer for manipulation and therefore require less motion. The number of points N_c represents the likelihood of colliding the cluster into another object. This would be counter productive for segmentation because objects should ideally be pushed individually.

3) Action Selection: The best action is selected as the push with the highest score. Formally, the selected action is given by

$$\mathbf{a}^* = \arg\max_{\mathbf{a} \in \mathscr{A}} s(\mathbf{a}). \tag{13}$$

B. Action Termination

During interaction the robot monitors the selected cluster. The robot reasons about the impact the induced motion will have on the subsequent segmentation quality and decides to terminate the action if there is sufficient information to confidently separate or merge parts of the scene.

Denote the set of neighbours to cluster $\mathbf{c}$ as $\mathscr{N}(\mathbf{c})$. During the push, optical flow is computed for every pixel of $\mathbf{c}$ and $\mathscr{N}(\mathbf{c})$. The optical flow is projected back to the point cloud to generate a motion vector for each point in 3D and the distance moved by each point is calculated. At each time t, the average distance from the start location for every point in the cluster is computed, denoted $\bar{d}_t^{\mathbf{c}}$. Then for each neighbouring node $n \in \mathscr{N}(\mathbf{c})$ we compute the following

$$\sigma_t^n = \sqrt{\frac{\sum_{k=1}^{N^n} (d_t^{nk} - \bar{d}_t^{\mathbf{c}})^2}{N^n - 1}}, \tag{14}$$

where d_t^{nk} is the distance traveled by point k of neighbour n that has total N^n points. Intuitively, this quantity is similar to the standard deviation, but it is computed with the mean value of the cluster. For this case, σ_t^n quantifies the amount of variation with respect to the cluster. A low value means that the distance travelled by neighbouring points is close to the mean distance of the cluster. Consequently, the neighbour has moved in a similar way as the cluster that was pushed and the neighbour is likely to belong to the cluster. On the other hand, a high value means that the distance travelled by neighbouring points is spread over a wide range of values and differ greatly to the average distance travelled by the cluster. In this case, a significant portion of the neighbour has moved differently to the pushed cluster and the components of the neighbour likely belong to different objects.

Furthermore, the pushed cluster is monitored and the variation of the distance of its points is measured with

$$\sigma_t^{\mathbf{c}} = \sqrt{\frac{\sum_{k=1}^{N^{\mathbf{c}}} (d_t^{\mathbf{c}k} - \bar{d}_t^{\mathbf{c}})^2}{N^{\mathbf{c}} - 1}}, \tag{15}$$

where $d_t^{\mathbf{c}k}$ is the travelled distance of point k with $N^{\mathbf{c}}$ points in total. This is precisely the standard deviation: High variation suggests that some parts of the cluster moved differently, therefore, the parts may belong to different objects.

This gives rise to three termination criteria

$$\forall n \in \mathscr{N}(\mathbf{c}) \qquad \bar{d}_{\mathbf{c}}^t > \delta_1 \wedge \sigma_t^n < \delta_2, \tag{16}$$

$$\exists n \in \mathscr{N}(\mathbf{c}) \quad \text{s.t.} \quad \sigma_t^n > \delta_3, \tag{17}$$

$$\sigma_t^{\mathbf{c}} > \delta_4. \tag{18}$$

The first criterion (16) means that the cluster has moved a significant distance δ_1 and all neighbours have moved a similar distance determined by the threshold δ_2. The second criterion (17) means that there is at least one neighbour that has moved in a different manner to the cluster as determined by the threshold δ_3. The third criterion (18) means that the cluster itself has accumulated a high amount of variation as determined by the threshold δ_4. In our experiments, we set the values to δ_1 to four times the voxel resolution and d_2, d_3, and d_4 are set to 0.5.

C. Arm Tracking

A major limitation of previous interactive segmentation work is the confinement of studying pairs of RGB-D frames or RGB images, where an observation is made before and after individual fixed-length pushes. The reason is because of the degradation of feature tracking algorithms caused by occlusion created by the tool that moves the objects.

To increase robustness to the occlusion, the manipulator of the robot is explicitly tracked during interaction. Before objects are pushed, the manipulator is expected to be the only moving object in the scene. The manipulator is identified by using spatial change detection in 3D. Euclidean clustering is then performed and the points belonging to the largest cluster are considered the manipulator, smaller clusters are considered noise. Each point of the manipulator is continuously tracked with dense optical flow and removed from the input to avoid association of the arm to the scene. Furthermore, the region of the scene occluded by the manipulator is identified. The corresponding nodes initialise the set of sparse features that are matched to the scene at the end of interaction once the manipulator moves to a new position. Details of dense and sparse tracking follows the procedure in Sec. IV-B.

An example of identifying the manipulator and tracking occluded points is shown in Fig. 4. The manipulator is detected (bottom row, second from left) and the occluded voxels are determined in the initial frame (bottom row, left) then matched to points in the final frame (bottom row, right).

Complex or fast motion may cause tracking failure. However, this method is sufficient for relatively short and slow push actions. More sophisticated manipulator tracking and removal, e.g., [6], could be implemented if necessary.

VI. Experiments

This section presents results from experiments with cluttered scenes of objects. The first set of experiments demonstrates the recursive probabilistic segmentation method to correctly identify the ambiguous regions and compares the segmentation accuracy against a static single input method. The second set of experiments showcases the interaction planning strategy by comparing performance against competing methods.

A. Metrics and Ground Truth Generation

In order to evaluate our interactive segmentation approach, the most likely segmentation is generated by performing a graph cut on the neighbourhood graph [9]. For all experiments, we set the cut threshold to 0.5. Segmentation performance is measured by adapting the method in [21] for point clouds. Each object from the segmented point cloud is compared to the ground truth labelled objects. The largest overlapping object is assigned to each ground truth object. The precision is computed as the fraction of points assigned to the ground truth object and the recall is the fraction of points from the ground truth assigned to the object. Precision and recall are combined to compute the F1 score, $F1 = 2 \times \text{precision} \times \text{recall} / (\text{precision} + \text{recall})$, which is the harmonic mean between the two quantities.

Ground truth labelling is computed at the beginning and end of each trial, no ground truth can be acquired for the intermediate stages when multiple interactions are applied. Ground truth segmentation is generated using the method of [24]. We place (or remove, for the end scene) one object at a time and record a point cloud. Scene differencing on the depth data is run at each stage of the sequence to detect the introduction (or loss) of an object. These points are assigned a unique label. The final scene has one label for the background and a unique label for each object.

B. Recursive Segmentation

The first set of experiments analyses our interactive segmentation approach in comparison to single frame segmentation. As a benchmark, we choose the method of [22], which will be referred to as *static* segmentation. A total of 10 cluttered scenes with two to eight objects were randomly generated. Our interactive segmentation approach applied a random number of pushes for each scene before terminating.

The precision, recall and F1 score for all object instances are shown in Fig. 5. For interactive segmentation, the result is computed from the graph cut output on the neighbourhood graph after the final interaction. For static segmentation, the results are generated by directly segmenting either the final or initial point clouds. Fig. 5 shows that interactive segmentation achieves a higher accuracy than static segmentation and that the final scene is segmented more accurately than the initial scene on average. This outcome highlights two important aspects about interactive segmentation. Firstly, intelligent induced motion improves accuracy for static segmentation approaches. This is often due to the clutter being dispersed and objects becoming singulated after being pushed. Secondly, although the final scene is more easily segmented, accumulating the information during a sequence of interactions leads to an even further boost in performance.

In Fig. 6 we plot the difference in F1 score between interactive and static segmentation on the final scene for each trial. The F1 scores are computed by averaging the scores of each object in the scene. This shows that most trials ($8/10$) are segmented better when motion information is incorporated. In two trials, interactive segmentation performs worse than the benchmark. This can occur sometimes when multiple large portions of the scene move together and objects become under-segmented.

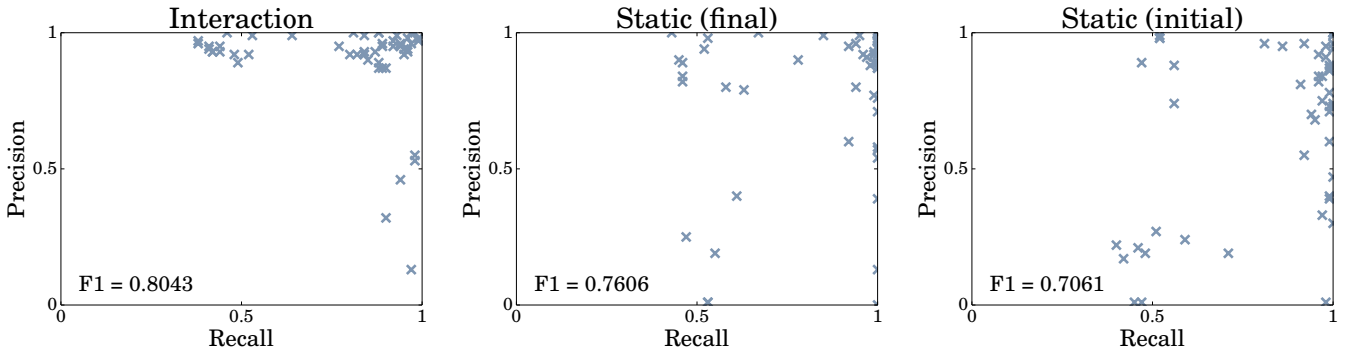


Fig. 5: Precision and recall of all objects for 10 sets of randomly cluttered scenes of varying number of objects and applied interactions. F1 score shown in bottom left corner. Left: Interactive segmentation. Middle: Static segmentation performed on the final scene (after all interactions). Right: Static segmentation performed on the initial scene (before any interaction).

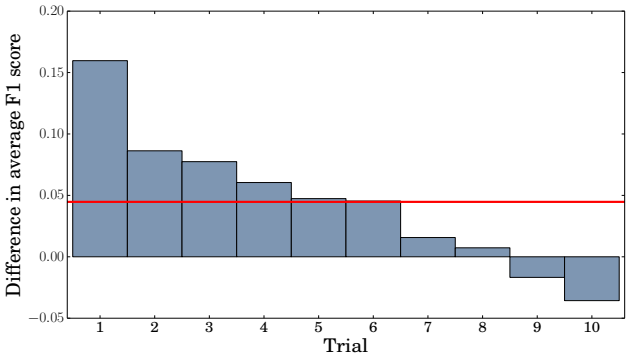


Fig. 6: Difference between segmentation accuracy for recursive over static method for 10 trials (sorted highest to lowest). Results for each trial is computed by averaging the F1 scores of the individual objects in the scene. Average improvement over all trials shown in red.

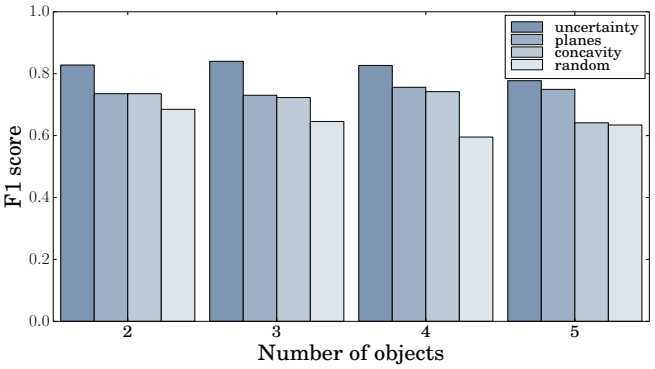


Fig. 7: Segmentation accuracy for different action selection strategies with increasing number of objects (clutter). Each bar represents F1 score by averaging three separate trials.

C. Action Selection

The second set of experiments evaluates the action selection strategy in comparison to other methods. In particular, comparisons are made with our implementations of pushing into concave corners [13], pushing along splitting planes [14] and random action selection. In these experiments, three interactions are made for each trial. Three trials are performed for each approach with sets of objects increasing in number from two to five. The segmentation accuracy for all trials is summarised in Fig. 7.

The results show that segmentation accuracy declines for all methods as clutter is increased (more objects are introduced). However, the segmentation accuracy using uncertainty to select actions declines at a slower rate. Segmentation accuracy when applying actions that attempt to split planes also remains almost constant but with lower accuracy than using uncertainty. Pushing into concavities performs well with few objects but suffers a significant performance decrease when the number of objects increases. The random strategy performs worst overall.

Intuitively, the methods of using uncertainty or splitting planes tend to push flatter surfaces and therefore often push a single object. Pushing into concavities does not consider the object hypotheses in the scene, consequently pushes tend to slip between objects. This often leads to singulation when the number of objects is small, but often leads to subsequent collisions when the number of objects is large. The collisions lead to difficult motions that result in incorrect segmentation decisions, hence the decrease in performance with five objects.

An example of action selection by each method is shown in Fig. 8. The push proposed by disambiguating uncertainty (white) targets a corner of a box. Corners can often be ambiguous because the local structure lacks smoothness. This is confirmed in the right of Fig. 8, which shows the corresponding segmentation uncertainty, computed by averaging the entropy of the edges for each node in the neighbourhood graph. Pushing into concave corners (cyan) often separates touching objects but this can sometimes be a wasteful action when objects are already easily segmented. Pushing along splitting planes (magenta) tries to confirm if object hypotheses on the opposite side of a visual edge belong together without considering the underlying segmentation algorithm. Our method, on the other hand, directly probes the scene where it is most ambiguous.

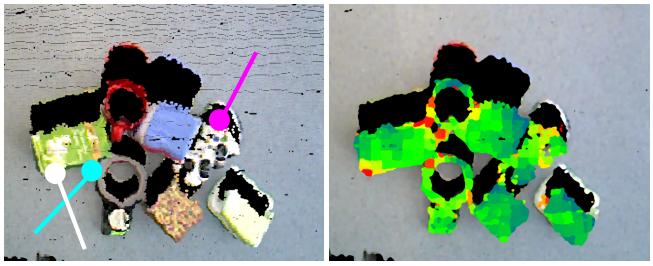


Fig. 8: Left: Example of the push actions chosen by the different interactive segmentation strategies. *Uncertainty* shown in white, *splitting planes* shown in magenta and *concavities* shown in cyan. Right: Heat map (green to red) of entropy in the neighbourhood graph, computed by averaging the entropy of the edges extending from each node.

VII. Conclusion

We have presented a method for improving object segmentation in clutter by directly interacting with the scene. Our approach identifies the uncertainty of object segmentation and selects actions that directly probe these regions in order to resolve the ambiguity. Using our proposed active approach, we are able to improve segmentation accuracy in comparison to a static segmentation method as well as maintain higher quality segmentation in cluttered scenes in comparison to competing action selection strategies.

One limitation of our work is that the camera must remain fixed. We intend to address this by incorporating SLAM or visual odometry to allow the robot to move with its base. This will permit a larger range of actions to be applied as well as more information to be gathered from new view points. We also plan to extend this to the problem of clearing piles of clutter, whereby confident object segmentation from interaction will be an important behaviour to enable successful recognition, pose estimation and grasping.

Acknowledgements

The authors thank Markus Suchi and Kiru Park for helping with generating ground truth data.

References

[1] C. Bersch, D. Pangercic, S. Osentoski, K. Hausman, Z. C. Marton, R. Ueda, K. Okada, and M. Beetz, "Segmentation of textured and textureless objects through interactive perception," in *Proc. of RSS: Workshop on Robots in Clutter: Manipulation, Perception and Navigation in Human Environments*, 2012, pp. 1–8.

[2] J. Bohg, K. Hausman, B. Sankaran, O. Brock, D. Kragic, S. Schaal, and G. S. Sukhatme, "Interactive perception: Leveraging action in perception and perception in action," *IEEE Trans. Robot.*, vol. 33, no. 6, pp. 1273–1291, 2017.

[3] J. Y. Bouguet, "Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm," *Intel Corporation Microprocessor Research Labs*, 2000.

[4] L. Chang, J. R. Smith, and D. Fox, "Interactive singulation of objects from a pile," in *Proc. of IEEE ICRA*, 2012, pp. 3875–3882.

[5] K. Chaudhary, C. W. Au, W. P. Chan, K. Nagahama, H. Yaguchi, K. Okada, and M. Inaba, "Retrieving unknown objects using robot in-the-loop based interactive segmentation," in *Proc. of IEEE/SICE SII*, 2016, pp. 75–80.

[6] C. G. Cifuentes, J. Issac, M. Wüthrich, S. Schaal, and J. Bohg, "Probabilistic articulated real-time tracking for robot manipulation," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 577–584, 2017.

[7] A. Eitel, N. Hauff, and W. Burgard, "Learning to singulate objects using a push proposal network," in *Proc. of ISRR*, 2017, pp. 1–13.

[8] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Proc. of SCIA*, 2003, pp. 363–370.

[9] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, 2004.

[10] P. Fitzpatrick, "First contact: An active vision approach to segmentation," in *Proc. of IEEE/RSJ IROS*, 2003, pp. 2161–2166.

[11] L. K. L. Goff, G. Mukhtar, P. H. L. Fur, and S. Doncieux, "Segmenting objects through an autonomous agnostic exploration conducted by a robot," in *Proc. of IEEE IRC*, 2017, pp. 284–291.

[12] M. Gupta, J. Müller, and G. S. Sukhatme, "Using manipulation primitives for object sorting in cluttered environments," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 2, pp. 608–614, 2015.

[13] K. Hausman, F. Balint-Benczedi, D. Pangercic, Z. C. Marton, R. Üda, K. Okada, and M. Beetz, "Tracking-based interactive segmentation of textureless objects," in *Proc. of IEEE ICRA*, 2013, pp. 1122–1129.

[14] T. Hermans, J. M. Rehg, and A. Bobick, "Guided pushing for object singulation," in *Proc. of IEEE/RSJ IROS*, 2012, pp. 4783–4790.

[15] D. Katz, M. Kazemi, J. A. Bagnell, and A. Stentz, "Interactive segmentation, tracking, and kinematic modeling of unknown 3D articulated objects," in *Proc. of IEEE ICRA*, 2013, pp. 5003–5010.

[16] J. Kenney, T. Buckley, and O. Brock, "Interactive segmentation for manipulation in unstructured environments," in *Proc. of IEEE ICRA*, 2009, pp. 1377–1382.

[17] E. S. Kuzmič and A. Ude, "Object segmentation and learning through feature grouping and manipulation," in *Proc. of IEEE-RAS HUMANOIDS*, 2010, pp. 371–378.

[18] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. of IJCAI*, 1981, pp. 674–679.

[19] R. M. Martín and O. Brock, "Online interactive perception of articulated objects with multi-level recursive estimation based on task-specific priors," in *Proc. of IEEE/RSJ IROS*, 2014, pp. 2494–2501.

[20] G. Metta and P. Fitzpatrick, "Better vision through manipulation," *Adapt. Behav.*, vol. 11, no. 2, pp. 109–128, 2003.

[21] E. Potapova, M. Zillich, and M. Vincze, "Attention-driven segmentation of cluttered 3D scenes," in *Proc. of ICPR*, 2012, pp. 3610–3613.

[22] A. Richtsfeld, T. Mörwald, J. Prankl, M. Zillich, and M. Vincze, "Segmentation of unknown objects in indoor environments," in *Proc. of IEEE/RSJ IROS*, 2012, pp. 4791–4796.

[23] D. Schiebener, A. Ude, and T. Asfour, "Physical interaction for segmentation of unknown textured and non-textured rigid objects," in *Proc. of IEEE ICRA*, 2014, pp. 4959–4966.

[24] M. Suchi, T. Patten, D. Fischinger, and M. Vincze, "Easylabel: A semi-automatic pixel-wise object annotation tool for creating robotic RGB-D datasets," in *Proc. of IEEE ICRA*, 2019, (accepted).

[25] A. Ude, D. Schiebener, N. Sugimoto, and J. Morimoto, "Integrating surface-based hypotheses and manipulation for autonomous segmentation and learning of object representations," in *Proc. of IEEE ICRA*, 2012, pp. 1709–1715.

[26] H. van Hoof, O. Kroemer, and J. Peters, "Probabilistic segmentation and targeted exploration of objects in cluttered environments," *IEEE Trans. Robot.*, vol. 30, no. 5, pp. 1198–1209, 2014.

[27] K. Xu, H. Huang, Y. Shi, H. Li, P. Long, J. Caichen, W. Sun, and B. Chen, "Autoscanning for coupled scene reconstruction and proactive object analysis," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 177:1–177:14, 2015.

Intuitive Human Machine Interaction based on Multimodal Command Fusion and Interpretation

Leopold Hiesmair[1], Matthias Plasch[1], Helmut Nöhmayer[1] and Andreas Pichler[1]

***Abstract*— The drastic transition from mass production to mass customization and small lot-sizes in production industry, requires intuitive interaction, programming and setup approaches for machinery and robotics in order to reduce setting-up time or adaption effort. Multimodal data fusion and analysis is considered as a potential enabling technology to achieve intuitive human machine interaction. Our work focuses on robust interpretation of commands, issued by a human actor, which are combined of single attributes created from different multimodal channels. The presented approach is demonstrated using an example of human robot interaction, where the user interacts with the robot to setup a robotic process sequence.**

I. Introduction

Despite the increasingly simple programming interfaces that are available for production machinery, a deeper understanding of these systems is needed. For the programming, maintenance and adaption of processes, a highly qualified technician is required in many cases, who is only available to a limited extent. Therefore, the setting-up time is increased, which consequently reduces the flexibility. Our goal is to enable an intuitive communication with various modalities so that non-expert users are able to interact with and to program production machineries. This paper presents the conception and development of a multimodal data recognition and fusion system, to interpret and process commands issued trough an human actor. The multiple command parts provided by various channels, such as speech, gesture and haptic interaction, are analyzed and interpreted in order to generate a valid command statement. This fusion system is intended to be used for human machine interaction in general. Within the scope of this work, an example was developed which considers the domain of human robot interaction (HRI). The resulting *Command Fusion and Interpretation* (CFI) module enables robust control of a robot system based on simple interaction commands. The outline of the paper is as follows. Section II gives an overview on the state of the art in multimodal data analysis and discusses related work. In section III, requirements to multimodal data fusion and command interpretation are presented, following the description of our implementation approach in sections IV and V. Section VI presents the testing environment test procedures for our approach. The paper is concluded by a brief discussion about the results and explaining future work items in section VII and VIII.

[1] Profactor GmbH; Im Stadtgut A2, 4407 Steyr-Gleink, Austria; Email: name.surname@profactor.at

II. State of the art and related work

Multimodal data fusion and processing has gained a lot of interest in different research fields, especially in the areas of multimedia analysis [1][9][15][23][12], human machine (HMI) and human robot interaction (HRI) [22][3][10][8][6]. Atrey et al. [4] provide an extensive survey on fusion strategies for multimedia analysis, discuss general characteristics and common challenges arising during the implementation of multimodal data fusion.

According to [4] and [19] data fusion is performed on either *feature level (early fusion)* or *decision level (late fusion)*. *Feature level fusion* is often applied in case of strongly coupled inputs (like e.g. voice and lips movement). The extracted information of each modality is combined into one single vector, which is sent to the interpretation unit [23]. *Decision level fusion* combines local feature decisions into a vector. To derive a final decision, a synchronization between the different feature decisions is required, due to the different latencies of the classifiers [16].

The fusion approaches can be categorized in (application references are mentioned in brackets): a) *Rule based* ([11][5]) b) *Classification based* ([1][9][12]) and c) *Estimation based* ([15]). A detailed description of these categories is provided by Atrey et al. [4].

Rule based fusion approaches can be realized based on Definite Clause Grammars (DCGs) [17]. DCGs are proven to be helpful to describe natural language and are built-in features of first-order logic languages like Prolog [7]. A comprehensive overview on grammars for multimodal data processing is provided in [7]. Shimazu et. al. [21] proposed MultiModal DCGs (MM-DCGs), which provide means to express chronological constraints and handle multiple modalities. In [7] an approach for generating multimodal grammars is presented, which is able to add additional semantics to grammar definitions.

HRI is realized in different forms, depending on the information that has to be communicated, and dependent on the role of the human (e.g. supervisor or operator) [20]. The information exchange is considered as a main topic of HRI research, to enable intuitive and user-friendly interaction. Similar to [13] we consider HRI based on multiple modalities as relevant, to realize a more complex information flow. This is needed in situations where parameters (like speed) should be communicated at the same time as higher level commands, like a coarse moving direction. In such cases using a single modality is not enough to express the intention. Within the HRI domain, Sucar et al. [22] applied fusion of speech

and gesture data to generate commands valid to control motion of a robot, based on intuitive motion instructions. In [3] and [2] *incremental fusion* approaches for HRI are targeted, where incremental means that distinct multimodal data packages are processed as they are being received. This also requires an incremental generation of command hypotheses until a final statement can be found. Ameri et. al. [3] stress the necessity of weighting the different multimodal channels to cope with error prone modalities. The authors in [18] focus on a bidirectional interaction approach (pointing gestures, voice, status display), to enable confirmation of the interpreted command to achieve high accuracy.

The works [19] and [14] introduce general architectures to realize robust multimodal HRI. Rossi et. al. [19] implemented a decision level and classification based approach using a Support Vector Machine within the fusion layer. Support for an arbitrary number of modalities is given. The work in [14] focuses on deep learning based feature analyses units, to classify data of multimodal channels (body posture, hand motion, voice commands). A late fusion engine is part of future work topics. Multimodal interaction with a group of robots is targeted in [6]. Decision level fusion is applied based on Naive Bayes classifiers. This approach requires a three-step training process: a) one step to train the unimodal classifiers, b) one step for the command recognition system (structures of the possible commands) and c) one step to adapt the thresholds for the command hypotheses.

Our work targets the development of a multimodal data fusion and interpretation architecture based on DCGs in order to generate valid commands for human machine interaction. In this paper we consider the application of controlling a robot based on commands issued by a human through multimodal channels. Those commands are based on keywords that are intuitive to humans (e.g. move up, slower, stop, and others), similarly as proposed by [22]. Although DCGs provide truly less expressiveness like explained in [7][21], we argue that DCGs are still sufficient to perform multimodal data fusion with reasonable results, for the reasons as follows.

- Available commands including their structure and parameters are known in prior for the automation or robotics domain. There is no requirement for generating the validation grammar or to train command classifiers. Determined grammars are easier to understand and lead to determined results.
- Temporal constraints for multimodal fusion can be considered using timeouts, alternatively to applying MM-DCGs. As data fusion and hypotheses are performed continuously during sensing, timeouts basically specify the time duration for a full command to be issued.

The presented solution also allows for online adaptation of command structures during runtime, thus providing a high grade of flexibility. Our work focuses on multimodal data fusion and interpretation of a valid command. Classification of the data emitted through multimodal channels is not in the scope of this work.

III. Requirements

In order to realize robust command interpretation based on multimodal inputs, requirements as follows were defined for the implementation of the CFI module.

- **Partly reception of commands**
  A command (e.g. `move object1 to location2`) can be partly received (command parts, henceforth *attributes*) in a random order and from different input channels.
- **Input validation**
  All input streams need to be checked for plausibility using a collection of rules. Invalid attributes need to be ignored.
- **Confidence level for unreliable sources**
  Most recognition systems indicate a probability measure in relation to the understood attribute. The CFI module has to analyze this confidence level and decide whether it is sufficient or not. The threshold depends on the type of information and the assigned source.
- **Prioritization of commands and channels**
  In the case of multiple valid command hypotheses, operations with higher safety level have to be prioritized. Futhermore, if identical attributes are received from different multimodal channels, but not corresponding to each other (e.g. `move up down`), the system has to prioritize the channel with the higher quality.
- **Flexibility through adjustable parameters**
  Each configuration parameter (e.g. priorities, confidence levels, number of input channels, and others) has to be adjustable during the execution.
- **Feedback and none-feedback mode**
  A feedback mode to present the resulting command to the user is required. The user has the opportunity of aborting the command execution.
- **Heart Beat signals**
  Motion commands are mostly incrementally, i.e. that the system executes a command as long as it is active. A Heart Beat signal maintains the active state.
- **Adaptable command definition during Runtime**
  In different machine operating states (e.g. automatic, hand move, and others), specific commands are allowed. Therefore, a functionality to enable or disable commands during runtime is necessary.

IV. Approach

The general approach targeted by the CFI module is to analyze the received attributes in an incremental fashion. Using such a strategy, a robust way of fusing multimodal data inputs, and generating a valid command statement is achieved. Based on the defined requirements, the approach as described below was realized.

A. Architecture

Figure 1 illustrates a scheme of the architecture of the CFI module. On the left side, different input channels are listed. The blocks on the top of the figure represent the configurability and feedback functionality. A simplified description of the

CFI module's processing sequence is depicted at the bottom. The core functionalities are presented as block diagram in the image center. The meaning of the functional units will be described in section IV-C. The functional workflow of the

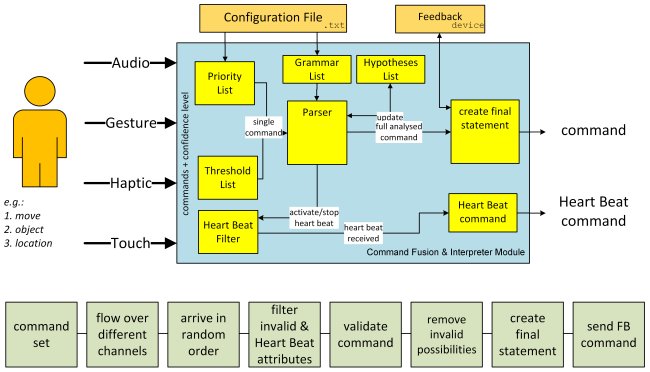


Fig. 1. Architecture of the CFI module

CFI module is as follows:

1) The user describes the desired command using different modality channels which are captured by the recognition systems.
2) All created command attributes are transmitted over several input channels to the CFI module.
3) The arrival times of the attributes are random and can overlap.
4) The CFI module filters invalid and active Heart Beat attributes to ensure a correct interpretation of the commands.
5) All attributes are validated by using the *Grammar List* and the *Threshold List*.
6) Each incremental step of validating command structures, the *Hypotheses List* is repetitively updated by removing invalid hypotheses.
7) If a command is fully received and analyzed, the module creates a final statement.
8) The created final statement is sent to the execution module.

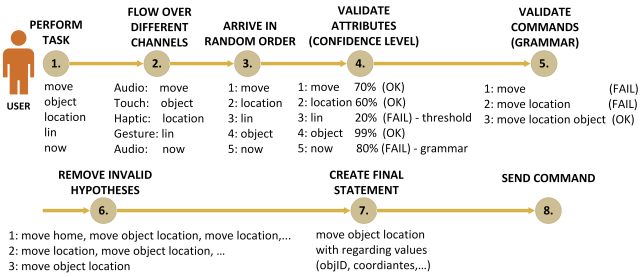


Fig. 2. Command fusion example

B. Working Principle

In order to get a better understanding of the CFI module, the following example in Figure 2 is considered, to explain the individual steps. The steps 1 to 3 are covered by individual recognition systems for multimodal channels and can not be influenced by the CFI module. Each command attribute arrives at the module in an random order and will be immediately validated in step 4. In this example, the invalid attribute `now` (not defined in the Grammar List) and the unreliable attribute `lin` (confidence level below threshold) are ignored. The incremental validation of the command is taken place in step 5 and 6, where each validation in step 5 creates command hypotheses in step 6. At every iteration, an attribute is appended to the command structure and hypotheses are generated. After all attributes are appended and validated, only one hypothesis is left and the final statement can be created and sent to the execution unit, as shown in step 7 and 8.

As mentioned in section III, Heart Beat signals are used to maintain the active state of a command (e.g `move up`). Figure 3 illustrates an example of a Heart Beat usage. Using the *Heart Beat start attribute* `up` repetitively, the active state is maintained. The defined interval time represents the maximum time span between two appearing Heart Beat attributes. If this time span is exceeded, the command will be automatically stopped, as highlighted with the expressions $T1 + 4s$ or $T3 + 4s$. Additionally, the command can be stopped immediately by using the *Heart Beat Stop attribute* `finish`, as shown in the last example of Figure 3.

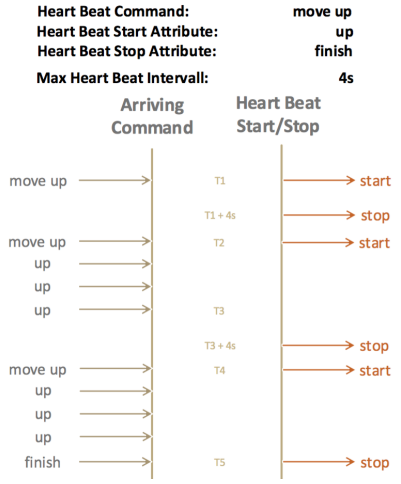


Fig. 3. Usage of Heart Beat attributes (e.g. start/stop)

C. Submodules and Functionalities

The depicted modules in Figure 1 basically consist of different lists and components that interact with each other. The purpose of each block/list is described below:

- **Input Channels**
  The input channels consist of different modalities. Those can be divided into unreliable channels, where recognition is based on trained classifiers (e.g. speech, gesture, haptic, etc.), and reliable channels with defined states (e.g. touch panel, button, etc.).
- **Priority List**
  The Priority List defines every available command with

their respective priority level. This level is used for prioritizing attributes in case of simultaneously arriving commands.

- **Grammar List**
  Unreliable modalities can cause unintended commands and attributes. Therefore, all invalid commands and attributes are filtered in order to prevent invalid states. The Grammar List summarizes all valid command structures with the valid combinations of attributes describing a possible command (e.g. `move object location`) using DCGs. Additionally, this list defines the output structure of the final statement.
- **Hypotheses List**
  After every validation of a command structure, a pool of hypotheses is created and stored into the Hypotheses List in order to reflect all further possibilities. The updated Hypotheses List indicates whether a command is fully received or not. After updating this list three cases are possible, depending on the list size:
  - Empty: command structure is not valid
  - Single entry: command structure is fully received
  - Multiple entries: command structure can be extended with further attributes
- **Threshold List**
  Several recognition systems use a distribution for the likeliness of the understood inputs (commands or attributes). These inputs are received with a corresponding confidence level and will only be accepted over a certain threshold. By analyzing the Hypotheses List a certain number of command attributes can be predicted to adapt the threshold levels for the respective attributes.
- **Parser**
  The multimodal Parser plays a central role in terms of analyzing the gathered information. It uses all lists for validating the commands and generating hypotheses. The algorithm is described in section V-B.
- **Output Sentence Structure**
  If a command is fully received, the CFI module creates a final output statement, which follows the defined command structure of the Grammar List.
- **Configuration File**
  The configuration file is required for configuring the behavior of the CFI module by adjusting the Priority List, Threshold List, feedback mode and other parameters.
- **Feedback**
  The feedback mode is enabled and adjusted by the configuration file. If activated, the validation algorithm conducts feedback from the user in order to ensure the correctness of the understood command. The command is illustrated at a device (e.g. screen) where the user is allowed to abort the analyzed command within a certain cancellation-timeout.
- **Heart Beat Filter**
  If a defined command structure includes a Heart Beat attribute (e.g. start or stop) the active state can be maintained by repeating the Heart Beat start attribute. During that repetition, the desired attribute is filtered to prevent an impact on the fusion algorithm.

After explaining the major functional blocks of the CFI module, the next section specifically focuses on an explanation of the fusion and interpretation workflow.

V. Algorithm

This section describes the relevant algorithm behind the CFI module. All incoming command attributes are validated, enriched with needed information such as timestamp and corresponding input channel and stored into the *Command Queue*. A prioritization algorithm sorts all commands in the queue and removes commands with lower priority. Lastly, the *Command Fusion Algorithm* is applied to form a full command statement.

A. Command Order Assumption

As defined in section III, the attributes can arrive in random order. This leads to complications in separating different commands. In the case of ambiguous command definitions, one attribute can belong to different commands. Therefore, the following assumptions were taken into account to provide a robust separation algorithm:

- Each command definition has to begin with a *Command Type* (e.g. `move, drill, sett`)
- Each received attribute has to arrive within a define time span to the last reception
- A command structure can be finished with a *Command End Key* (e.g. `go, ok, finish`)

B. Command Fusion Algorithm

The algorithm considered in this section combines all stored attributes to a final statement. The main challenge is a correct separation and validation of non-distinct commands. By applying the mentioned assumption, new commands will be accepted after exceeding the configured command times span or once a new Command Type is received.
Figure 4 shows the working principle of the command fusion. The upper part of the flowchart deals with the decision, whether the attribute to be analyzed belongs to the current command or to a new one. The decision is based on the command time span and a covered Command Type. If the attribute belongs to the same command, the Hypotheses List will be updated, otherwise, it will be renewed. With this new or updated Hypotheses List, the availability of the desired command can be analyzed. If no possibilities can be created from the collected attributes, we assume that the command is not defined and therefore not valid. But if possibilities are present in the Hypotheses List, the potential command is available. At this point, the specification of a fully received command has to be taken into account. Therefore, three conditions were established for defining a fully received command:

- Collected command attributes form a single hypotheses
- Configured command time span is exceeded
- Command End Key is received

If one of these conditions is satisfied, a fully received command is assumed. Otherwise, the algorithm remains in the waiting state until the command time span is exceeded.

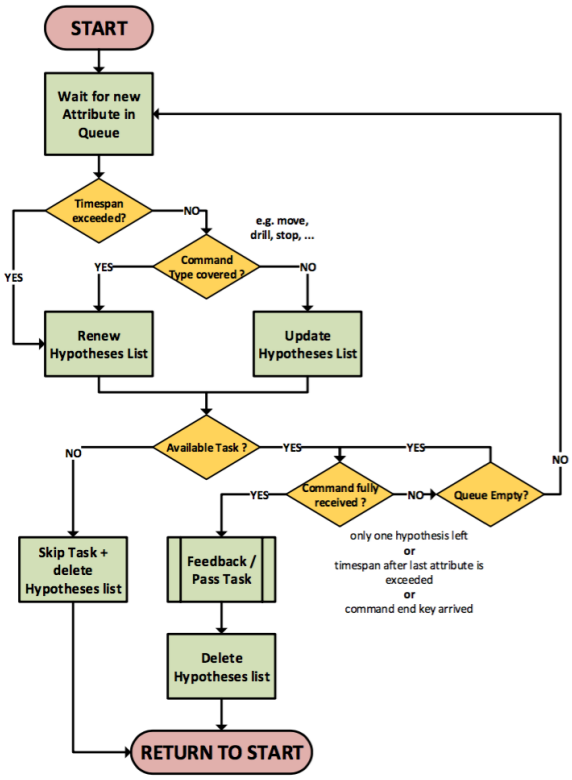


Fig. 4. Command fusion algorithm flowchart

VI. Testing and Environment

In order to test the CFI module, an environment is needed, including recognition systems as input channels such as speech, haptic, gesture and touch. Additionally, an execution module is needed for testing the impact and robustness of the output statement of the CFI module. Three different modalities over four recognition systems were used which includes the *CMUSphinx*[1] as an offline speech recognition system for low-resource platforms, the *Microsoft Kinect* and *Leap Motion*[2] as commercial gesture capturing devices specialized for hand or body motions and the *FBRT*[3] as a java based runtime environment of IEC 61499 function block for graphical user interfaces. The output of the CFI module is received by either an Universal Robot simulation (UR)[4] software URSim or the robot control unit of the real UR. All developed modules are executed using *FORTE*[5], which is an IEC 61499 compliant runtime environment. By combining these recognition and control systems, a testing framework can be built.

For testing the CFI module a representative use case was established for programming a robot. Therefore, two modes were established:

- **Teach Mode:**
  Robot can be moved and digital IOs can be set by using defined commands (e.g. `move up`, `gripper close`). All commands are stored with the corresponding timestamp and form a kind of a recipe. In this mode, all defined commands are available.
- **Replay Mode:**
  A built recipe will be executed to replay the robot movement. In this mode, only management and security commands are available.

A *Mode Manager* was developed for switching between these two modes and configuring the CFI module accordingly using management command (e.g. `record`, `replay`, `default`). All other commands are used for controlling a robot.

Test Case

Using the established test environment, a simple pick and place application was conducted in order to evaluate the usability and robustness of the system. A Universal Robot UR10 CB3 was equipped with a vacuum gripper and integrated in a worktable. The user had the task to use the available modalities (speech, hand gesture, body gesture and touch) in order to form statements to navigate the robot to a workpiece, grip it with the vacuum gripper and place it somewhere else on the table. Afterwards, the mode manager was used to record this procedure for repetition.

Based on this use case, a video was recorded, which explains the multimodal command fusion and interpretation functionalities. The clip can be retrieved from the following link[6].

VII. Discussion

The developed CFI module represents a technology to fuse and interpret predefined command structures from several input modalities. By configuring the module, the different input channels can be weighted via a priority key in order adapt the fusion process to the individual needs. Unreliable sources such as gesture and speech recognition can produce contradicting and invalid statements which were taken into account. Additionally, these sources often provide a confidence level related to the understood command, which are included into the fusion process as well. In order to provide a highly flexible module, all defined command structures can be adjusted during the execution time.

Systems which integrate a CFI module allow the non-expert users to use several modalities for programming, controlling or adjusting a machine. Due to the definable Grammar, which can be adjusted during runtime, the valid commands can be adapted according to the operators needs or experience level and create an intuitive interface. The command order assumption (discussed in section V-A), i.e. a

[1] `https://cmusphinx.github.io`
[2] `https://www.leapmotion.com`
[3] `https://www.holobloc.com`
[4] `https://www.universal-robots.com`
[5] `https://www.eclipse.org/4diac/en_rte.php`
[6] `https://youtu.be/AbJ8VaxxwzI`

key command (e.g. `move, drill, set`) has to be the first attribute, did not prove to be a limitation of the usability of the system. As known from different commercial recognition system, key words are used to indicate a command. Therefore, ordinary users natively initiates a operation with this key command and thus increases the usability of the system.

Using individual tailored Grammar for every user, no expert is needed for programming a machine. This increases the flexibility of a system, because highly qualified technicians are often temporarily unavailable.

The test use case indicated, that the used recognition systems were not accurate enough to gain a high usability of the system. The speech recognition was very sensitive to noise and therefore not usable during the active vacuum gripper. The generic command structures enabled programming up to a certain granularity. Thus, high accuracy programming turned out to be challenging. Parameterized commands (e.g. move to coordinates, move distance) are part of future work to increase the usability of the system. After a procedure was recorded properly, the replay functionality of the Mode Manager imitated the movements accurately.

VIII. Future Work

In order to gain a higher stability of generating complex command structures, investigations into new recognition systems are needed. Furthermore, the modalities should be evaluated regarding their usability of generating robot relevant commands. Additionally, the modalities have to be analyzed regarding the possibility of generating parameters for commands, e.g. coordinates, velocities or identifiers.
Since the CFI module was only tested by the developer team, new testing scenarios with ordinary users have to be established to test the stability and user-friendliness of the CFI system. Based on a survey of these users, the used robot commands can be evaluated in terms of intuitiveness.

IX. Acknowledgment

This work was funded by the project European research project SYMBIO-TIC[7] (European Commission; Horizon 2020 program, under grant agreement 637107).

References

[1] W. H. Adams, G. Iyengar, C.-Y. Lin, M. R. Naphade, C. Neti, H. J. Nock, and J. R. Smith, "Semantic indexing of multimedia content using visual, audio, and text cues," vol. 2003, no. 2, pp. 1–16.

[2] B. Akan, A. Ameri, B. Çürüklü, and L. Asplund, "Intuitive industrial robot programming through incremental multimodal language and augmented reality," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3934–3939.

[3] E. A. Ameri, B. Akan, and B. Çürüklü, "Incremental multimodal interface for human robot interaction," in *2010 IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1–4.

[4] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli, "Multimodal fusion for multimedia analysis: a survey," *Multimedia systems*, vol. 16, no. 6, pp. 345–379, 2010.

[5] B. Burger, I. Ferrané, F. Lerasle, and G. Infantes, "Two-handed gesture recognition and fusion with speech to command a robot," vol. 32, no. 2, pp. 129–147.

[6] J. Cacace, A. Finzi, and V. Lippiello, "Robust multimodal command interpretation for human-multirobot interaction." in *AIRO@ AI* IA*, 2017, pp. 27–33.

[7] A. D'Ulizia, F. Ferri, and P. Grifoni, "Generating multimodal grammars for multimodal dialogue processing," vol. 40, no. 6, pp. 1130–1145.

[8] G. A. Farulla, L. O. Russo, V. Gallifuoco, and M. Indaco, "A novel architectural pattern to support the development of human-robot interaction (HRI) systems integrating haptic interfaces and gesture recognition algorithms," in *2015 IEEE Computer Society Annual Symposium on VLSI*, pp. 386–391.

[9] M. Gandetto, L. Marchesooti, S. Sciutto, D. Negroni, and C. S. Regazzoni, "From multi-sensor surveillance towards smart interactive spaces," in *2003 International Conference on Multimedia and Expo, 2003. ICME '03. Proceedings*, vol. 1, pp. I–641–4 vol.1.

[10] R. Gomez, K. Nakamura, T. Kawahara, and K. Nakadai, "Multi-party human-robot interaction with distant-talking speech recognition," in *2012 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pp. 439–446.

[11] H. Holzapfel, K. Nickel, and R. Stiefelhagen, "Implementation and evaluation of a constraint-based multimodal fusion system for speech and 3d pointing gestures," in *Proceedings of the 6th International Conference on Multimodal Interfaces*, ser. ICMI '04. ACM, pp. 175–182.

[12] L. Kessous, G. Castellano, and G. Caridakis, "Multimodal emotion recognition in speech-based interaction using facial expression, body gesture and acoustic analysis," vol. 3, no. 1, pp. 33–48.

[13] E. Lakshantha and S. Egerton, "A diagrammatic framework for intuitive human robot interaction," *Journal of Ambient Intelligence and Smart Environments*, vol. 8, pp. 21–33, 01 2016.

[14] H. Liu, T. Fang, T. Zhou, Y. Wang, and L. Wang, "Deep learning-based multimodal control interface for human-robot collaboration," *Procedia CIRP*, vol. 72, no. 1, pp. 3–8, 2018.

[15] A. P. Loh, F. Guan, and S. S. Ge, "Motion estimation using audio and video fusion," in *Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th*, vol. 3, pp. 1569–1574 Vol. 3.

[16] S. Oviatt, A. DeAngeli, and K. Kuhn, "Integration and synchronization of input modes during multimodal human-computer interaction," in *Referring Phenomena in a Multimedia Context and Their Computational Treatment*, ser. ReferringPhenomena '97. Association for Computational Linguistics, pp. 1–13.

[17] F. C. Pereira and D. H. Warren, "Definite clause grammars for language analysis—a survey of the formalism and a comparison with augmented transition networks," *Artificial intelligence*, vol. 13, no. 3, pp. 231–278, 1980.

[18] C. P. Quintero, R. Tatsambon, M. Gridseth, and M. Jägersand, "Visual pointing gestures for bi-directional human robot interaction in a pick-and-place task," in *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 349–354.

[19] S. Rossi, E. Leone, M. Fiore, A. Finzi, and F. Cutugno, "An extensible architecture for robust multimodal human-robot communication," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2208–2213.

[20] J. Scholtz, "Theory and evaluation of human robot interactions," in *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*. IEEE, 2003, pp. 10–pp.

[21] H. Shimazu, S. Arita, and Y. Takashima, "Multi-modal definite clause grammar," in *Proceedings of the 15th conference on Computational linguistics-Volume 2*. Association for Computational Linguistics, 1994, pp. 832–836.

[22] O. M. I. E. Sucar, S. H. Aviles, and C. Miranda-Palma, "From HCI to HRI - usability inspection in multimodal human - robot interactions," in *The 12th IEEE International Workshop on Robot and Human Interactive Communication, 2003. Proceedings. ROMAN 2003*, pp. 37–41.

[23] M.-T. Yang, S.-C. Wang, and Y.-Y. Lin, "A multimodal fusion system for people detection and tracking," vol. 15, no. 2, pp. 131–142.

[7] `http://www.symbio-tic.eu`

Using High-Level Features and Neglecting Low-Level Features: Application Self-Localization

Felix König[1] and Markus Bader[2]

Abstract— **Common self-localization algorithms as well as trajectory algorithms for autonomous vehicles rely on low-level features such as laser readings. Identifying higher-level features or objects increases the system quality, but often contradicts the sensor noise model used, especially in the case of dynamic features such as doors, humans or other vehicles. A laser scan of an open door, for example, can look like the end of a corridor, leading to false data association between a map and features. The novelty in this work is that features which belong to the category of dynamic objects are only used as high-level features and are removed from the low-level feature pool. In the work presented here, rgb cameras as well as a laser scanner readings are used to detect doors and to estimate their opening angle. These dynamic features are used as landmarks for self-localization, and the corresponding laser scan readings are ignored by particle weighting. The resulting method is currently a work in progress, and preliminary results are shown. The software developed for this paper is publicly available and integrated into the open-source mobile robot programming toolkit (MRPT).**

I. Introduction

For tackling self-localization within autonomous robotics, a vast set of probabilistic [5], [13] and optimization-based approaches [16] have emerged from current research. Those algorithms rely on external and internal sensors such as cameras, laser scanners and odometry measurement devices for perceiving the environment. During localization, certain assumptions about the world have to be made. The most common assumption is the static world assumption, in which a robot assumes the perceived environment to be static [5], [13], i.e. consisting solely of non-movable objects. In order to be able to deal with dynamic or unexpected objects, the sensor noise model must incorporate unexpected obstacles like P_{unexp} in the Beam-based Proximity Model [13]. The idea behind this work is to remove data corresponding to dynamic objects in the localization algorithm and to alter the noise models accordingly. In order to present a proof of concept, we detect doors with their opening angles. Identifying high-level objects requires a suitable object detection framework, ideally applied during SLAM, in order to construct a map in which the position of all non-movable high-level objects is known. A further benefit to using high-level objects within the localization procedure is better initialization of self-localization. The initial pose of the robot has to be set manually or can be computed automatically by using a large amount of particles, thereby increasing computational complexity [13], [14]. Since this method uses high-level objects, initialization can be improved, as described in Section III.

This paper is organized as follows: Section II discusses related work on the topic of high-level object recognition and integration into self-localization. Section III outlines the approach in depth. Section IV discusses the experiments and a conclusion is presented in section VI.

II. Related Work

Self-localization algorithms can be coarsely classified into laser-scan-based and vision-based works [4], [16]. The latter operate on images, identifying landmarks across images and using bundle adjustment [15] for position estimation. The map-building process is usually performed online (SLAM) and used only to detect loop closures [10]. During pure odometry estimation [6], [8], no map is computed and no loop closures are possible, leading to poorer location accuracy with a detailed evaluation in [16].

Scan-based techniques operate on a two dimensional laser scan, matching the measurements with the known map [13] in order to determine the correct location. As this paper focuses on a more precise treatment of dynamic non-movable objects within a known environment, and laser scans offer limited capability of object detection, it is best suited to improving this class of algorithms, provided the autonomous vehicle is equipped with a camera used for object detection. Vision-based algorithms could, however, also benefit from object detection, especially during the optimization procedure described in [6] and [16], in which a weighted loss function is used to filter outliers after features have been matched. The weighted loss could be replaced by the detection of dynamic movable objects.

Regarding object detection, a vast amount of literature as well as open source software is available. This work focuses on object detection on laser scans as well as within camera images. The current state-of-the-art in image-based object detection uses techniques emerging from deep learning [7], [12]. The authors of [11] present an improvement to [12] that is real time capable, operating at 60-70 frames per second and therefore perfectly suited to a mobile robot scenario. The algorithm identifies a vast set of object classes, including objects in a typical lab environment such as chairs, desks, people and doors, delivering bounding box hierarchies corresponding to detections.

Regarding object detection in the context of mobile robot

*The research leading to these results has received funding from the Austrian Research Promotion Agency (FFG) according to grant agreement 855409 (AutonomousFleet) and 854865 (TransportBuddy).

[1]Felix Koenig and Markus Bader are with the Institute of Computer Engineering, Vienna University of Technology, Treitlstr. 1-3/4. Floor/E191-3, 1040 Vienna, Austria [firstname.lastname]@tuwien.ac.at

self-localization, a significantly smaller set of publications is available. In [1], the authors focus on identifying and modeling doors within a lab environment. The color and shape properties of doors and walls are learned with an expectation maximization (EM) algorithm, which is also used for localization, resulting in a tightly coupled localization and detection. The shape properties of doors are continuously updated during the travel time of the robot. They can subsequently be reused in the same or a different environment.
Current state-of-the-art SLAM algorithms, including object discovery, focus on non-movable objects and introduce them as landmarks, reusing them for loop closure as outlined in [3], [9]. Thrun et al. [17] propose an extension to the Monte Carlo localization framework, tightly integrating movable objects in order to improve localization accuracy. Neither does their approach make use of machine learning for object discovery, nor does it filter out semi-static objects such as doors. Furthermore, the moving object tracking is integrated into the particle filter (PF) used for localization.

III. Approach

The algorithm is presented in two subsections. First, the door detection algorithm is described, and second, the localization algorithm is explained.

A. Door detection and parameter initialization

As mentioned, the object detection of [11] is used in combination with a detector working solely on the laser scan. The idea behind the laser-scan-based segmentation is to identify door hinges and, subsequently, properties of the doors such as their opening angles. The initial procedure, segments the laser scan into contours. The Segmentation is based on the change in depth data between neighboring laser ranges. During experimentation on different datasets with varying thresholds a value of 0.25m has been determined to yield a stable true positive rate across the whole map. Subsequently, a line-fitting algorithm is then performed on each contour segment. If a line segment corresponds to a door as detected by [11], the closest neighboring line segment is determined to be the wall. Finally, the opening angle of the door can be determined as well as the location of the hinge and the length of the door. The landmark-based localization will subsequently be used to correct the parameters.

B. Localization

In this section, the particle filter is outlined based on the Rao-Blackwellized Particle Filter implementation (RBPF) within MRPT [2], [5]. For a more in-depth introduction on PFs, which is out of the scope of this paper, Thrun et al. [13] offer a detailed review of the topic. If k denotes a specific point in time, the robot's states are then written as x_k, with

$$\mathbf{x}_0^k := \{x_0, x_1, \ldots, x_k\}.$$

The states themselves are composed of the position and angular orientation with respect to the world coordinate system.

$$x_k := \begin{pmatrix} x_k \\ y_k \\ \theta_k \end{pmatrix}$$

Additionally, static objects are defined as landmarks l_k, motion sensor measurements as u_k and laser scan measurements as z_k, with the sets defined similarly to in $\mathbf{x}_0^k$.

$$\mathbf{l}_0^k := \{l_0, l_1, \ldots, l_k\}$$
$$\mathbf{u}_0^k := \{u_0, u_1, \ldots, u_k\}$$
$$\mathbf{z}_0^k := \{z_0, z_1, \ldots, z_k\}$$

The PF equation derived models the distribution, capturing the most likely robot states according to their corresponding measurements. It can be decomposed into a *perceptive* and a *predictive* part.

$$p(\mathbf{x}_0^k|\mathbf{z}_0^k, \mathbf{u}_0^k) \propto \overbrace{p(z_k|x_k)}^{\text{perception}} \cdot \underbrace{\int_0^k p(x_k|x_{k-1}, u_k) p(\mathbf{x}_0^{k-1}|\mathbf{z}_0^{k-1}, \mathbf{u}_0^{k-1}) dx_{k-1}}_{\text{prediction}}$$

The predictive element models the likelihood of the actual state with regard to its previous states and measurements, from which the most probable state is recursively inferred.
The perceptive part tries to model the correspondence of the recent state to the current measurement and is referred to as the *beam model* in the literature covering laser scan approaches [13]. This term is of interest within our framework as it also includes the noise in the data, and this is precisely how objects and doors factor into the equation. The beam model is formulated as a mixture of four densities, depicted in Fig. 1, capturing the sensor properties of the laser scanner. All of them incorporate noise in the measurements to a degree. The probability that a laser beam hits known obstacles, for instance walls, is described by p_{hit}. It is defined by a Gaussian distribution corrupted by noise around the true but unknown reading z_k^*. The distributions p_{max} and p_{rand} explain the maximum sensor reading as well as general noise in the data, respectively.

C. Occlusions, Noise and Dynamic Objects

What happens during the algorithm when an unknown obstacle is registered by a beam? In this case, p_{short} aims at capturing this occurrence. The idea is, if multiple dynamic obstacles are present, it is more likely that they will be registered at a short distance from the front of the robot, since they cause occlusions of other obstacles. Therefore, p_{short} is modeled as a Gaussian falloff curve. This term essentially captures the features from dynamic objects as noise in the data. Therefore, the approaches that tightly integrate the dynamic object tracking and estimation into the localization [17], [1], actually treat them at two different levels during localization, which is redundant and erroneous. First, they appear in the sensor noise model, second, despite

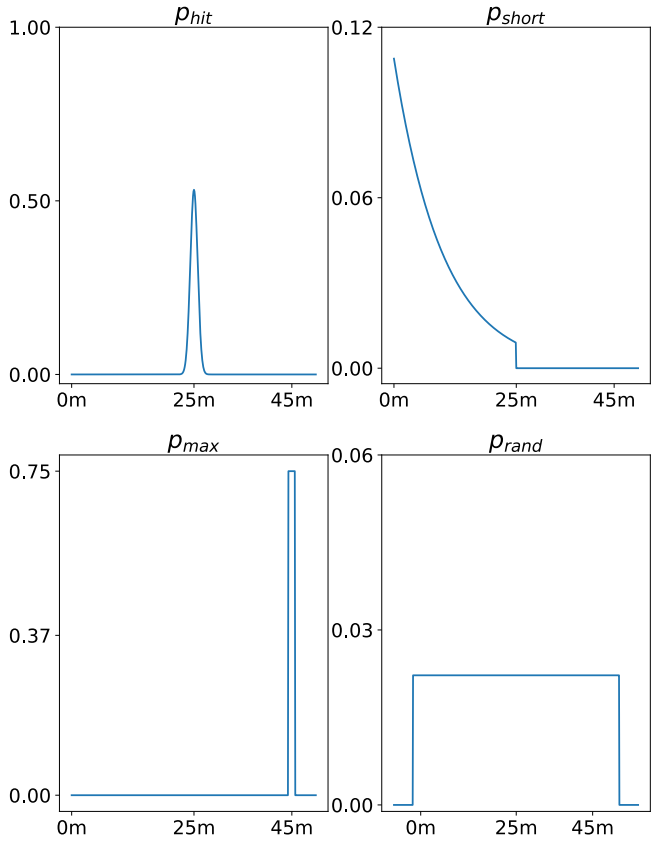


Fig. 1: Sensor model of the laser scan's measurement uncertainty model with $z_k^* = 25$ and $z_{max} = 45$

their parameters are known and estimated, the sensor noise model is not adapted and does not benefit from the gained knowledge. This has not been addressed in the literature, at least to the best of our knowledge. The successive section in this work studies the effects that a complete removal of dynamic objects has on the sensor readings and discusses possible implications.

IV. Experiments

A. Setup

This section describes the preliminary results of the above presented approach. At the moment, the door detector is employed and doors are removed from the scan. The sensor readings are then analyzed and a conclusion about the noise model is made. The following two experiments have been conducted:

1) Driving along an office hallway with opened doors.
2) Standing in front of an open door.

In each experiment a map of the environment is present and used for localization. Doors have either not been modeled within the map or they are modeled as being closed. Fig. 2 depicts a robot travelling in the office environment. The traversable region is black and walls are colored in gray. The map has a resolution of 3737×2338 with each pixel representing a 1cm^2 grid, limiting the accuracy of our evaluation to this range. The evaluation has been performed via ray-tracing. Each laser beam, as shown in red in Fig. 2, is traversed until it hits the first pixel on the wall. The pixel coordinates are subsequently converted from the world into the laser reference frame and its euclidean norm represents the expected distance measurement $z_{i,k}^*$ for the i^{th} beam at time k. The measured laser range value is denoted as $z_{i,k}$. Within the Figure, expected measurements are colored in green, while observed measurements are colored in blue.

In order to transform points into the reference frame of the laser and successively establish a relationship between the world and robot coordinate system the position of the robot is needed. Localization errors happen during abrupt movements of the robot and are not entirely avoidable. In this particular case, they will factor into the sensor noise model together with the beams hitting dynamic obstacles. This can be observed in Fig. 2 where the expected laser ranges do not precisely coincide with the observed ones on the northern wall. However, it is shown in the next section that this does not affect the overall quality of our comparison between the measurements with and without dynamic obstacles. In order to provide an example without localization inaccuracies, a second experiment has been conducted with the robot standing still in front of the door. During the first experiment, a total of 7 doors have been opened in a $45-120°$ angle (being closed is equal to angle 0) along the hallway and the robot is travelling from right to left (see Fig.2). In the second experiment, the robot is standing in front of a door and is not moving.

B. Evaluation

The two experiments have been evaluated as follows. First, the robot is driving along the hallway and records all the laser measurements into a ROS-bag file. During the travel time the localization runs in parallel so that the correct frame transformations can be associated to the measurements. Second, the data is then processed offline both with and without door detection.

The space of all possible laser ranges is represented as a two-dimensional histogram h where each cell has the dimensions of 25×25cm with the entry $h(0,0)$ denoting the robots origin. For each pair of expected measurements $z_{i,k}$ and observations $z_{i,k}^*$, the cell of $h(id(z_{i,k}), id(z_{i,k}^*))$ is incremented by 1, resulting in a correlation matrix. Here, the function $id()$ determines the correct grid cell for each measurement. Finally the values are normalized along each column that represents all observations for a single expectation:

$$\forall l,i : h(l,i) = \frac{h(l,i)}{\sum_j h(j,i)}$$

V. Results

Fig. 3 shows the matrix h for the scenario in experiment 1. The correct observations are present on the diagonal and, for the purpose of visualization, have been clipped at 0.25%. The left image depicts the matrix h_u obtained from equation IV-B using the raw laser scan whereas the right image shows the

Fig. 2: Map of the office environment used for conducting the experiments together with a laser scan at time k.

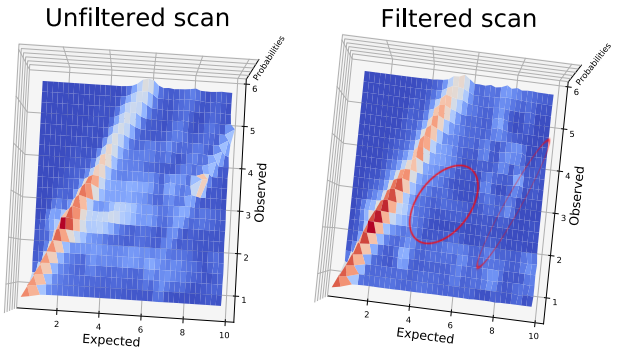


Fig. 3: Plot of h_u and h_f from the first experiment. The red ellipses show the region affected by removing outliers corresponding to doors.

resulting histogram h_f with filtered doors. It is apparent that the line parallel to the diagonal of h_u is not present in h_f and is caused by travelling towards a door opened directly in front of an expected wall. Therefore, the observations are shifted by the distance between the door and the wall. Furthermore, it is also visible that the entries along the diagonal on the right are higher than those on the left, especially along entries where the door has been filtered in $h_f(4\ldots6,4\ldots6)$.
The noise cluster within $h_u(2\ldots4,2\ldots8)$ has been filtered accordingly. Since we drove close to the doors and then around them, they are observable in the vicinity of the robot. Fig. 4 includes the resulting histograms of the second experiment where the robot is standing in front of the door.

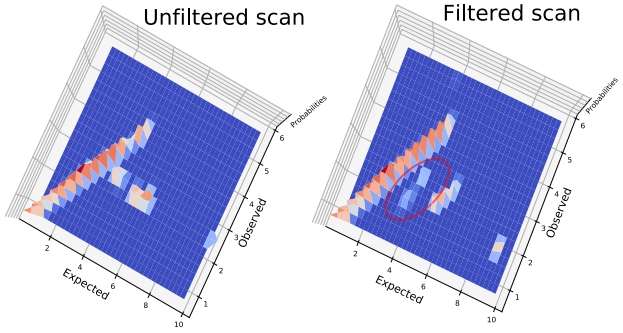


Fig. 4: Plot of h_u and h_f from the second experiment. The red ellipse shows the region affected by removing outliers corresponding to doors.

The filtered result is still noisy primarily due to outliers that do not belong to the door but have not been filtered, leading to the conclusion that the filtering is more beneficial in the moving robot scenario of the first experiment. Both experiments show that sensor measurements observed from dynamic objects manifest themselves in the sensor noise model in the range $[0,z_k^*]$. It is therefore necessary to make different assumptions of the model and potentially remove or redesign the function p_{short}. Future work will aim at addressing this issue.

VI. Conclusion

This work shows the importance of treating object tracking and detection at the level of the sensor noise model. Furthermore, an approach is outlined describing a basic door detector in conjunction with a localization procedure. Precise treatment of obstacles within the sensor noise model and successive integration into the localization procedure remains the topic of future research.

References

[1] D. Anguelov, D. Koller, E. Parker, and S. Thrun, "Detecting and modeling doors with mobile robots," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 4, April 2004, pp. 3777–3784 Vol.4.

[2] J.-L. Blanco-Claraco, "MRPT," https://www.mrpt.org/, 2019, [Online; accessed 14-March-2019].

[3] S. Choudhary, A. J. B. Trevor, H. I. Christensen, and F. Dellaert, "Slam with object discovery, modeling and mapping," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sep. 2014, pp. 1018–1025.

[4] M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on robotics and automation*, vol. 17, no. 3, pp. 229–241, 2001.

[5] A. Doucet, N. d. Freitas, K. P. Murphy, and S. J. Russell, "Rao-blackwellised particle filtering for dynamic bayesian networks," in *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, ser. UAI '00. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000, pp. 176–183. [Online]. Available: http://dl.acm.org/citation.cfm?id=647234.720075

[6] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Mar. 2018.

[7] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, cited By 7777.

[8] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015, cited By 241.

[9] J. McCormac, R. Clark, M. Bloesch, A. J. Davison, and S. Leutenegger, "Fusion++: Volumetric object-level SLAM," *CoRR*, vol. abs/1808.08378, 2018. [Online]. Available: http://arxiv.org/abs/1808.08378

[10] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018, cited By 11.

[11] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," *CoRR*, vol. abs/1612.08242, 2016. [Online]. Available: http://arxiv.org/abs/1612.08242

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 91–99. [Online]. Available: http://papers.nips.cc/paper/5638-faster-r-cnn-towards-real-time-object-detection-with-region-proposal-networks.pdf

[13] S. Thrun, W. Burgard, D. Fox, and R. Arkin, *Probabilistic Robotics*, ser. Intelligent robotics and autonomous agents. MIT Press, 2005. [Online]. Available: https://books.google.at/books?id=2Zn6AQAAQBAJ

[14] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Monte carlo localization for mobile robots," in *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 1999.

[15] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment – a modern synthesis," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1883, pp. 298–372, 2000.

[16] V. Usenko, J. Engel, J. Stueckler, and D. Cremers, "Direct visual-inertial odometry with stereo cameras," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.

[17] C.-C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," *International Journal of Robotics Research*, vol. 26, no. 9, pp. 889–916, 2007.

Safety of Mobile Robot Systems in Industrial Applications

Alexandra Markis[1], Maximilian Papa[2], David Kaselautzke[2], Michael Rathmair[3], Vinzenz Sattinger[2], and Mathias Brandstötter[3]

***Abstract*— The fourth industrial revolution is in full swing and according to "BCC Research" a compound annual growth rate of 23.1 % will be expected on the global market for the period of 2018 to 2023. Leading new technologies as mobile robotics and manipulator systems will facilitate more flexible and efficient production processes. Unfortunately, mentioned in the latest "Statista" report, the complexity of mobile robotic systems and missing standards are one of the major obstacles for a broad rollout of mobile robot systems. This paper presents a selection of what is already possible in the field of mobile robots and mobile manipulation systems and gives an outlook on current and upcoming leading edge developments. We focus on the requirements of the industry and addresses the related barriers concerning the design and implementation of safe applications. As a result, we propose best practice, recommendations and first concepts to overcome the discussed challenges in implementation.**

I. INTRODUCTION

Mobile robot systems have become more and more embedded in modern industrial production systems in recent years. In Austria 99.7 % of all companies are small and medium-sized enterprises (SMEs), however they only account for 64 % of the net sales revenue shares [12]. Most of them could not afford the technologies needed for the fourth industrial revolution (Industry 4.0) according to their limited resources. This difference between SMEs and large enterprises is noticeable in their internal warehouse logistics and production lines. Mobile robots as used by large enterprises increase the efficiency and flexibility in logistic, assembly and production processes, whereby human resources will be relieved from monotonous work [3]. When talking about mobile robots different terms are used. The most common ones are autonomous mobile robots (AMRs) and automated guided vehicles (AGVs) and up to now, there is no absolutely clear distinction established. In general, AGVs are, per definition, in-house, floor-bound conveyor systems with automatically controlled vehicles whose primary task is material handling. The guidance of the vehicle is realized by some infrastructure hardware (wires, black lines, etc.) placed in the surrounding environment of the moving robot. In contrast, an AMR can move around autonomously and performs a specific task (e.g., household vacuum cleaner robot). The navigation through its environment is given by sensors mounted on the robot [5]. Mobile robots are much more flexible in their use and allow features as dynamic customer specific modifications, situation aware movement planning, enhanced collaborative operation, etc. Thus, they are highly relevant for state of the art Industry 4.0 applications. However, using mobile robotics in industrial applications results in increased safety requirements and a well organized co-existence between moving machines and humans.

The remainder of this paper is organized in the following manner. First, we give some background information about existing types of mobile robots and how they are already used in industry. In Section III, we present existing safety-standards for mobile robot systems and discuss associated problems. Based on that, we match these safety problems to challenges occurring by integrating mobile robots into real-world applications. Finally, in Sections V and VI we generalize our results, propose some recommendations and concept for successful implementation and conclude the paper by a summary of lessons learned so far.

II. TYPES OF INDUSTRIAL MOBILE ROBOT SYSTEMS

As already introduced above there are different types of mobile robots. In this section, first we briefly describe two selected important components of mobile robot systems with respect to safety and second focus into two industrial-important types of mobile robots (wheeled robots and mobile manipulators) in more detail.

A. General Safety-Relevant System Components

A robot system consists of a large number of individual components, which in combination provide a high degree of flexibility and application variety. These include, among others, the human-machine interface, task allocation and scheduling, sensors, sensor fusion and perception, motion planning, actuators and robot control. We would like to take a brief look at the most safety-relevant components present in every mobile robot system.

a) Sensors: In the case of sensors that are mounted on a robot, safety instrumented systems must be separated from non-critical information sources. Safety-related sensors have high relevance for the overall safety of the system and must

*This work was supported by the Austrian Research Promotion Agency (FFG) within the project DR.KORS – Dynamic reconfigurability of collaborative robot systems (number 864892) and by Vienna City Administration (MA23) within the project SIP 4.0 – Sicherheit in intelligenten Produktionsumgebungen.

[1]Alexandra Markis is with TÜV AUSTRIA HOLDING AG, 2345 Brunn am Gebirge, Austria alexandra.markis@tuv.at

[2]Vinzenz Sattinger, Maximilian Papa and David Kaselautzke are with Fachhochschule Technikum Wien, Höchstädtplatz 6, 1200 Wien, Austria vinzenz.sattinger@technikum-wien.at

[3]Michael Rathmair and Mathias Brandstötter are with ROBOTICS, Institute for Robotics and Mechatronics, JOANNEUM RESEARCH Forschungsgesellschaft mbH, 9020 Klagenfurt am Wörthersee, Austria mathias.brandstoetter@joanneum.at

be components certified by standards (e.g., EN 61508) that guarantee a certain safety level.

b) Robust Navigation with Sensor Fusion: Robust navigation of mobile robots is necessary, especially if it does not move in a static manner steered by fixed tracks. The localization of the mobile robot, the mapping of the environment as well as the path planning from one station to another are important areas of the robust navigation [10]. Problems with sensor failures or inaccurate results can lead to misinterpretations of the current environmental situation or robot status, resulting in potential safety risks. Sensor fusion is a proven solution for this problem, as information of different types of sensors is used. Strictly speaking, a sensed value is validated by at least a second sensor using a different measuring method. This approach enables, that the failure of one sensor can be compensated by a set of other sensors built in. For example, a slipping wheel will not be detected by two or more odometry sensors but in combination with an accelerometer and gyro sensor or a global sensor such as Indoor-GPS [23].

c) Robot Control: The robot operating system (ROS) is an open source software framework, which enables an easy implementation of robot application [17] and control of a robot system. Most of the mobile robot systems today are working with ROS, because the framework delivers many useful packages for various different problems. Therefore, a problem does not have to be solved always from zero knowledge. Different sensors and actuators are preprogrammed in this framework, ready to use. Different localization and navigation methods for mobile robots were programmed with ROS hence, see [7], [1].

A major advantage of using ROS with mobile manipulators in terms of safety is that the two main hardware components (mobile base and robot arm) are controlled in one software framework. In case a safety-relevant situation occurs in one of these two components, it can be transferred directly and adequately to the other component.

B. Wheeled Mobile Robots

As already mentioned in the introduction we differentiate between automated guided vehicles (AGVs) and autonomous mobile robots (AMRs). AGVs are working with deterministic methods, which deliver a simple binary result. A line tracking sensor for example will decide whether it is on the line or not. However, fixed tracks are required for the navigation of such AGVs [20], [22], [6]. Concerning this guidance of AGVs there are three different types:

- Wire guidance uses wires embedded in the plant floor that are sensed inductively by the vehicle in order to determine its lateral position.
- Inertial guidance uses transponders, embedded in the floor, for verifying the course of the AGV and gyroscopes to identify and correct the inevitable drift of the system. Magnets can be placed in strategic locations for the AGV to read and to reset the system.
- Laser guidance uses a laser transmitter-receiver carried by the AGV. It senses retroreflective landmarks strategically placed throughout the plant. By sensing the landmarks, the vehicle can then triangulate its position.

Fig. 1. Mobile manipulator (compilation of an UR10 and MiR100) solving an assembly task in a prototype industrial application

On the other hand, researchers are working mostly on AMRs using probabilistic methods. They suit better for real world problems such as localization in dynamic environments, obstacle detection and collisions free navigation in real time according to the detected obstacles. Furthermore, AMRs do not need any kind of mechanical installation such as rails or guiding tracks, allowing more flexible movements of the mobile robots [9], [14], [16], [21].

C. Mobile Manipulation

A mobile manipulator as illustrated in Fig. 1 can be seen as a system that combines a mobile robot and a serial manipulator equipped with sensors and at least one actuator. The actuator usually corresponds to a gripper system in order to manipulate objects in the necessary manner. The combination of a mobile and a serial robot unites the advantages of both robot systems and has the potential to be used flexibly. Numerous handling tasks can be performed with such a system, regardless of a predefined location. Depending on the field of application or the task assigned to them, such robot systems are operated remotely, semi-automatically or completely autonomously.

In addition to remote-controlled exploration and rescue robots, which this article does not want to focus on, sensitive mobile manipulators are also used as service robots. Service tasks can be fulfilled in the public and private sectors. Examples of applicable platforms are Tiago from PAL Robotics [18] and Care-O-bot 4 from Fraunhofer IPA [11]. For the economical operation of a mobile manipulator in a production environment, the design differs from that of a service robot platform. Robustness, payload, adaptability, programmability are some of the important features and functions that such a system must provide. The implementation of these requirements is still the subject of research by numerous institutions and consortia. Within the EU project STAMINA [13], the company is working intensively on the

collection and delivery of assembly-related components in the production halls of the car manufacturer Groupe PSA (formally PSA Peugeot Citroën). The intralogistics transport tasks are supported by a fleet of mobile manipulators.

Although a number of research and service platforms of this type of robot systems are available on the market, the variety of industrial solutions is limited. Examples: KUKA KMR, Stäubli HelMo; Compilation examples: OTTO 1500 and Yaskawa SDA20D, MiR100 and UR10. Research and development projects are currently aiming to make mobile manipulators usable for the picking of heavy or bulky components. By using mobile manipulators for this kind of tasks failures, which can cause high follow-up costs, should be prevented. In addition, human workers would be freed from long-term, highly repetitive and therefore unhealthy tasks, which becomes more and more relevant as the average age of the workforce increases further. The use of sensitive mobile manipulators in manufacturing environments (assembly, production and logistics) is currently reaching the limits of safety engineering. Especially when such a mobile robot system has to perform non-deterministic pick and place tasks [19].

III. SAFETY STANDARDS FOR MOBILE ROBOT SYSTEMS

Before discussing selected standards for AGVs, AMRs and mobile manipulators in detail, we present a general overview of relevant documents for the standardization of mobile robotic applications. Fig. 2 illustrates selected laws, ISO and IEC standards as well as guidelines. At the head of all documents and thus most important are laws as the Machinery Directive and national laws for the protection of human safety. The main purpose of the Machinery Directive is to ensure an equal safety level for machines placed on the market or put into service in all European member states and to enable freedom of movement within the European Union. The second stage of the illustrated hierarchy contains international ISO and IEC standards. ISO Standards are categorized in A, B and C types, whereas type A are basic safety standards including general aspects and design principles, type B are generic safety standards covering aspects applicable for a wide range of machinery, and type C are safety standards dealing with requirements for particular machines. Standards highlighted by a green check mark are harmonized with the overriding importance Machine Directive. This means, using such a harmonized standard a developed product/application comply with the relevant legislation. Last, there are recommended guidelines and technical specifications (TS) which reflect the field's technological state of the art but are not obligatory for any standardization.

A. Standards for Wheeled Mobile Robots

AGVs operate on general in-house traffic routes while the demands on their safety are correspondingly high. The regulations for the usage of AGVs is nationally and at EU level mostly clearly regulated by the Machinery Directive 2006/42/EC. It obligates the manufacturer of a machine or plant to perform a risk assessment (according to EN ISO 12100:2010) and, if necessary, to take measures to decrease the risks for the operator [8]. Besides the machine directive, national and international standards are in place. The EN 1525:1997 (*Safety of industrial trucks - Driverless trucks and their systems*) is the main standard concerning AGVs. The present problem is that this standard no longer addresses the current state of the art technology (release date 1997) and is also not harmonized with the Machinery Directive. EN ISO 3691-4:2018-05 is currently in development and should replace EN 1525:1997. This means that manufacturers are currently obliged to fulfill the requirements of the Machinery Directive without a corresponding state of the art standard.

Similar to AGVs, a comprehensive applicable standard for autonomous mobile robots (AMRs) is still missing and the promised standard EN ISO 3691-4:2018-05 is still in development. Until this standard is published, there exists only one comparable standard called ISO 13482:2014-02 (*Robots and robotic devices - Safety requirements for personal care robots*) concerning personal care robots, which is not applicable to industrial applications.

B. Standards for Mobile Manipulators

For mobile manipulators, which are a combination of a movable platform and a robotic arm, different standards have to be applied, depending on the purpose of the robotic arm:

1) *Robot arm rests, when AGV is in motion:* The robot arm can be considered as a load and the risk assessment for the total hazard by analogy to regular AGV standards has to be assessed. A critical aspect is the clear definition of a safe position in which the robot will be transported. For the interfaces on which the report performs its task, robotic standards have to be considered.
2) *Robot arm does assembly or processing steps during motion:* All relevant (collaborative) robotic standards must be applied for the robot arm when performing the necessary risk assessment.

The corresponding standards are given by ISO TS 15066:2016 (*Robots and robotic devices - Collaborative robots*) and ISO 10218-1:2011 (*Robots and robotic devices - Safety requirements for industrial robots*) The core standard EN ISO 10218-1 is currently in revision as it no longer represents state of the art technology. The whole application, meaning the moving platform, robot arm and the surrounding environment (roads and work areas), must then be evaluated according to part 2 of the EN ISO 10218 standard, which is dealing with the integration of industrial robot applications.

There are currently no fully-compliant standards, guidelines or design proposals for this type of robot, so the potential legal consequences of injury or damage are difficult to predict. When interacting with human workers, completely new hazards arise that are currently not covered in the present standards. To avoid a severe limitation of the extent and flexibility of enforceable solutions compared to the already

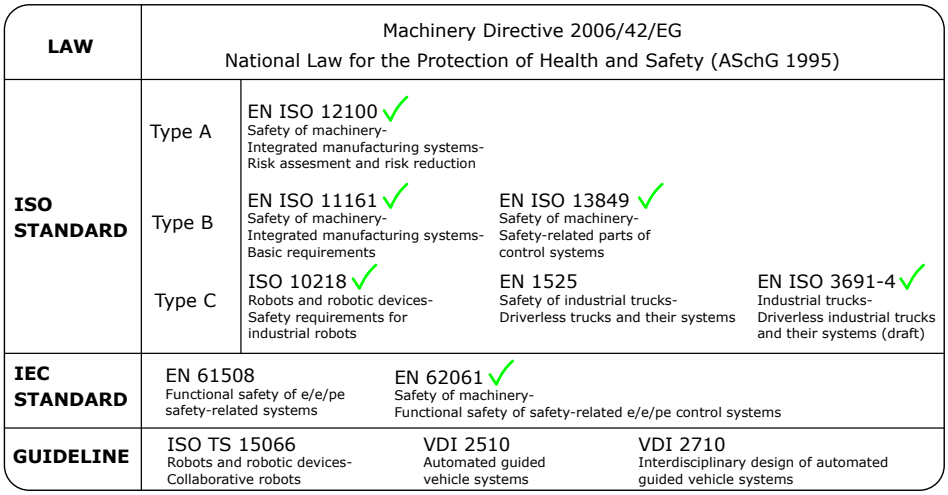


Fig. 2. Overview over relevant laws, standards and guidelines for mobile robot systems (automated guided vehicles and mobile manipulators). Standards highlighted by a green tick are harmonized with the machinery directive.

technically feasible ones, new approaches and safety models are highly needed.

In the USA the Robotic Industries Association (RIA) in cooperation with the American National Standards Institute (ANSI) are working on a conjunct national standard for mobile robotic systems called ANSI/RIA R15.08 (Draft). This proposed standard tries to bridge any gaps between regulations for AGVs, AMRs and mobile manipulators. ANSI/RIA R15.08 is in draft state since 2017 and is announced to be published in the early 2019 [2].

IV. CHALLENGES AND OPTIONS IN PRACTICAL APPLICATION DOMAINS

Implementing any kind of mobile robot or mobile manipulator for practical applications will lead to different problems. We discuss selected challenges within this section considering the present situation of missing applicable standards.

A. Challenges Associated with Wheeled Mobile Robots

A common drawback for authorities implementing a mobile robot application is the lack of know-how in the form of missing standards and guidelines. Talks with industrial partners have shown that most of the SMEs do not know which robot they need, and which criteria are important to differentiate between available products on the market.

A significant structural challenge is a required safety distance as required by the standard. For example, a minimum distance of 500 mm besides the robot track has to remain free to enable potential escape possibilities for humans. Industrial experiences showed that this minimum safety-distance is difficult to reach, especially when AGVs are integrated into existing systems where space requirements are not planned for their usage and paths are not intended to be utilized for mixed man-machine usage.

Additionally, the standard requires a limitation of velocities (0.3 m/s in linear/driving direction and 0.7 m/s in angular direction). This strict velocity constraints lead to the (erroneous) assumption that safe and certified components could be omitted by strictly limiting the velocities to the mentioned borders or that a risk assessment in consideration of the whole work system and the interfering inner-factory traffic is not necessary. On the contrary, a risk assessment might even help to allow higher velocities in some cases to increase the overall productivity of the implemented application.

Furthermore, employees in production environments are afraid of getting replaced by mobile robots (however, this is not the only fear). A survey resulted that the probabilistic navigation and autonomous obstacle avoidance with dynamic plan reconfiguration at high speed leads to a skeptical attitude of the workers. Deterministic navigation, on the other hand, results to a less scary feeling, because employees know exactly where the mobile robot is moving and do not have to be afraid of any unexpected movement behavior. Slower and ”relaxed” movements of the mobile robot resulted also positive in this survey. However, the production time will increase according to the slower driving mobile robots. Therefore, a compromise including the production time and the employees’ opinion has to be made [15].

B. Challenges Associated with Mobile Manipulators

The practical application opportunities for the use of sensitive mobile manipulators are extensive. Handling tasks of all kinds dominate, especially in the industrial context. The tasks in the production areas are usually assigned to intralogistics or internal goods handling. Transporting crates, removing and inserting components from and into machines or shelves, simple positioning tasks for work preparation, are the main assignments. Sensitive mobile manipulators are basically capable of carrying out these activities during normal operation with humans in the transport area. Nevertheless, the currently customary transport speeds of around 4 km/h do not compete with those of, e.g., motorized industrial trucks. In so-called ghost shifts, undisrupted progress of work is possible and is less critical in terms of cycle time. In addition, risks for humans are largely reduced during this period. However, if the above operations are to be performed

during normal operation, the safety assessment of the robot system comes to the fore again.

V. RECOMMENDATIONS AND CONCEPTS TO OVERCOME THESE CHALLENGES

Many of the previously mentioned problems were solved already by different companies successfully implementing mobile robots in industrial environments. Therefore, recommendations are mentioned in the following as lessons learned and a concept to overcome the safety issue for mobile manipulation are introduced.

A. Recommendations for Wheeled Mobile Robots

The UAS Technikum Vienna is currently working on a research project named "SIP 4.0 - Sicherheit in intelligenten Produktionsumgebungen" funded by Vienna City Administration. One output of this project will be a best practice guiding document, which should enable a low effort implementation of mobile robots in intelligent production systems, considering the small budget and know-how of SMEs. Most important, every robot application has to pass a risk assessment and has to prove that potential hazards are tolerable. As defined in ISO 12100:2010 a group of several people have to line up different risks in touch with the robotic system demarcated to the predefined system boundaries.

Environment: The perfect mobile robot for an enterprise depends on the needs and cannot be generalized. Most of the enterprises do already have a given infrastructure making it difficult to integrate mobile robots. Regardless, if an infrastructure is already given or a new one has to be built, it is important to include safety experts and labour inspectors from beginning to avoid problems in the final implementation state. In addition, different non-safety-certified components could be considered to provide a more comfortable feeling for the employees with mobile robots. Furthermore, rules for human-machine-collaboration with defined routes for mobile robots, traffic light systems at intersections and priority rules can accomplish this. All mentioned measures are applicable regardless of the implemented navigation method.

Sensor systems: Additional sensors for detecting humans are highly recommended. As seen at the LogiMAT 2019, all manufacturer of AMRs had safety-certified laser scanners implemented for detecting hindrances/humans in their products, independent of any velocity constraints given in the standards. Even most of the AGVs at the LogiMAT had safety laser scanners installed, apart of AGVs working in human free areas. In addition to human safety, the safety of mobile robots itself should also be considered especially for industrial environments. For instance, a mobile robot may destroy itself at low passages or dropping objects leading to an economical damage.

Tradeoff between AGVs and AMRs: An established technology might be a better option even if a newer technology is available. As seen at the LogiMAT the ratio between AGVs compared to AMRs was still 50:50. Probabilistic robotics used in AMRs is often preferred because of its flexibility. However, the paths of the mobile robot will not change daily in industrial environments and employees do not completely trust the automatically dodging mobile robots.

B. Concept to Overcome the Safety Issue for Mobile Manipulation

Especially in collaborative robotic applications, it must be ensured that forces and pressures remain below defined limits before being taking them into operation. However, the robot system or plant must not be modified without restarting this process. This circumstance poses a major challenge for applications with mobile manipulators because the robot system operates in a dynamic and therefore mostly non-deterministic environment.

We propose an alternative approach to partially solve this practically highly relevant problem. Not only a specific robot application is to be approved in terms of safety, but also a kind of class of an application. This means that, for example, a pick location is not considered explicitly, but a space as generous as possible in which an object can be picked up safely. The challenge is (a) to define reasonable limits of the space and (b) to demonstrate that all picking points within the defined limits are safe. This space is a so-called modification dimension, in which the application can be modified without the need for a new risk assessment. Other modification dimensions can be, for instance, a geometrical dimension of the workpiece, the position of the robot base or the manipulation velocity.

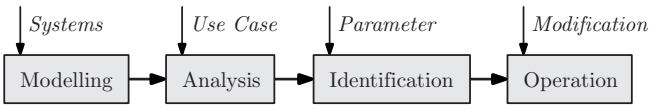


Fig. 3. Simplified structure of the proposed approach; the main action blocks and their input information.

To enable a permitted modification on one or more dimensions some steps are required to show that safety guaranteed. Fig. 3 shows a simplified structure of the proposed task sequence that can lead to this goal. Based on a set of mathematical models of different robots and safety devices (*Systems*) the actual application is analyzed and modeled (*Use Case*). An important next step is the identification of specific parameters that characterize the application and the system in use (*Parameter*). The decisive step is the merging of all system and application describing models into one overall model [4]. It is able to assess whether a variation within a modification dimension is actually admissible during operation (*Modification*). This is achieved by (a) assigning limits to a modification dimension, (b) performing selected biofidelic measurements at the edge and within the range, and (c) demonstrating through the mathematical models that all points within the range can be declared safe. With this

method, the practical application flexibility of collaborative robots can be significantly increased. In its full extension stage, to be counted as future work, safe mobile manipulation can thus be achieved.

VI. CONCLUSIONS AND OUTLOOK

Each company has to choose between an AGV or AMR implementation. This depends on their personal needs such as production time, transportation tasks and the number of employees working in the expected mobile robot environment. Depending on this choice, the complexity of the implementation and the potential problems vary. No matter which decision is made the usage of additional safety measures for human safety like sensors are essential. Organizational measures such as taped lines for demarcations and traffic light systems at intersections for a more comfortable feeling with the mobile robots can have a major impact on the acceptance by the employees, which in the end is crucial for a successful application. Applicable standards are currently missing or in revision and the industry is in high need for technical support and new models in this regard so that new technologies can be still be used and applied safely without harming productivity. Apart from that, another major constraint using mobile robots and manipulators is the lack of flexibility when adjusting or changing workplaces or tasks. The aim of up and running research and development projects is to build up (software-) systems to (semi-) automatize the evaluating scenarios concerning their hazards. Still, it is not certain that a system like that can actually guarantee that all necessary safety aspects for the relevant modification dimensions are identified and addressed. To be a relevant support system this software would also have to evaluate the external and internal safety equipment and models accordingly and as an overall system, with all existing overlaps and interconnections. Technical feasibility is still uncertain.

In the need for quicker adoption to new requirements and more flexibility, safety topics and issues will become highly relevant during the planning and design phase. Therefore, methods and tools have to be created to enable virtual enhanced testing methods for flexible work-systems. In the future, sensor systems will become more advanced and cheaper. This will make it possible to equip the facility in a bigger manner and to bring the safety from the device itself into the surrounding space. In addition, Machine Learning algorithms for an autonomous configuration of machine new evaluation, testing and verification methods will have to be developed and applied. Summarized, the future will not bring a ”cookbook standard” that addresses all issues and needs. The standards that are currently in development show a trend towards more individual responsibility of the user in regards to the (still) required risk assessment.

REFERENCES

[1] H. A. Ahmed and J.-W. Jang, “Design of cloud based indoor autonomous navigation with turtlebot3,” *International Conference on Future Information & Communication Engineering*, vol. 10, no. 1, pp. 118–122, 2018.

[2] R.-R. I. Association. Industrial mobile robot safety standards on the forefront. [Online]. Available: https://www.robotics.org/content-detail.cfm/Industrial-Robotics-Industry-Insights/Industrial-Mobile-Robot-Safety-Standards-on-the-Forefront/content_id/6710

[3] Automations Praxis. (2017) Mobile Robotik löst langwierigen Transport (german). [Online]. Available: https://automationspraxis.industrie.de/servicerobotik/mobile-robotik-loest-langwierigen-transport/

[4] M. Brandstötter, T. Komenda, F. Ranz, P. Wedenig, H. Gattringer, L. Kaiser, G. Breitenhuber, A. Schlotzhauer, A. Müller, and M. Hofbaur, “Versatile Collaborative Robot Applications through Safety-rated Modification Limits,” in *29th International Conference on Robotics in Alpe-Adria-Danube Region*, Kaiserslautern, Germany, 2019, *Accepted for publication*.

[5] Cross robotics and machine automation. The Difference Between AGVs and Mobile Robots. [Online]. Available: www.crossco.com/blog/difference-between-agvs-and-mobile-robots

[6] S. K. Das and M. K. Pasan, “Design and methodology of automated guided vehicle-a review,” 2016.

[7] X. Fang, X. Fu, and M. Sun, “The improved locating algorithm of particle filter based on ros robot,” *IOP Conference Series: Materials Science and Engineering*, vol. 322, no. 5, p. 052034, 2018.

[8] Federal Ministry for Digital and Economic Affairs, “Machinery directive,” 2010.

[9] S. Guo, T.-T. Fang, T. Song, F.-F. Xi, and B.-G. Wei, “Tracking and localization for omni-directional mobile industrial robot using reflectors,” *Advances in Manufacturing*, vol. 6, no. 1, pp. 118–125, 2018.

[10] J.-S. Gutmann, *Robuste Navigation autonomer mobiler Systeme (german)*. Aka, 2000.

[11] R. Kittmann, T. Fröhlich, J. Schäfer, U. Reiser, F. Weißhardt, and A. Haug, “Let me introduce myself: I am care-o-bot 4, a gentleman robot,” *Mensch und computer 2015–proceedings*, 2015.

[12] KMU Forschung Austria. (2016) KMU-DATEN (german). [Online]. Available: https://www.kmuforschung.ac.at/zahlen-fakten/kmu-daten//

[13] V. Krueger, A. Chazoule, M. Crosby, A. Lasnier, M. R. Pedersen, F. Rovida, L. Nalpantidis, R. Petrick, C. Toscano, and G. Veiga, “A vertical and cyber–physical integration of cognitive robots in manufacturing,” *Proceedings of the IEEE*, vol. 104, no. 5, pp. 1114–1127, 2016.

[14] H. Li and A. V. Savkin, “An algorithm for safe navigation of mobile robots by a sensor network in dynamic cluttered industrial environments,” *Robotics and Computer-Integrated Manufacturing*, vol. 54, pp. 65 – 82, 2018.

[15] Y. Liang and S. A. Lee, “Fear of autonomous robots and artificial intelligence: Evidence from national representative data with probability sampling,” *International Journal of Social Robotics*, vol. 9, no. 3, pp. 379–384, 2017.

[16] I. Nielsen, Q.-V. Dang, G. Bocewicz, and Z. Banaszak, “A methodology for implementation of mobile robot in adaptive manufacturing environments,” *Journal of Intelligent Manufacturing*, vol. 28, no. 5, pp. 1171–1188, 2017.

[17] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “Ros: an open-source robot operating system,” in *ICRA Workshop on Open Source Software*, 2009.

[18] M. A. Roa, D. Berenson, and W. Huang, “Mobile manipulation: toward smart manufacturing [tc spotlight],” *IEEE Robotics & Automation Magazine*, vol. 22, no. 4, pp. 14–15, 2015.

[19] J. Saenz, C. Vogel, F. Penzlin, and N. Elkmann, “Safeguarding collaborative mobile manipulators - evaluation of the VALERI workspace monitoring system,” *Procedia Manufacturing*, vol. 11, pp. 47–54, 2017.

[20] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*. Scituate, MA, USA: Bradford Company, 2004.

[21] C. Sprunk, B. Lau, P. Pfaff, and W. Burgard, “An accurate and efficient navigation system for omnidirectional robots in industrial environments,” *Autonomous Robots*, vol. 41, no. 2, pp. 473–493, 2017.

[22] G. Ullrich, *Fahrerlose Transportsysteme: Eine Fibel - mit Praxisanwendungen - zur Technik - für die Planung (german)*, ser. Fortschritte der Robotik. Vieweg+Teubner Verlag, 2011.

[23] X. Yun, J. Causdian, and M. Audette, “Autonomous operations of mobile robots in a full range of environments,” 2018.

Independent Offset Wheel Steering: An Improved Approach to Wheeled Mobile Robot Locomotion

Firat Berk Cakar[1], Eugen Kaltenegger[1], George Todoran[1] and Markus Bader[1]

***Abstract*— In this paper, a new wheel configuration for mobile robot locomotion called IWOS (Independent Wheel Offset Steering) is presented. This approach offers quasi-omnidirectionality, collision detection and mitigation, expressive navigation capabilities with a simple mechanical design. First, an overall study of popular wheel designs and configurations is provided and then a detailed explanation of IWOS as well as it's distinct advantages are given. A proof of concept is shown using the physics simulation (GazeboSim) simulating various scenarios.**

I. INTRODUCTION

During moving in a crowd or between people, collisions are inevitable. Currently, this problem is bypassed in mobile robotics by maintaining a sufficient safety distance to people and objects. This results in a very limited work-space for mobile robots or too long waiting times. The solution is a drive concept that mitigates collisions and dissipates the crash energy. In addition, a vehicle that can be pushed away and can also push people would be desirable. Such a technology, which is called compliance, is already used for robot arms.

Legged locomotion could solve this problem but has many disadvantages such as the mechanical complexity, power and energy efficiency. Therefore, wheel-based designs are preferred as they offer stability, maneuverability and are mostly easy to control [4][9].

A new wheel configuration, Independent Offset Wheel Steering (IWOS) shown in Fig.1, will be presented in this paper. Different wheel designs are being utilized in the field of wheeled locomotion, namely the standard wheel, castor wheel, Swedish wheel and spherical wheel. Each of them come with their particular strengths and weaknesses [4][9]. IWOS consists of two actuated wheels each mounted on a steerable (compliance controlled) arm and a passive castor wheel to support the chassis. This new approach combines some of the advantages of the compliance configurations, allowing for better maneuverability, collision detection and mitigation, as well as better human-robot interaction [7].

In the section Related Work, commonly-used wheel types and wheel configurations will be illustrated in order to cultivate a better understanding of the advantages offered by IWOS. In the following section, the physical configuration,

*The research leading to these results has received funding from the Austrian Research Promotion Agency (FFG) according to grant agreement 855409 (AutonomousFleet) and 854865 (TransportBuddy)

[1]Firat Berk Cakar, Eugen Kaltenegger, George Todoran and Markus Bader are with the Institute of Computer Engineering, Vienna University of Technology, Treitlstr. 1-3/4. Floor/E191-3, 1040 Vienna, Austria [firstname.lastname@tuwien.ac.at]

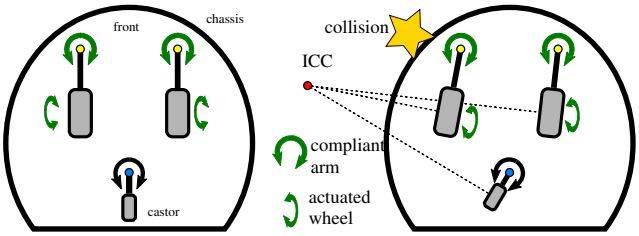


Fig. 1: IWOS with actuated/complient joints in green. A collision will be mitigated and the arms will be steered away from the contact point.

advantages and a some of the distinct use cases such as collision detection and mitigation, expressive navigation and adjustable ICC (Instantaneous Center of Curvature) will be presented. The penultimate section Simulation presents preliminary results with GazeboSim. Finally, the results are discussed in the conclusion section.

II. RELATED WORK

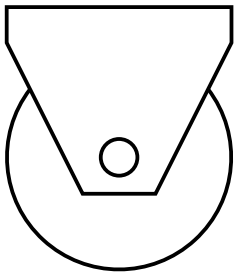
In this section, a brief overview of the most common wheel designs and wheel configurations in use will be presented, as well as the criteria used to evaluate the pros and cons of each. Detailed kinematic models are not included in this section, since there is a quantity of detailed literature on the subject, exceeding the scope of this paper.

A. Wheel Design

There are four major wheel classes shown in Fig. 2, each with specific advantages and disadvantages. Each wheel class has its own specific kinematics which effects the overall kinematics of robot motion to a great extent [4][9].

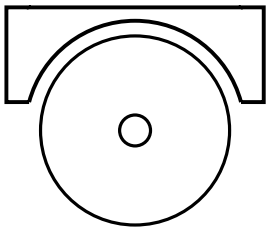
1) Standard Wheel: This wheel class, as the name suggests, is the most basic design with two degrees of freedom, the first being around the wheel axle and the second one being over the contact point with the ground. The center of rotation passes through this point. In order to move in a direction other than the one that the wheel is facing, the wheel must be steered along a vertical axis [3][4][9].

2) Castor Wheel: This design is an extension of the standard wheel. It still has two degrees of freedom, but the second one is the rotation around an offset steering joint. This offset between the ground contact point and the center of rotation leads to a force being applied to the robot chassis during a rotation [2][3][9]. IWOS exploits this effect to detect a collision [7].



(a) Standard Wheel

(b) Castor Wheel[1]



(c) Spherical Wheel

(d) Swedish Wheel[2]

Fig. 2: Four most common wheel types

3) Swedish Wheel: In this wheel design, there are passive rollers along the wheel which help reduce friction in other movement directions.

There is an extra degree of freedom in comparison to the standard wheel: rotation around the rollers. The angle that the rollers are mounted dictates the direction with reduced movement resistance. For example, there is a 90-degree variant called Swedish 90, which allows for movement perpendicular to wheel orientation. The main advantage is the following: With only one actively-powered joint, moving in many different trajectories with little friction is possible [3][6][9].

4) Spherical Wheel: This design allows for true omni-directional movement. A spherical wheel can be powered to spin in any given direction. The main disadvantage lies in the difficulty of realization of this design [3][4][9].

A final consideration that must be taken into account is suspension. In order to maintain contact with the ground, proper suspension must be employed, especially when there are more than three wheels in a given configuration. The most primitive approach would be to use rubber tires around the wheels[9].

B. Evaluation Criteria for Wheel Configurations

Prior to discussing some of the common configurations used in the main types of wheels, a brief overview of evaluation criteria will be given. Those criteria are important in assessing the various pros and cons of any particular configuration.

[1]By Lambtron - Own work, Public Domain, https://commons.wikimedia.org/w/index.php?curid=9925292

[2]Von Imetron GmbH - Imetron GmbH, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=28613180

1) Stability: The minimum number of wheels to achieve stability is just two. As long as the center of mass lies along the wheel axle, stability can be maintained. However, this configuration requires impractically large wheel diameters. In addition, motors with very high torque can lead to the robot making a third point of contact with the ground at initial start up.

The most common approach used to assure stability is to use three wheels. In this approach, the center of mass must remain inside the triangle formed by wheel's contact points. Although it is possible to further increase stability by using even more wheels, the resulting necessity for suspension diminishes the usability of such configurations [4][9].

2) Maneuverability: In mobile robotics, maneuverability is highly sought-after. The ideal scenario would be to achieve omni-directional movement, i.e. being able to move in any given direction, independently of where the robot is facing [2][3][4]. This can be obtained by Swedish or spherical wheels, however the mechanical complexity of realization is a deterrent.

Another approach to the achievement of omni-directional movement is to use four castor wheels with eight motors: four for spinning and four for steering. With this approach, even if the desired direction of motion is not ahead of the robot, castor wheels can be actively steered to reach the desired position. Furthermore, some robots can rotate on their own axes without changing their footprints. Although that would not be true omni-directional movement, it replicates the capability to some extent. This behavior can be achieved with a circular-shaped differential drive robot, which has its axis of rotation in the center[4][9].

Lastly, a car-like robot (Ackermann drive) has very poor maneuverability compared to the examples above. It has a turning radius larger than its footprint, and parking requires too many maneuvers[4][9].

3) Controllability: Usually, better maneuverability comes with worse controllability. This is due to more degrees of freedom offered by omni-directional designs. In order to achieve the desired motion, an increased amount of computation is required [8][9].

It should also be noted that the more maneuverability a design offers, the more difficult it is to maintain a robot's trajectory. For example, with an Ackermann drive, locking the front wheels is sufficient for travelling in a given direction, but with a differential drive, each wheel has to be kept at the same velocity with a precision which is not always easy to achieve[4][9].

All in all, there is no optimal wheel configuration that maximizes the advantages while minimizing the disadvantages. A configuration must be chosen based on its strengths for a given use case.

C. Common Wheel Configurations

In this section, a brief overview of frequently-employed wheel configurations will be given. Neither detailed kinematic models nor a great magnitude of small variations that

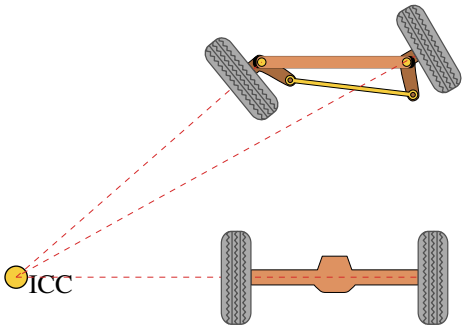


Fig. 3: Ackermann Drive [3]

utilize the general idea will not be covered, since it is beyond the scope of this paper.

1) Ackermann Drive: The Ackermann drive was developed in the mid-nineteenth century to achieve better maneuverability for coach-cars and is widely used in many commercial vehicles. The main idea behind the Ackermann drive was to provide steerable wheels with a correlated pivot, such that their axis intersect at a point with the rear wheel axis. This is achieved via an Ackermann linkage, which transfers the steering command to the wheels in such a way that they have different steering angles that satisfy the ICC without a slip in the wheels [1]. This relatively easy-to-understand concept is illustrated in Fig. 3.

This configuration is quite popular in hobby robotics, since it is relatively easy to use an RC car chassis. It offers poor maneuverability compared to an omni-drive but it is rather easy to control. This is a non-holonomic vehicle [3][4][9].

Fig. 4: Four Castor Wheels with eight motors - Nomad [4]

[3]User:Bromsklossderivative
work: Andy Dingley (talk) -Ackermann.svg, CC BY-SA 3.0,
https://commons.wikimedia.org/w/index.php?curid=11038290

2) Omni-Drive - Four Castor Wheels and Eight Motors: Omni-directionality can be achieved with four castor wheels that are actuated by four motors that control the individual spin of each wheel and four other motors that control the steering [5][9]. Nomad XR4000 shown in Fig. 4, which is no longer produced, employs this configuration. A similar configuration, independent four-wheel steering, is used in the robot Blue [10].

Fig. 5: Four Swedish Wheels - Uranus [5]

3) Omni-Drive - Four Swedish Wheels: This configuration has been employed in many research robots, e.g. on Carnegie Melon Uranus shown in Fig. 5. It provides for omni-directional movement, which allows movement in any given direction and can even rotate around its own axis at the same time. This can be achieved with different rotation directions and speeds on each wheel [9].

4) Skid-Steer Wheels: This configuration is usually employed in tanks and similar mobile robots that utilize threads instead of wheels. A very popular wheeled implementation is the ”differential drive”, which is very common in mobile robotics shown in Fig. 6. The underlying idea is to control the direction by employing different velocities in each wheel. There is no additional mechanism to control steering. The robot moves along a circle, dictated by wheel geometry and the velocity difference between wheels. Unlike any other drive configuration mentioned here, wheel slippage is unavoidable, making operating on certain ground types challenging. This is an non-holonomic vehicle [2][4][9].

D. Holonomic vs Non-Holonomic Constraints

The concept of holonomy has a very broad definition in mathematics. In mobile robotics, it specifically refers to kinematic constraints of the robot chassis.

[4]”Distributed temporal event mapping and fusion - Scientific Figure on ResearchGate. Available from: `https://www.researchgate.net/figure/The-Nomad-XR4000-mobile-robot_fig3_237445040`

[5]By Gwpcmu - Own work, CC BY 3.0,
https://commons.wikimedia.org/w/index.php?curid=11440618

[6]By NikNaks talk - gallery - wikipedia 11:44, 30 May 2010 (UTC) - DifferentailSteering.gif; original file: [1], CC BY-SA 3.0,
https://commons.wikimedia.org/w/index.php?curid=10494749

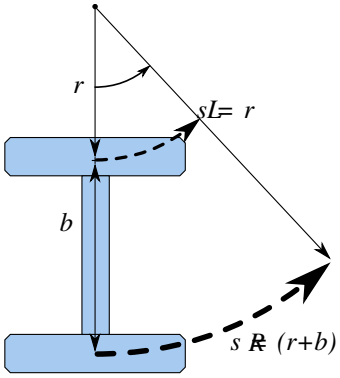


Fig. 6: Differential Drive [6]

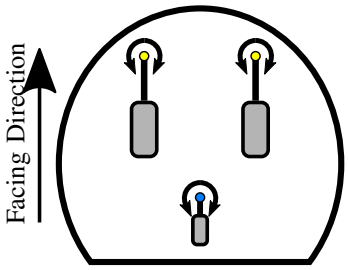


Fig. 7: Mechanical configuration

A holonomic robot has zero non-holonomic constraints whereas a non-holonomic robot has at least one non-holonomic constraint. Non-holonomic constraints are defined by a differential relationship of position variables and can not be integrated to acquire the position variable [9].

There is various literature on this topic that explains it in a detailed fashion for each application, but in a very broad sense it can explained in the following way: An omnidirectional drive is holonomic since it allows movement in any given direction. This can also be expressed as "having number of degrees of freedom equal to the number of coordinates needed to specify the configuration of the system" [5]. On the other hand, an Ackermann drive or a skid/steer configuration is non-holonomic and can not move to every desired position directly [3][4].

III. INDEPENDENT OFFSET WHEEL STEERING

In this chapter, a new approach for wheeled mobile robot locomotion will be presented. First, the mechanical configuration will be illustrated. Following that, four uses cases that would maximize the advantages of this configuration will be given.

A. Mechanical Configuration

The fundamental mechanical configuration consists of two standard wheels attached at the end of two steerable arms parallel to the ground which connect to the robot chassis and a third passive castor wheel. There are four motors in total, two for each arm in which one motor is used for spinning the wheel and the other for steering the arm. The third wheel is not motorized and adjusts itself to the ICC (Fig. 7).

The arms have preferably the same length and the wheels, the same diameter. They are attached to the robot's chassis on the front (relative to the robot's default direction of movement) and the wheels are attached to the arms further back relative to the steerable arm contact point. Ideally, they should be placed symmetrically on the chassis.

The third wheel has a smaller wheel diameter compared to the wheels attached to the arms and is closer to ground. It is also located further back compared to the steerable arms.

The arms are steerable to a limited range, allowing for a steering angle of less than 180 degrees. The key point here is that the steering is done at an offset from the ground contact point, allowing the robot chassis to change its orientation independent of the actual driving direction.

The robot chassis has preferably a circular or semi-circular shape, which is to its advantage in collision mitigation.

B. Possible Use Cases

IWOS offers various possibilities that can be utilized in wheeled locomotion. In this section, four distinguishing use cases will be presented.

1) Collision Mitigation: In the event of a collision, the steerable arms experience a torque which changes their orientation in such a way that the robot chassis automatically turns away from the point of collision. This could be used in a non-motorized use case of steerable arms, for example, stabilized by springs (Figure 1).

In the case of proposed active steering, once a collision has been sensed, the steerable arms could be actively controlled to steer away from the point of collision.

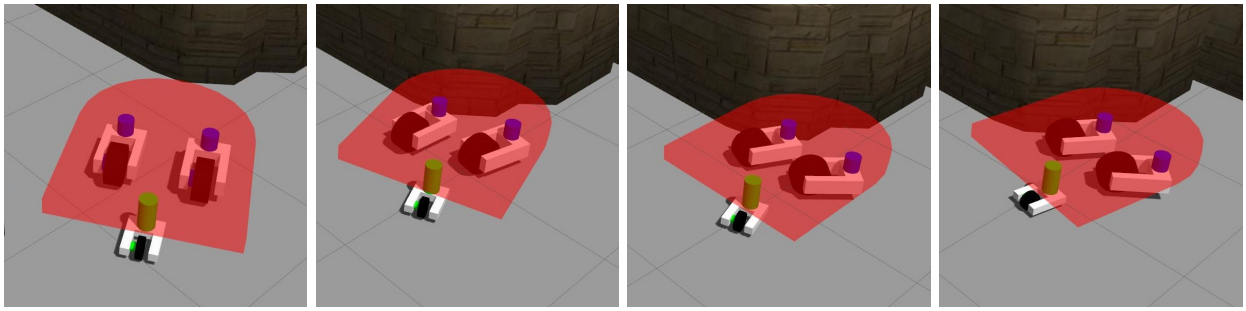
In both cases, a collision would be mitigated and any possible damage to the robot chassis or to the collision point would be considerably reduced.

If contact with the collision point is maintained, a sliding motion could be achieved rather than a head on crash with a small adjustment to the steerable arm's orientation. To further enhance this behavior, small wheels could be horizontally mounted to the front of the robot. Those small wheels would assist the robot greatly even in the case of a high friction environment.

2) Lateral Collision Detection: In the case of a lateral collision, the robot chassis would experience a force from the collision point. This force would be transferred to the steering arms, forcing them to change their orientation. With a simple pressure sensor or with proper torque control of the motors, the torque experienced could be processed to determine the direction of collision. This could be further used to mitigate damage resulting from a collision (not avoidance, since a collision would have already happened).

It is more suitable to have a circular or a semi-circular chassis in order to better transfer the collision force experienced as torque into the steerable arms.

3) Expressive Navigation: Signaling: Many mobile robots face the momentary direction in which they are driving rather



(a) The vehicle approaches an object with the steerable arms mounted on a spring.

(b) The vehicle is in contact with the wall and the collision has been mitigated.

(c) Due to the torque applied, the springs that hold the steering arms twist and steer the vehicle away from the object.

(d) The vehicle is steering away from the object. Because of the passive suspension, the vehicle will follow the wall.

Fig. 8: Collision mitigation and recovery. An IWOS vehicle is capable of mitigating the collision and leading the vehicle away from the contact point.

than their desired goal position. This causes problems, especially in environments with humans, since this movement is counter-intuitive and makes it harder for humans to predict robots' behavior.

Thanks to the fact that there is an offset between the wheel-ground contact point and robot-steerable arm connection, a mobile robot using IWOS possesses the capability of driving in a direction that differs from the direction it faces. The robot can rotate its front without changing its driving direction, allowing it to signal upcoming changes in movement. This feature can be especially useful in indicating evasion maneuvers in narrow environments shown in Fig. 9.

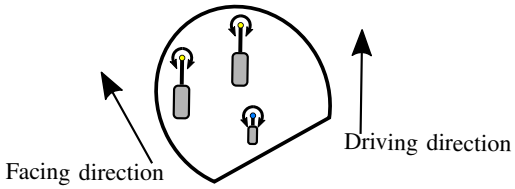
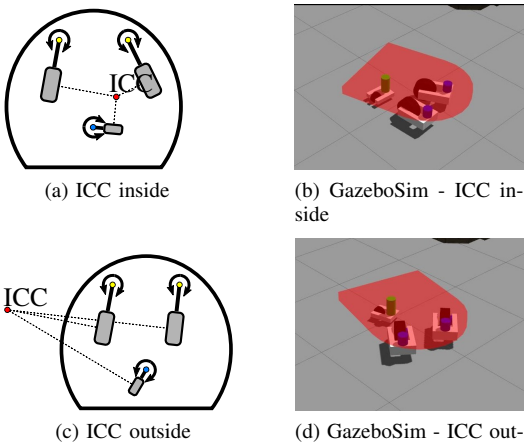


Fig. 9: The actual driving direction differs from the actual facing direction and can signal upcoming turn maneuvers to humans.

4) Adjustable ICC: Better Maneuverability and Collision Avoidance: Another advantage offered by IWOS is that of being able to adjust the ICC solely by steering the arms. Depending on the arm orientation, the ICC can be under the robot's footprint, which allows for rotation along a very small circle or even at its own axis. Another possibility is to have the ICC away from the robot, which allows for better maneuverability. The ability to place the ICC on different locations makes IWOS nearly omni-directional; only the maximum steering angle dictated by the arms hinders omni-directionality shown in Fig. 10.

C. Advantages

IWOS presents many advantages compared to its popular counterparts. In this section those advantages will be listed and briefly explained.



(a) ICC inside

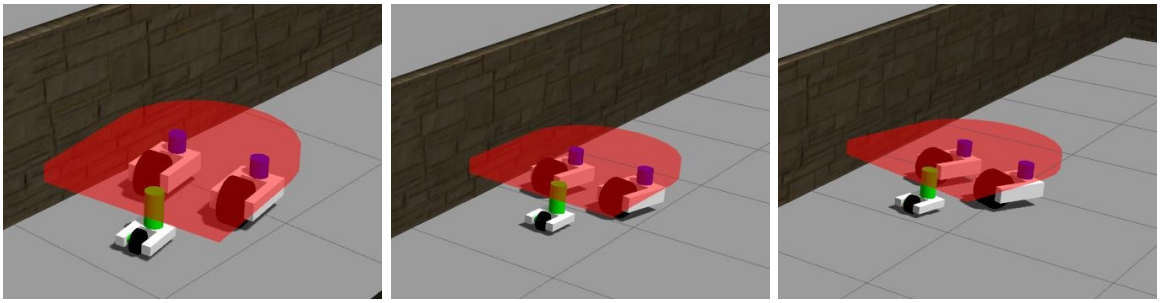
(b) GazeboSim - ICC inside

(c) ICC outside

(d) GazeboSim - ICC outside

Fig. 10: IWOS offers adjustable ICC only by steering the arms.

- High Maneuverability: IWOS has nearly omni-directional movement capability.
- High Controllability: Although high controllability doesn't usually come hand-in-hand with high maneuverability, with IWOS it is possible to control the robot with limited effort. It is even possible to run any pure differential drive navigation method by just locking the steerable arms.
- Better Collision Recovery: In the event of a collision, going backwards is not the only option that IWOS has. Being able to move in a less limited way provides better collision recovery. In addition, possible sliding capability further enhances this advantage.
- Relatively Simple Mechanical Design: Operating with standard wheels and a castor, IWOS has a simpler mechanical construction.
- Better Human Interaction: Thanks to its ability to indicate future movements, IWOS presents a more intuitive human-robot experience.



(a) The vehicle will press itself parallel against the wall. A vehicle with differential drive would become wedged.

(b) The vehicle has contact.

(c) The vehicle is able to leave the contact without scratching itself further.

Fig. 11: Sliding motion that allows the contact with the collision point to be broken in a way that is not possible with a differential drive.

IV. SIMULATION & PRELIMINARY RESULTS

A vehicle that employs IWOS has been simulated With ROS and GazeboSim to test its advantages. To minimize vehicle control, a spring with constant torque have been mounted on each steerable arm. Fig. 8 demonstrates how the vehicle behaves in an event of collision. The vehicle steering leads the vehicle away from the contact and thus minimizes the impact. An active control can slow down the vehicle or change the steering angle. If the contact with the object is exactly in the middle, the arms would not experience a torque and not rotate. In this case an active control with contact sensors would be necessary to turn the vehicle away from the collision point.

Fig. 11 shows a vehicle driving parallel to the wall. Such a maneuver is critical with a differential drive because it can become wedged together with the wall. If a vehicle with a differential drive finds itself in this situation, it can at best be released by a reverse movement. If this is not possible, the vehicle must rotate while in contact with the object which can lead to possible damage to the vehicle or to the object. On the other hand, a vehicle with IWOS can break contact with the collision point by steering the arms which causes a lateral movement of the chassis. This enables the vehicle to drive away safely form the object.

V. CONCLUSIONS

IWOS combines many advantages of commonly-used wheel configurations in a rather simple mechanical structure. A simulation done in Gazebo which includes a proper physics engine, demonstrates preliminary results and illustrates some of the possible use cases. This approach presents itself as a valid and easily-implementable wheel configuration with many extra applications that are begging to be explored.

REFERENCES

[1] D. S. Apostolopoulos, "Analytical configuration of wheeled robotic locomotion," Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, Apr. 2001. [Online]. Available: https://www.ri.cmu.edu/publications/analytical-configuration-of-wheeled-robotic-locomotion/

[2] D. R. P. B B V L Deepak and A. K. Jha, "Kinematic model of wheeled mobile robots," *Recent Trends in Engineering & Technology*, vol. 05, Mar. 2011.

[3] I. Doroftei, V. Grosu, and V. Spinu, *Omnidirectional Mobile Robot - Design and Implementation*, 09 2007.

[4] K. Goris, "Autonomous mobile robot mechanical design," Master's thesis, Vrije Universiteit Brussel, Brussels, Belgium, 2004-2005.

[5] R. Holmberg and O. Khatib, "Development and control of a holonomic mobile robot for mobile manipulation tasks," *The International Journal of Robotics Research*, vol. 19, no. 11, pp. 1066–1074, 2000. [Online]. Available: https://doi.org/10.1177/02783640022067977

[6] G. Indiveri, "Swedish wheeled omnidirectional mobile robots: Kinematics analysis and control," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 164–171, Feb 2009.

[7] T. U. of Vienna, "Mobiles system," Austrian Patent Request 18 740, Mar. 20, 2018.

[8] H. Oliveira, A. Sousa, A. Moreira, and P. Costa, *Modeling and Assessing of Omni-Directional Robots with Three and Four Wheels*, 12 2009.

[9] D. S. Roland Siegwart, Illah R. Nourbakhsh, *Introduction To Mobile Autonomous Mobile Robots*. Cambridge, Massachusetts: The MIT Press, 2011, ch. 3.

[10] G. Todoran and M. Bader, "Expressive navigation and local path-planning of independent steering autonomous systems," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 4742–4749.

An Autonomous Mobile Handling Robot Using Object Recognition

Johannes N. Rauer[1]; Wilfried Wöber[1] and Mohamed Aburaia[1]

Abstract— **Due to the trend away from mass production to highly customized goods, there is a great demand for versatile robots in the manufacturing industry. Classic fixed-programmed industrial robots and rail-bound transport vehicles, which are restricted to transporting standardized boxes, do not offer enough flexibility for modern factories. Machine learning methods and 3D vision can give manipulators the ability to perceive and understand the environment and therefore enable them to perform object manipulation tasks. State of the art grasp-detection methods rely on data with cumbersome annotated grasp-poses, while labelled data for object recognition only is easier to gather.**

This work describes the development of an automatic transport robot using a sensitive manipulator and 3D vision for autonomous transport of objects. This mobile manipulator is able to drive flexible paths, localize predefined objects and grasp them using an out-of-the-box neural network for object detection and hand-crafted methods for extracting grasp-points from depth images to avoid cumbersome grasp-point-annotated training data. Furthermore, this paper discusses problems occurring when a neural network trained on human-captured photos is applied to robot-view images.

I. Introduction

Since the beginning of the 1960s, traditional fixed-base robots have been used in factories [40]. Classic industrial robots are unable to deal with the uncertainties in the real world, due to a lack of sensing capabilities. Therefore, stationary manipulators have mainly been deployed in mass production, where a constant environment can be assumed and thus highly engineered programs can work efficiently [33], [35]. Giving manipulators the ability to perceive the environment and using these perceptions for machine learning, flexible part recognition in unstructured environments is possible [8]. Due to the trend away from mass production to highly customized goods, there is a great demand for versatile robots in the manufacturing industry [17], [16]. Over the last few years e.g. the production of cars has been highly individualized. As Pavlichenko et al. [27] describe, this has made ”kitting” necessary, where all parts of a car are collected in a warehouse and brought to the assembly line just in time as a ”kit”. This task is frequently performed by warehouseman due to the high flexibility needed to find, collect and transport specific components [27]. Therefore, part handling during assembly is the only task in the automotive industry with an automation level below 30% [17].

For such intra-logistical transport tasks, mobile manipulators have been developed. They consist of sensitive manipulators which are associated with mobile transport vehicles to combine the advantages of both types of robots: Working together with humans and grasping individual objects, as well as changing the location autonomously to extend the workspace of the manipulator [16], [7]. A mobile manipulator should be able to drive autonomously through a shop floor, detect parts with sensors, grasp them with its manipulator and transport them to production facilities [27].

While there already exist well established methods for path planning and safe navigation of mobile robots, grasping objects flexibly remains a challenging task [25], [4]. Since a large amount of grasp-pose-annotated training data is needed for automatic grasp-detection in machine learning, training new objects is complex and expensive [28].

Therefore, this paper describes the development of a mobile manipulator using an out-of-the-box object recognition framework and handcrafted grasp-point calculation algorithms to allow grasping of standard objects with pre-trained algorithms and not requiring cumbersome annotated grasp-data. It describes a software concept for a mobile platform extended with a sensitive manipulator, a gripper and a 3D sensor. This system enables path planning, searching for predefined objects using an image stream and machine learning, as well as grasping and transporting them. Since the mobile robot's integrated path planner is being used, this part is not considered in more detail. Furthermore, this paper discusses problems occurring when a neural network for object detection trained on human-captured photos is used with robot-view images.

The following chapters are structured as follows: Section II presents the state of the art of 3D-vision-systems, grasp detection and object detection in machine learning with a focus on deep convolutional neural networks. In section III, the software representation of the robot and the implemented software are described, as well as the hardware and structure of the mobile manipulator. Section IV explains the abilities of the robot and verifies and discusses them using experiments, followed by summary and future work in section V.

II. State of the Art

In the following section, the state of the art of 3D-vision-systems, grasp detection and machine learning is discussed.

A. 3D Vision Systems for Robotic Manipulation

Visual sensors play an important role in object detection and manipulation. 3D vision systems, which map the 2D image pixels to 3D world coordinates, are most suitable for this task, if they are mounted to the arm of a robot [42], [2].

According to Giancola et al. [13], there are three main 3D vision technologies on the market which are suitable for such

[1]Department of industrial engineering, University of Applied Sciences Technikum Wien, 1200 Wien, Austria <mr17m019,woeber,aburaia>@technikum-wien.at

an application: Time-of-Flight (TOF), structured-light and active stereoscopy, whereby the last two use a triangulation process to estimate the depth. TOF-systems measure depth-distance directly from the time a light beam needs until it is reflected back [13]. Active stereoscopy devices have the advantage over classical stereoscopic depth systems, that they project an infrared pattern and are therefore able to find matching points for the triangulation also on texture-less surfaces [19]. 3D cameras often include an RGB-sensor and therefore provide RGB-D images [13].

B. Machine Learning for Object Detection

To extract information such as object types and positions from image data, neural networks are used [6], [1]. Deep neural networks [15] consist of numerous layers with artificial neurons, which allow them to handle big data effectively. A convolutional neural network (CNN) [20] is a special type of deep network, which is designed to work with data with a grid-like topology, such as images [1], [15]. An example CNN is presented in Fig. 1. It consists of a series of convolutional and pooling layers which act as feature extractors or filters and fully connected layers for classification or regression [1], [15].

To predict besides class probabilities also bounding boxes for objects in a single-shot, Redmon et al. have presented the CNN YOLOnet [31]. Classic approaches [1], [15] work with a sliding window where a classifier is run over the complete image. More recent R-CNN methods first propose regions where objects may be and then run a classifier on these bounding boxes [14], [9], but due to the high number of individual components, these networks are slower and hard to optimize [31]. YOLOv3 [32] can process images at up to 45 fps to allow fast reaction to changes in the environment which is particularly important for object manipulation [30].

C. Grasp Detection

The goal of robotic grasp detection is to detect and calculate graspable regions in images of objects and compute a trajectory to them [21], [41], [5]. Due to challenging variations of the objects and light conditions, as well as occlusions and clutter, grasp-detection systems lag far behind human performance levels [4].

Some grasp-detection methods focus on the object pose estimation and lookup of the corresponding grasp points or gripper pose in a database [41], [24], [5]. At other approaches, neural networks are trained to provide the full grasp configuration [30], [39] or the success-probability for a grasp in a given gripper pose [28], [22]. Due to the creation of an appropriate training dataset being currently a big obstacle, numerous approaches generate data automatically using robots or simulations [28], [24], [39]. Labeling data manually is a challenging and time-consuming task due to objects being graspable in multiple ways and the necessary label being of a high dimension and gripper-dependent [28].

Therefore, in this work grasp points are estimated using an out-of-the-box object detection network and hand-crafted, object dependent methods which extract features from depth-data. This approach does only require training data for object detection, which is available in large number [10].

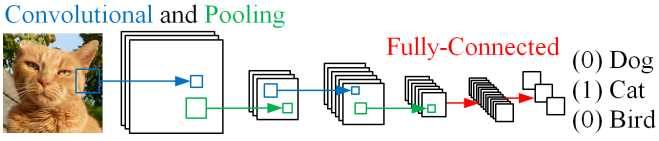


Fig. 1. Typical structure of a CNN: Blue - Convolutional Layer, Green - Pooling Layer, Red - Fully-Connected Layer

III. Methods and Implementation

In this section, the representation of the mobile manipulator and its environment with coordinate frames is discussed and the structure of the robot as well as all components are introduced. Furthermore, the implementation of the robot's software is explained.

A. Robot Representation

The presented mobile manipulator consist of a MiR100 mobile platform [26] and an UR5 articulated robot [37], as well as a gripper and a depth camera (see III-C). All components are represented using coordinate frames which are connected with each other and with the world-frame via geometrical relations (see Fig. 2).

The *map*-frame represents the robot's world coordinate system and functions thereby as fixed origin. As the robot moves, the *odometry*-frame is moved according to the wheel-odometry and signals from the inertial measurement unit continuously in the *map*-frame. The *base*-frame is rigidly attached to the mobile platform and moved in the *map*-frame according to sensor observations, such as the localization in the map using laser-scanner signals [23]. The *base-UR*-frame is attached to the *base*-frame statically and serves as reference for the frames in the joints of the manipulator. At the end of the kinematic chain, a frame for the tool center point of the gripper – in which planning of arm trajectories takes place – as well as a frame for the camera – in which the pose of the object is calculated – are connected statically to the *EEF*-frame (end-effector-frame).

If an object is detected by the robot, its xyz-position relatively to the camera is calculated. With the knowledge of the transformations between all the other frames, the pose of the object in the robot's world is computable. The transformations between the *object*-frame and the *EEF*-frame are determined by calibrating the camera.

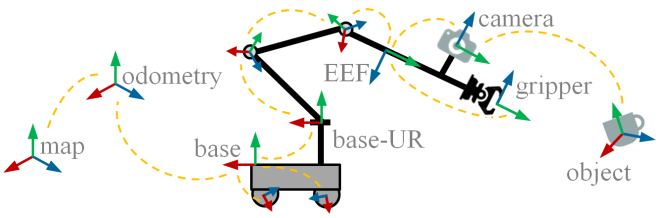


Fig. 2. Positions and connections of the robot's frames. Orientations are exemplary. To reduce complexity, unimportant frames are not shown or labeled. The dashed lines illustrate the mutual dependencies of the frames.

B. Implementation of the Software

The software of the mobile manipulator is based on ROS (Robot operating system) [29]. The structure of the basic ROS-nodes, which provide communication to the hardware and the ROS-Core, is presented in Fig. 3. When the task-execution is started, the connections to the basic nodes are established as illustrated. The program flow of the task-execution is presented in Fig. 4 and explained in III-B.2. The basic nodes are described in the following section:

1) Basic Nodes: The basic nodes are responsible for providing communication with the hardware and elementary information of the robot, such as its kinematic structure. Therefore, a robot model including its structure and the coordinate frames is sent to the core by the *Model-Uploader*. The *ROS-Core* provides it to all nodes to enable visualizations and calculation of trajectories.

Using this robot-description, tf [12] builds a tree as illustrated in Fig. 2. The module *TF-Transformer* serves as interface between the *Task-Execution-Software* and tf-tree, providing necessary transformations for robot movements. A TCP-IP connection to the UR5 is established using *UR-Driver* [3]. This node provides the actual robot pose and is able to send trajectories to the robot. The ROS-package *MoveIt!* [36] communicates with the *UR-Driver* and serves as motion planner and kinematics solver for the UR-robot. It offers a programming interface to get the robot's joint states and arm pose. Given a target pose it solves the inverse kinematics and calculates a trajectory.

The communication between the industrial PC and the MiR's controller is established using *MiR-Driver* [11]. Since the MiR launches its own ROS-Core, this bridge establishes a connection to it, reads all topics and provides the same topics in the local ROS-environment on the industrial computer. It reads all messages of the MiR and publishes them to the local ROS-Core – e.g. the actual pose. If e.g. a goal pose is sent to a MiR-topic, it is forwarded to the MiR and published to its

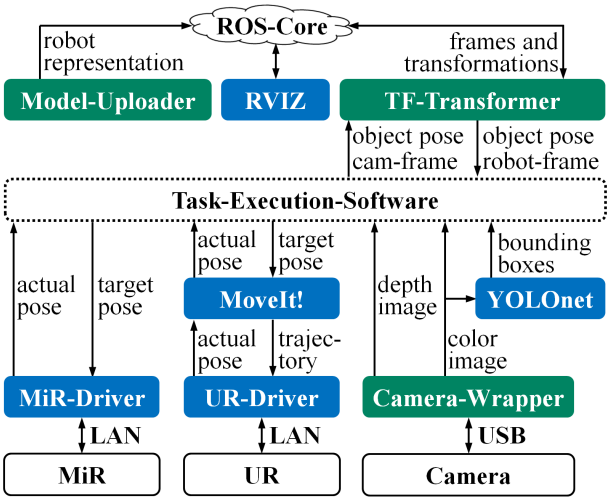


Fig. 3. Structure of the basic nodes of the software, which provide topics to communicate with the hardware. Green nodes have been developed. Blue nodes symbolize provided ROS-Packages.

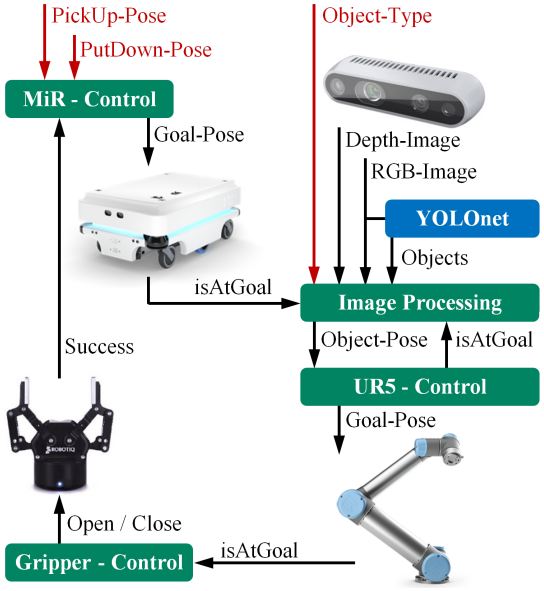


Fig. 4. Program flow of the robot's task-execution-software. Controls all components to grasp an object of given type at given pick-up-position and to transport it to given put-down-position. User inputs are marked red. Green nodes have been developed. Blue nodes symbolize provided ROS-Packages.

ROS-Core. Path planning and safe movement is implemented by the manufacturer and therefore not discussed further.

In addition to driving and moving the robotic arm, perceiving the environment is necessary to enable the mobile manipulator to grasp objects. Therefore, *Camera-Wrapper* connects to the RGB-D-camera and publishes color and depth images. The CNN *YOLOnet* subscribes to the color-image-stream and publishes the object-types and -positions in the image frames as bounding boxes [32].

With the discussed basic services and communications, the actual task execution of driving, searching, locating and grasping the object is possible, which is explained in the following section:

2) Task execution: The program which is necessary for executing the task itself is started by user-input and calls different modules with special purposes. This enables to change modules and implement e.g. a different grasping algorithm with no effect to other program parts. In Fig. 4, the general procedure for grasping an object is presented abstractly and following the parts are described in detail.

A complete task-execution consist of driving to a goal, searching for the object and calculating the grasp point, as well as moving the robotic arm and grasping the object at the calculated point. Then the robot can move to a goal position and put the object down. For sending the robot to a target pose, the module *MiR-Control* builds a ROS-Message from a given PickUp-Pose, that is forwarded to the robot. The MiR-robot calculates its path and drives to the goal without further intervention, as it is able to perform localization, path planning and obstacle detection autonomously [25].

When the MiR is at the goal, the pose of the object has to be determined. The camera's RGB-image is analyzed by *YOLOnet*, which offers the object-types and positions of the

bounding boxes for each recognized object in the image [32]. Together with the depth-image and the camera intrinsic parameters, the image-processing-module calculates the coordinates of the demanded object in 3D-coordinates relatively to the camera. As the pose estimation is not precisely when analyzing a single shot, the UR is moved multiple times and different image processing steps are performed: First the UR is moved to search positions where the camera has a good overview of the scene to search for the object. If it is detected, the robot moves about 35 cm over the item with the camera parallel to the plane under the object for grasp detection. It is not possible to only scan the table from this planar position, since object detection does not work for certain poses, as described in IV-B.

Grasp detection is performed for each object individually and the processes are designed manually. To detect grasping points, objects are separated from background using the depth-image and circle detection is performed (see Fig. 5). Bottles can be grasped using the estimated circle's center. The grasp point of cups is calculated using the center of the circle and the end of the handle, which is the furthest point from the center. Using the positions of these points with regard to the camera, the necessary grasp angle is calculable. These grasp points are provided to the tf-tree to calculate the grasp point relatively to the robot's frame. Using this transformation, *UR5-Control* calculates the goal pose of the UR5 to grasp the object.

After the UR has reached the aimed grasping pose, *Gripper-Control* sends a command to the robot to close the gripper. The gripper is connected to the UR to make it controllable with the robot's teach pendant. Therefore, commands have to be sent to the gripper via the UR's LAN-connection. If the object has successfully been grasped by the robot, the MiR is sent to the PutDown-Pose where the UR5 can lie down the object.

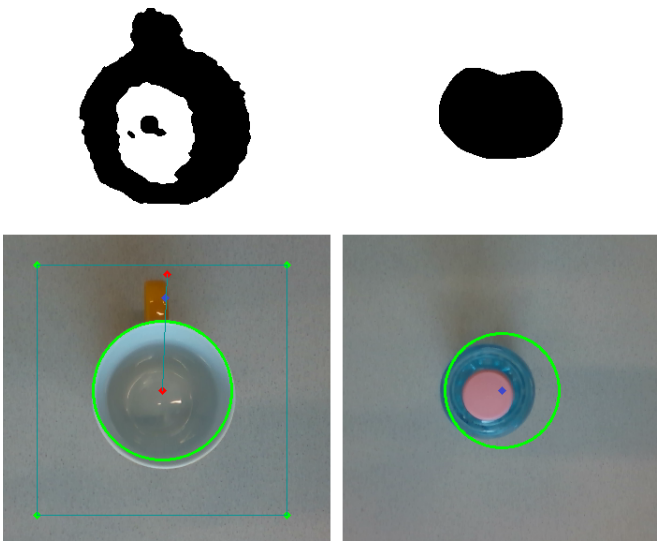


Fig. 5. Detection of grasp point (blue). Top: Depth-images separated from background. Left: Grasp-Point located at the handle of a cup. Right: Grasp-Point at the center of a bottle.

C. Components & Structure of the Robot

The mobile platform which serves as base for all components is a MiR100 [26]. It is able to calculate and travel the path to a given goal autonomously and is thereby aware of safety issues [25]. On top of an aluminum structure, a collaborative articulated UR5 robot [37] is installed. This manipulator weights about 20 kg at a payload of 5 kg and a maximum grasp-distance of 850 mm. It is connected to the MiR's ethernet-switch via LAN.

A 3D-printed fixture with an Intel Realsense D435 RGB-D-camera [18] is mounted onto the flange of the manipulator. This active stereoscopy based camera provides RGB-D-images between 0.2 and 4.5 meters in depth at up to 90 frames per second. It is connected via USB-C to the industrial computer. The parallel-jaw gripper 2F-85 [34] is mounted onto the camera fixture at the end of the kinematic chain of the robot. With its payload of 5 kg at an opening width of 85 mm it is suitable to grasp small objects. It is connected to the control unit of the UR5 via USB.

To control all subcomponents of the mobile manipulator, the industrial computer ECS-9100-GTX1050T [38] is installed on the mobile robot. With its NVIDIA GeForce GTX 1050 Ti graphics processor it is suitable for image processing tasks. The industrial computer runs on Ubuntu 16.04 and hosts a ROS-Kinetic-Core. It is connected to the ethernet-switch of the MiR and thereby able to access and control all components.

IV. Results & Discussion

In the following section, the abilities of the robot are described and verified using experiments, focusing on the quality of object detection. Furthermore, reasons for problems of the object recognition are presented and discussed.

A. Abilities of the Robot & Success Rate

The presented mobile manipulator is able to localize and classify predefined objects, to perform a path and movement planning, to drive to the object, and to grasp and transport it as well as put it down. The current supported objects are cups and bowls, whereby a software platform has been created which can be extended to manipulate other objects.

A qualitative experiment has been performed to evaluate the robot's success rate: the mobile manipulator is sent to the same goal multiple times, where one of the objects is randomly placed. The robot is able to grasp the object successfully in each of 30 repetitions, although the robot's pose varies up to approximately 25 cm and 20°.

Separation of object detection and grasp-point calculation enables simple and fast implementation of grasp-control for standard objects using a pre-trained network without the need of cumbersome training-data generation and annotation. It has to be noted that new methods must be hand-crafted for additional objects and grasp-point estimation from depth-data as presented is error-prone if multiple objects are visible. Therefore, if to some extend constant ambient conditions can not be ensured, switching to an integrated solution with more

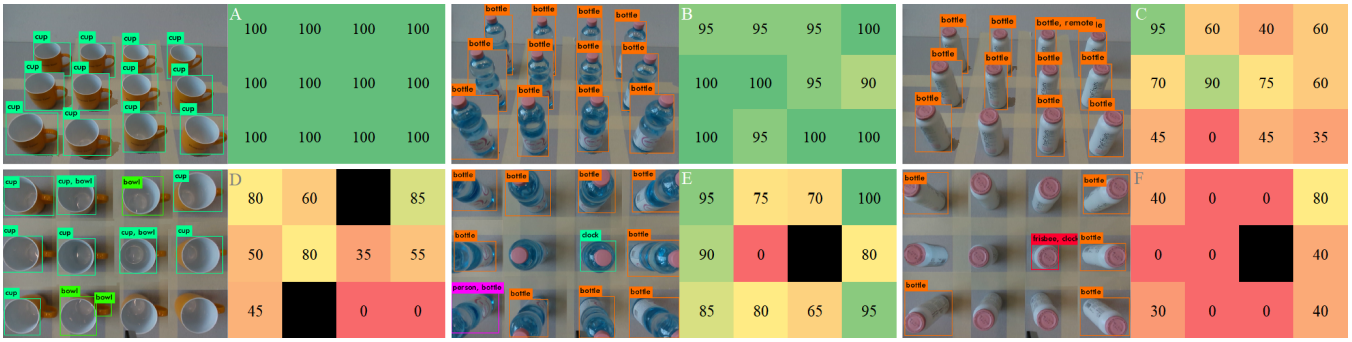


Fig. 6. Quality of object detection: Cup (left), Standard water bottle (center), Unconventional smoothie bottle (right). Camera has been oriented 40° (top) and 90° (bottom). Images show the merged pictures of all positions. Colored fields illustrate the calculated probability of the object being of the correct class. Black fields indicate wrong detections.

complex training data such as PoseCNN [41] or Silhonet [5] might be preferable.

B. Object Recognition

An object can only be grasped if it is recognized correctly first. Since the hand-crafted methods for grasp-point calculations need a planar view onto the object, scanning a surface for objects with the camera parallel to it would be most appropriate. This could save time since only one pose before grasping the object would be possible. The quality of the object detection, depending on the angle of the camera and the position of the object in the captured image has therefore been evaluated practically using experiments. For this, the robot has been located in front of a table showing a 4x3 grid under constant lighting conditions. The camera has been oriented by manipulating the robot's arm to show the whole grid at a specific angle. Afterwards, an object has been presented to the camera at the different positions of the grid under constant orientation. It has been evaluated if the object is recognized in the captured image and the quality of this detection has been analyzed using the mean class-probabilities provided by the neural network. This procedure has been performed at camera angles of 40° and 90° at a distance of 750 and 600 mm to the grid's center respectively, using three different objects: A cup with great contrast, a standard water bottle and a particular smoothie bottle. The objects have been oriented with the cup's handle pointing 90° to the right and the bottle's labels pointing to the robot.

The results of these experiments are presented in Fig. 6. The color of the fields indicate the probability of an object being of the correct class, whereby red marks positions where no class could be determined and black illustrates detections of a wrong class. The detection rate of cups and standard water bottles from a side view are very high, as Fig. 6 (A, B) shows. Unusual bottles as captured in (C) are less likely to be recognized correctly. The experiments show significant lower success rates of the object recognition for top view images. Cups are detected less accurate at all locations, whereby especially the lowest positions cause problems. Detecting cups incorrectly as bowls is also a common mistake (D). The detection rate of water bottles from the top view (E) is also smaller than from the side, whereby no correct detection of bottles shown from exactly above at the center positions are observable. This is visible for the smoothie bottles too (F). It is also interesting to note that bottles in the top view images (E, F) at a specific location have always been identified as clocks with a probability of 50%.

These experiments show clear weaknesses of the used training data for a robotic application. The ImageNet dataset [10] which has been used to train YOLOnet is using images found on the Internet. Therefore most of the objects are captured in their natural position as humans see them – e.g. most of the images of cups show medium-sized ceramic mugs with the handle clearly visible from an angle of about 45°. There are only few images showing cups in aerial perspective, from a flat angle or upside-down. The neural network learns from this insufficient dataset and thereby struggles to generalize to objects in new or unusual poses or configurations.

The results demonstrate the importance of an accurate and application specific training-dataset for neural networks. To enable high detection rates, images of concrete objects to be recognized from real-life perspectives have to be included in sufficient number.

V. Summary and Future Work

In this paper, an automatic transport robot using a sensitive manipulator and 3D vision sensors for autonomous object-transport has been presented. This mobile manipulator is able to drive autonomously, localize predefined objects and grasp them using an out-of-the-box neural network for object detection and hand-crafted methods for extracting grasp-points from depth images. A modular software based on ROS has been developed, which has the advantage of the basic nodes and hardware communication being decoupled from the actual task execution software.

Experiments have shown that despite inaccuracies of the mobile robot's goal, objects can be grasped successfully if the scene is captured from different perspectives. The main weakness is detecting objects reliable from exactly above, which is the necessary pose for the hand-crafted grasp-point detection to work, due to insufficient training of the neural network.

Therefore, in future work the neural network should be trained using a different dataset which shows objects as the robot's camera captures them. Using a neural network which performs object pose estimation would make detection of grasp points independent of image processing and therefore raise robustness. To enable an easy extendability to detect and grasp new objects, automatic generation of training data using a robot or rendering of objects should be considered.

REFERENCES

[1] Q. Abbas, M. E. Ibrahim, and M. A. Jaffar, "A comprehensive review of recent advances on deep vision systems," May 2018.

[2] M. M. Ali, H. Liu, R. Stoll, and K. Thurow, "Arm grasping for mobile robot transportation using kinect sensor and kinematic analysis," in *2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, May 2015, pp. 516–521.

[3] T. T. Andersen, *Optimizing the Universal Robots ROS driver*. Technical University of Denmark, 2015.

[4] U. Asif, M. Bennamoun, and F. Sohel, "Real-time pose estimation of rigid objects using rgb-d imagery," in *Proceedings of the 2013 IEEE 8th Conference on Industrial Electronics and Applications, ICIEA 2013*. IEEE, June 2013, pp. 1692–1699.

[5] G. Billings and M. Johnson-Roberson, "Silhonet: An rgb method for 3d object pose estimation and grasp planning," *CoRR*, vol. abs/1809.06893, Sept. 2018.

[6] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2006.

[7] Z.-E. Chebab, J.-C. Fauroux, N. Bouton, Y. Mezouar, and L. Sabourin, "Autonomous collaborative mobile manipulators : State of the art," in *TrC-IFToMM Symposium on Theory of Machines and Mechanisms*, Izmir, Turkey, 2015.

[8] C. H. Corbato, M. Bharatheesha, J. Van Egmond, J. Ju, and M. Wisse, "Integrating different levels of automation: Lessons from winning the amazon robotics challenge 2016," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4916–4926, Nov. 2018.

[9] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *NIPS*, 2016.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

[11] DFKI Robotics Innovation Center. mir_robot. [Online]. Available: http://wiki.ros.org/mir_driver

[12] T. Foote, "tf: The transform library," in *Technologies for Practical Robot Applications (TePRA), 2013 IEEE International Conference*, ser. Open-Source Software workshop, April 2013, pp. 1–6.

[13] S. Giancola, M. Valenti, and R. Sala, "A survey on 3d cameras: Metrological comparison of time-of-flight, structured-light and active stereoscopy technologies," in *SpringerBriefs in Computer Science*, ser. SpringerBriefs in Computer Science. Cham: Springer International Publishing, 2018.

[14] R. Girshick, "Fast r-cnn," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 1440–1448.

[15] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[16] H. Hirsch-Kreinsen, "Digitization of industrial work: development paths and prospects," *Journal for Labour Market Research*, vol. 49, no. 1, pp. 1–14, July 2016.

[17] D. Holz and S. Behnke, "Fast edge-based detection and localization of transport boxes and pallets in rgb-d images for mobile robot bin picking," in *Proceedings of ISR 2016: 47st International Symposium on Robotics*, June 2016, pp. 1–8.

[18] Intel Corporation. Intel realsense depth camera d435. [Online]. Available: https://click.intel.com/intelr-realsensetm-depth-camera-d435.html

[19] L. Keselman, J. I. Woodfill, A. Grunnet-Jepsen, and A. Bhowmik, "Intel realsense stereoscopic depth cameras," *Computer Vision and Pattern Recognition*, 2017.

[20] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

[21] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, Jan. 2015.

[22] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," in *Robotics: Science and Systems*, July 2017.

[23] W. Meeussen, "Coordinate frames for mobile platforms," 2010. [Online]. Available: http://www.ros.org/reps/rep-0105.html

[24] C. Mitash, K. E. Bekris, and A. Boularias, "A self-supervised learning system for object detection using physics simulation and multi-view pose estimation," in *IEEE International Conference on Intelligent Robots and Systems*, Vancouver, Canada, Sept. 2017, pp. 545–551.

[25] Mobile Industrial Robots, *MiR100 User Guide*, Mobile Industrial Robots, 2019.

[26] Mobile Industrial Robots ApS. Mir100 - mobile industrial robots. [Online]. Available: http://www.mobile-industrial-robots.com/en/products/mir100

[27] D. Pavlichenko, G. M. García, S. Koo, and S. Behnke, "Kittingbot: A mobile manipulation robot for collaborative kitting in automotive logistics," in *Intelligent Autonomous Systems 15 - Proceedings of the 15th International Conference IAS-15, Baden-Baden, Germany, June 11-15, 2018*, 2018, pp. 849–864.

[28] L. Pinto and A. Gupta, "Supersizing self-supervision : Learning to grasp from 50k tries and 700 robot hours," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2016, pp. 3406–3413.

[29] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *International Conference on Robotics and Automation (ICRA), workshop on open source software*, 2009.

[30] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 1316–1322.

[31] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2016, pp. 779–788.

[32] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv*, 2018.

[33] M. A. Roa, D. Berenson, and W. Huang, "Mobile manipulation: Toward smart manufacturing [tc spotlight]," *IEEE Robotics Automation Magazine*, vol. 22, no. 4, pp. 14–15, Dec. 2015.

[34] Robotiq. 2f-85 gripper. [Online]. Available: https://robotiq.com/products/2f85-140-adaptive-robot-gripper

[35] J. Sturm, *Approaches to Probabilistic Model Learning for Mobile Manipulation Robots*, ser. Springer Tracts in Advanced Robotics (STAR), B. Siciliano and O. Khatib, Eds. Springer, 2013.

[36] I. A. Sucan and S. Chitta. Moveit! [Online]. Available: http://moveit.ros.org

[37] Universal Robots A/S. Ur5 collaborative robot arm. [Online]. Available: https://www.universal-robots.com/products/ur5-robot

[38] Vecow. Ecs-9200/9100 gtx1050. [Online]. Available: http://www.vecow.com/dispUploadBox/PJ-VECOW/Files/3238.pdf

[39] U. Viereck, A. ten Pas, K. Saenko, and R. Platt, "Learning a visuomotor controller for real world robotic grasping using simulated depth images," *CoRR*, vol. abs/1706.04652, 2017.

[40] J. Wallén, "The history of the industrial robot," *Technical report from Automatic Control at Linköpings universitet*, 2008.

[41] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "PoseCNN: A convolutional neural network for 6d object pose estimation in cluttered scenes," *Robotics: Science and Systems (RSS)*, 2018.

[42] S. Xie, E. Haemmerle, Y. Cheng, and P. Gamage, "Vision-guided robot control for 3d object recognition and manipulation," in *Robot Manipulators*. InTech, Sept. 2008, pp. 521–546.

Dynamic parameter identification of the Universal Robots UR5

Nemanja Kovincic[1], Andreas Müller[1], Huber Gattringer[1],
Matthias Weyrer[2], Andreas Schlotzhauer[2] and Mathias Brandstötter[2]

***Abstract*— In this paper, methodology for parameter identification of an industrial serial robot manipulator is shown. The presented methodology relies on the fact that any mechanical system can be written in form linear with respect to some set of parameters. Based on experimental measurements done on the Universal Robots UR5, the presented methodology is applied and the dynamical parameters of the robot are determined in two ways. First by use of the Moore-Penrose pseudoinverse, and then by use of optimization. At the end, the ability of the determined parameters to predict measurements other than the ones used for the identification is shown.**

I. INTRODUCTION

Mathematical model of a real physical system is as good as it can predict what experiments show. In order to have a good model both its structure, meaning taking into account all relevant dynamics, and its parameters must be correct. Some model parameters, like masses and lengths of robot links, can be measured, while others, such as temperature dependent dry and viscous friction, axial and centrifugal moments of inertia or position of center of mass of segments, are almost always unknown and must be identified. However, each parameter can not be separately identified but only linear combinations of them. The vector whose elements are linear combination of parameters that can be identified is called vector of identifiable parameters or vector of base parameters.

In this paper, procedure for determination of base parameters and for their identification is explained. Then, using experimental measurements, the procedure is applied to parameter identification of the Universal Robots UR5 manipulator. At the end, in order to validate the obtained parameters, they are used for predictions of experimental measurements not used for the identification.

II. MATHEMATICAL MODELING

A. Robot dynamics

Differential equations of motion describing dynamics of a serial robot consisting of N rigid bodies can be written in a well known form as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{c}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + \mathbf{Q}_R(\dot{\mathbf{q}}) = \mathbf{Q}_M, \tag{1}$$

where $\mathbf{q} \in \mathbb{R}^N$ denotes vector of generalized coordinates, dot over the symbol stands for the derivative with respect to time, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{N,N}$ denotes symmetric positive definite mass matrix, $\mathbf{c}(\mathbf{q},\dot{\mathbf{q}}) \in \mathbb{R}^N$ stands for vector of centripetal and Coriolis terms, and $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^N$ denotes gravity vector. Furthermore, vector $\mathbf{Q}_R \in \mathbb{R}^N$ stands for friction forces while $\mathbf{Q}_M \in \mathbb{R}^N$ denotes torques acting on bodies, i.e active or control torques.

B. Friction model

Dissipative forces is the model are assumed in the form of Coulomb's dry and viscous friction, leading to

$$Q_{R_i} = r_{v_i}\dot{q}_i + r_{c_i} sign(\dot{q}_i)\ , \quad i = 1 \ldots N, \tag{2}$$

where r_{v_i} and r_{c_i} are respectively coefficients of viscous and dry friction. In order to avoid non-smooth function in the model, sign function is approximated with tangent hyperbolic function as

$$sign(\dot{q}_i) \approx \tanh(\frac{\dot{q}_i}{\varepsilon}), \tag{3}$$

where ε is very small number chosen to make slope of the tangent hyperbolic function very steep around zero.

C. Motor and gearbox dynamics

Assuming that at each joint, a motor and a gearbox are located leads to motor dynamics in the form

$$i_{G,i}^2 C_{M,i}\ddot{q}_i = i_{G,i} M_{Mot,i} = Q_{M,i}, \quad i = 1 \ldots N, \tag{4}$$

where, $C_{M,i}$ stand for the rotors axial moment of inertia corresponding to rotation axis while $M_{Mot,i}$ stands for the motor torque. Note that the previous equations can be divided by $i_{G,i}$, however between a motor and a body is the gearbox, thus torque $Q_{M,i}$, acting on body i, is $i_{G,i}$ times greater than the motor torque. Also, note that although rotor in a motor rotates around an axis that is itself in motion and thus making rotors motion complex in the parallel sense, dynamics of a motor and gearbox is taken in a much simplified form. Namely, assuming known gear ratio $i_{G,i}$, rotor of a motor driving body i spins around the joint axis with angular velocity $i_{G,i}$ times greater than relative angular velocity of the corresponding bodies. Since this rotation is dominant compared to the motion of the joint axis itself, only it is taken into account.

III. METHODOLOGY FOR IDENTIFICATION OF DYNAMICAL PARAMETERS

Methodology for identification of robot parameters is based of the fact that the equations describing motion of a system of rigid bodies can be written in linear form with respect to some set of dynamical parameters, see [2], [3]. For an overview on robot dynamic parameter identification see [10].

*This work was supported by FFG project Dr.KORS (Project number: 864892)

[1]Insitute of Robotics, Johannes Kepler University Linz nemanja.kovincic@jku.at

[2]JOANNEUM RESEARCH Forschungsgesellschaft mbH, ROBOTICS – Institut für Robotik und Mechatronik mathias.brandstoetter@joanneum.at

A. Parameter linear form of the equations of motion

Having the previous in mind, (1) is written as

$$\sum_{i=1}^{N}\Theta_{Ti}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\mathbf{p}_{Ti}=\Theta_T(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\mathbf{p}_T=\mathbf{Q}^o, \qquad \Theta_T\in\mathbb{R}^{N,10N},\ \mathbf{p}_T\in\mathbb{R}^{10N}, \tag{5}$$

where

$$\Theta_{Ti}=\mathbf{F}_{K_i}^T\times \left[\begin{array}{c|c|c} (\dot{\mathbf{v}}_K+\tilde{\omega}\mathbf{v}_K-\mathbf{g}) & (\dot{\tilde{\omega}}+\tilde{\omega}\tilde{\omega}) & 0 \\ \hline 0 & -(\dot{\mathbf{v}}_K+\tilde{\omega}\mathbf{v}_K-\mathbf{g})^{\sim} & (\dot{\Omega}+\tilde{\omega}\Omega\ \mid\ -\dot{\hat{\Omega}}-\tilde{\omega}\hat{\Omega}), \end{array}\right],$$

$$\mathbf{F}_{K_i}=\left[(\frac{\partial_K\mathbf{v}_K}{\partial\dot{\mathbf{q}}})^T\ \ (\frac{\partial_K\omega_{IK}}{\partial\dot{\mathbf{q}}})^T\right]_i^T\in\mathbb{R}^{6,N}. \tag{6}$$

For the derivation of the previous equation see [7].

Parameter vector $\mathbf{p}_{Ti}$ is

$$\mathbf{p}_{Ti}=\left(m,m\rho_{Sx},m\rho_{Sy},m\rho_{Sz},A,B,C,D,E,F\right)_i^T\in\mathbb{R}^{10}, \tag{7}$$

where ρ_{Sx}, ρ_{Sy} and ρ_{Sz} are projections of the center of mass of body i onto x, y and z axes of the coordinate frame positioned, and rigidly connected, to the joint of that body and whose one axis is the rotation axis of that body. In the same coordinate system, moments of inertia of i-th body are denoted as A,B,C,D,E,F. Furthermore, in (6) matrices Ω and $\hat{\Omega}$ stand for

$$\mathbf{J}_{K\,K}\omega_{IK}=\begin{bmatrix}A&-F&-E\\-F&B&-D\\-E&-D&C\end{bmatrix}\begin{pmatrix}\omega_x\\\omega_y\\\omega_z\end{pmatrix}=\left[\Omega\ \middle|\ \hat{\Omega}\right]\begin{pmatrix}A\\B\\C\\D\\E\\F\end{pmatrix}, \tag{8}$$

$$\Omega=\begin{bmatrix}\omega_x&0&0\\0&\omega_y&0\\0&0&\omega_z\end{bmatrix},\quad \hat{\Omega}=\begin{bmatrix}0&\omega_z&\omega_y\\\omega_z&0&\omega_x\\\omega_y&\omega_x&0\end{bmatrix}, \tag{9}$$

vector $\mathbf{g}$ denotes acceleration vector of gravity, and $\tilde{(\cdot)}$ is a skew-symmetric matrix corresponding to a vector $(\cdot)$. Note that the inertia matrix $\mathbf{J}_K$ and all vectors in (6) are written in the body coordinate frames positioned at joints. Vector $\mathbf{Q}^o$ in (5), in the absence of motor dynamics and friction, denotes vector of body torques, while for the case of friction and motor dynamics is defined in what follows.

B. Parameter linear form of the motor dynamics and friction forces

Differential equations (4) describing motor dynamics is written in parameter linear form as

$$\mathbf{Q}_M=\left[diag(\ddot{q}_i)\right]\begin{pmatrix}i_{G,1}^2C_{M,1}\\\vdots\\i_{G,N}^2C_{M,N}\end{pmatrix}=\Theta_{TM}\,\mathbf{p}_{TM}, \tag{10}$$

where vector of parameters is

$$\mathbf{p}_{TM}=\begin{pmatrix}i_{G,1}^2C_{M,1}\\\vdots\\i_{G,N}^2C_{M,N}\end{pmatrix}. \tag{11}$$

Dissipative forces defined in (2) are written in parameter linear form as

$$\mathbf{Q}_R=\left[\ diag(\dot{q}_i)\ \middle|\ diag(sign(\dot{q}_i))\ \right]\begin{pmatrix}r_{v_1}\\\vdots\\r_{c_N}\end{pmatrix}=\Theta_R\,\mathbf{p}_R, \tag{12}$$

where $diag(\cdot)$ denotes diagonal matrix, and where parameter vector is

$$\mathbf{p}_R=\begin{pmatrix}r_{v_1}\\\vdots\\r_{c_N}\end{pmatrix}. \tag{13}$$

C. Parameter linear form of the equations describing the whole system

When equations describing all element of the model, i.e. rigid bodies, motors and friction, are written in parameter linear form, writing the same form of equations describing the system in whole is very easy. Namely, combining (5), (10) and (12), linear form of equations describing the whole system is

$$\left[\ \Theta_T\quad\Theta_{TM}\quad\Theta_R\ \right]\begin{pmatrix}\mathbf{p}_T\\\mathbf{p}_{TM}\\\mathbf{p}_R\end{pmatrix}=\Theta\,\mathbf{p}=\mathbf{Q}_M, \qquad \Theta\in\mathbb{R}^{N,13N},\ \mathbf{p}\in\mathbb{R}^{13N},\ \mathbf{Q}_M\in\mathbb{R}^N, \tag{14}$$

where matrix Θ is known as the regressor matrix of the system. From the previous equations vector $\mathbf{Q}^o$ from (5) is

$$\mathbf{Q}^o=\mathbf{Q}_M-\Theta_R\,\mathbf{p}_R-\Theta_{TM}\mathbf{p}_{TM}. \tag{15}$$

D. Determination of the base parameters

Before determination of the base parameters, zero columns in the regressor matrix are identified and eliminated. Namely, in the regressor matrix defined in (5), the most general type of rigid body motion, i.e. translation plus rotation, is assumed for every body in the kinematic chain. However, when it comes to robot manipulators, the motion of the first segment in chain can be described as pure rotation around an axis. Thus, only columns in the regressor corresponding to the moments of inertia related to the axis of rotation in parameter vector (7) are not equal to zero. All other columns in the regressor matrix for the first body in chain are equal to zero. Note that if the coordinate frame, located at joint axis of the second body in chain, is positioned in such a way that the velocity of its origin is always equal to zero, then the projection of the center of mass of that body, on the axis of rotation can not be identified. However, this can be easily avoided by moving that frame along the axis of rotation.

Computation of the base parameters is based on determination of independent columns of the regressor matrix Θ by use of the QR decomposition. This procedure is explained in details in [5], Appendix 5. Here it is assumed that the base parameters and the corresponding independent columns are determined. Thus, (14) can be written as

$$\Theta(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\,\mathbf{p}=\Theta_B(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\,\mathbf{p}_B=\mathbf{Q}_M, \qquad \Theta_B\in\mathbb{R}^{N,b},\ \mathbf{p}_B\in\mathbb{R}^b,\ \mathbf{Q}_M\in\mathbb{R}^N \tag{16}$$

where Θ_B is the new regressor with all columns being independent, and where $\mathbf{p}_B$ is the vector of base parameters. Note that the system of equations (45), with N equations and $13N$ unknowns, is undetermined and thus have none or infinitely many solution. Also note that the elimination of zero columns is not necessary because when calculating the base parameters, the parameters corresponding to zero columns are not present. However, elimination of zero columns is the standard procedure in the determination of base parameters.

In order to determine the base parameters (16), the real system is excited with specially chosen excitation trajectory and the generalized coordinates and the motor torques are measured at m time instances. From the generalized coordinates, the generalized velocities and accelerations are calculated using filtering and than the new regressor, called information matrix, is formed as

$$\begin{pmatrix} \Theta_B|_{t_1} \\ \vdots \\ \Theta_B|_{t_m} \end{pmatrix} \mathbf{p}_B = \begin{pmatrix} \mathbf{Q}_M|_{t_1} \\ \vdots \\ \mathbf{Q}_M|_{t_m} \end{pmatrix} + \mathbf{r}_n, \tag{17}$$

or written in a simpler form as

$$\overline{\Theta}_B \mathbf{p}_B = \overline{\mathbf{Q}}_M + \mathbf{r}_n, \tag{18}$$

where $\mathbf{r}_n$ is the residual error vector. Now instead of an undetermined, an over determined system of equations is obtained. This system usually does not admit a solution or it can be found only for some special cases. However, an approximate solution of the problem can be found by solving least squares problem

$$\min_{\mathbf{p}_B} \| \frac{1}{2} \mathbf{e}^T \mathbf{e} \|, \quad \mathbf{e} = \overline{\Theta}_B \mathbf{p}_B - \overline{\mathbf{Q}}_M. \tag{19}$$

where the solution is

$$\begin{aligned} \left\{ \frac{\partial}{\partial \mathbf{p}} \left[\frac{\mathbf{e}^T \mathbf{e}}{2} \right] \right\}^T &= \overline{\Theta}_B^T \overline{\Theta}_B \mathbf{p}_B - \overline{\Theta}_B^T \overline{\mathbf{Q}}_M = 0 \\ \implies \mathbf{p}_B &= \left[\overline{\Theta}_B^T \overline{\Theta}_B \right]^{-1} \overline{\Theta}_B^T \overline{\mathbf{Q}}_M, \end{aligned} \tag{20}$$

provided that the matrix $\left[\overline{\Theta}_B^T \overline{\Theta}_B \right]^{-1}$ exists, i.e. if $\overline{\Theta}_B$ has full column rank. Since the matrix $\overline{\Theta}_B$ has linearly independent columns it is a full rank matrix. Note that the matrix $\left[\overline{\Theta}_B^T \overline{\Theta}_B \right]^{-1} \overline{\Theta}_B^T$ is a pseudo inverse of the matrix $\overline{\Theta}_B$, or more precisely the left Moore-Penrose inverse. Instead of using the pseudo inverse, the minimization problem (19) can also be solved using direct numerical optimization.

Assuming that the matrix $\overline{\Theta}_B$ is deterministic and that ρ_n is zero mean additive independent noise, the standard deviation of the i -th parameter is, see [5],

$$\sigma_i = \sqrt{(\left[\overline{\Theta}_B^T \overline{\Theta}_B \right]^{-1})_{i,i}}. \tag{21}$$

If the standard deviation of a parameter is big, then parameter is considered to be poorly identified.

In order to quantify how good calculated base parameters predict measured torques, normalized error

$$e_N = \frac{1}{m} \sqrt{\mathbf{e}^T \mathbf{e}}, \tag{22}$$

is used, where m stands for the number of time samples used for the calculation of the information matrix. Assuming that all degrees of freedom are rotational, the unit of this error is the newton meter.

Here, it is important to note that (19) and (22) have sense only is all degrees of freedom are of the same type, e.g. rotational. Otherwise, dimensionless quantities must be introduced first and only then (19) and (22) have sense.

Finally, note that the good approximate solution of Eq. (19) can be found only if the excitation trajectory excites all dynamical parameters of the robot. Determination of such trajectory is the subject of the next subsection.

E. Determination of the identification trajectory

The identification trajectory that excites all dynamic parameters, and thus yields good approximate solution for the parameter identification problem (19), is usually called the persistent excitation trajectory. The term "persistent" means that all parameters must be excited persistently throughout time, that is, on every time interval during the identification process. There are various criteria for calculating persistent excitation, see [8], [1], [4]. However, one of the most used is the condition number of the matrix $\Lambda = \overline{\Theta}_B^T \overline{\Theta}_B$ because it measures the sensitivity of the solution of the least squares problem to the modeling errors and noise. Thus, "good" excitation trajectory is the one whose points in time give a small condition number of the matrix Λ. Several condition number based criteria for calculating the persistent excitation exist in the literature, see [5]. Here, the criteria

$$\min_{\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}} \quad cond(\Lambda(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})) = \frac{\sigma_{\max}}{\sigma_{\min}} \geq 1 \tag{23}$$

is used where $\sigma_{\max}$ and $\sigma_{\min}$ denote the maximum and the minimum singular value of the matrix Λ, respectively. Since real physical robot cannot achieve arbitrary values of coordinates, velocities and accelerations, the previous minimization problem is solved together with constrains

$$\begin{aligned} \mathbf{q}_{\min} \leq \mathbf{q} \leq \mathbf{q}_{\max}, \\ |\dot{\mathbf{q}}| \leq \dot{\mathbf{q}}_{\max}, \\ |\ddot{\mathbf{q}}| \leq \ddot{\mathbf{q}}_{\max}, \end{aligned} \tag{24}$$

where the vectors $\mathbf{q}_{\min}$ and $\mathbf{q}_{\max}$ denote minimal and maximal allowed values of the generalized coordinates, the vector $\dot{\mathbf{q}}_{\max}$ stands for maximal generalized velocities and the vector $\ddot{\mathbf{q}}_{\max}$ denotes maximal allowed generalized accelerations. If the robot can self collide during motion, than also the requirement that there is no self collision is used as the constraint. Besides condition number, determinant of the matrix Λ can also be used for calculating persistent excitation, see [4].

In order to solve the minimization problem (23) together with constrains (24), following [9] the minimization trajectory will be taken in form of a finite Fourier series as

$$q_i(t) = \sum_{l=1}^{L_i} \left(\frac{a_{i,l}}{\omega_i l} \sin(\omega_i l t) - \frac{b_{i,l}}{\omega_i l} \cos(\omega_i l t) \right) + q_{i,0}, \tag{25}$$

where L_i is the order of the series, ω_i is the base frequency, $q_{i,0}$ is the coordinate offset, and $a_{i,l}$ and $b_{i,l}$ are coefficients of the series. In the general case, all constants in the previous equation can be used as optimization variables. However, usually the order of the series is fixed and the rest variables are used in optimization. With the Fourier series representation the infinite-dimensional optimization problem (23) is substituted with finite dimensional one given as

$$\min_{\mathbf{a},\mathbf{b},\omega,\mathbf{q}_0} \quad cond(\Lambda(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})) \tag{26}$$

where

$$\begin{aligned} \mathbf{a} &= \left[a_{1,1}\dots a_{1,L_i}\dots a_{N,L_i}\right]^T \\ \mathbf{b} &= \left[b_{1,1}\dots b_{1,L_i}\dots b_{N,L_i}\right]^T \\ \omega &= \left[\omega_1\dots\omega_N\right]^T \\ \mathbf{q}_0 &= \left[q_{1,0}\dots q_{N,0}\right]^T, \end{aligned} \tag{27}$$

which is again solved together with the constrains (24) and the condition that there is no self collision of the robot. Finally, instead of optimizing all previously mentioned variables, for example the coordinate offset $\mathbf{q}_0$ can be predefined or the basic frequency ω_i can be the same for all bodies. This lowers the dimension of solution of the problem and thus also the time needed for optimization algorithm to find the solution.

IV. Universal robots UR5

As an example for demonstrating the previously described methodology for parameter identification, the Universal Robots UR5 manipulator is used, see Fig. 1. This manipulator has six degrees of freedom and is a lightweight collaborative robot.

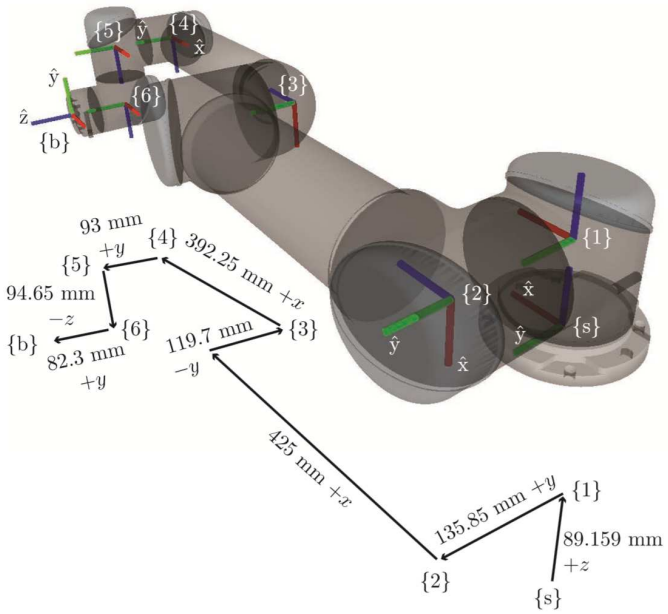


Fig. 1. Universal Robots UR5, taken from [6].

In Fig. 1, UR5 robot is shown at initial configuration, together with the coordinate systems of interest and distances between them. In what follows, the relative position and orientation of these coordinate frames at initial configuration are described. Furthermore, all generalized coordinates to be introduced are measured in positive mathematical directions, relative to the previous body in the kinematic chain, and given in radians.

Inertial frame of reference is denoted as K_sxyz, where x, y and z axis are shown as red, green and blue axis, respectively. Translating the inertial frame along its z axis, for value $l_{1,z}$, coordinate frame K_1xyz attached to the first segment is obtained. The orientation of these two coordinate frames is the same at initial configuration, while an arbitrary orientation of the first body is achieved by rotating it in positive mathematical direction around K_1z axis, where the generalized coordinate describing this rotation is denoted as $q_1 = q_1(t)$. Further, coordinate frame K_2xyz is obtained by translating the frame K_1xyz along K_1y axis, for value $l_{2,y}$, and then by rotating it in the positive mathematical direction around the same axis for angle $\pi/2$. An arbitrary orientation of the second body, with respect to the first, is obtained by rotation around K_2y axis for angle $q_2 = q_2(t)$. Next, translating the coordinate frame K_2xyz along K_2y and K_2z axis for values $l_{3,y}$ and $l_{3,z}$, respectively, coordinate frame K_3xyz is obtained. Rotating the third body around K_3z axis, again in the positive mathematical direction, its arbitrary orientation is achieved. This rotation is described with generalized coordinate $q_3 = q_3(t)$. The coordinate frame K_4xyz is obtained by translating frame K_3xyz along K_3z axis for value $l_{4,z}$ and by rotating it in the positive mathematical direction around K_3y axis for angle $\pi/2$. An arbitrary orientation of the fourth body is achieved by rotation around K_4y for angle $q_4 = q_4(t)$. For obtaining coordinate frame K_5xyz attached to the fifth body in the kinematic chain, coordinate frame K_4xyz is translated along the K_4y axis for value $l_{5,y}$. By rotating the fifth body around K_5z axis, its arbitrary orientation is obtained, where the generalized coordinate describing that rotation is $q_5 = q_5(t)$. Translating the coordinate frame K_5xyz along K_5z axis for value $l_{6,z}$, coordinate frame K_6xyz is obtained. An arbitrary orientation of the sixths body is achieved by rotation around K_6y axis, and that rotation is described by the generalized coordinate $q_6 = q_6(t)$. Finally, the coordinate frame K_7xyz, positioned at the end effector is obtained by translating frame K_6xyz along K_6y for value $l_{7,y}$ and than rotating it in the negative mathematical direction for angle $\pi/2$.

With the previously introduced generalized coordinates,

the orthogonal transformation matrices are

$$\begin{aligned}
&\mathbf{R}_{K_s,K_1} = \mathbf{R}_z(q_1),\\
&\mathbf{R}_{K_1,K_2} = \mathbf{R}_y(\frac{\pi}{2})\mathbf{R}_y(q_2) = \mathbf{R}_y(\frac{\pi}{2}+q_2),\\
&\mathbf{R}_{K_2,K_3} = \mathbf{R}_y(q_3),\\
&\mathbf{R}_{K_3,K_4} = \mathbf{R}_y(\frac{\pi}{2})\mathbf{R}_y(q_4) = \mathbf{R}_y(\frac{\pi}{2}+q_4),\\
&\mathbf{R}_{K_4,K_5} = \mathbf{R}_z(q_5),\\
&\mathbf{R}_{K_5,K_6} = \mathbf{R}_y(q_6),\\
&\mathbf{R}_{K_6,K_7} = \mathbf{R}_x(-\frac{\pi}{2}),
\end{aligned} \tag{28}$$

where the rotation matrices corresponding to rotation around x, y and z axis, are

$$\mathbf{R}_x(q_i) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(q_i) & -\sin(q_i) \\ 0 & \sin(q_i) & \cos(q_i) \end{pmatrix}, \tag{29}$$

$$\mathbf{R}_y(q_i) = \begin{pmatrix} \cos(q_i) & 0 & sin(q_i) \\ 0 & 1 & 0 \\ -\sin(q_i) & 0 & \cos(q_i) \end{pmatrix} \tag{30}$$

and

$$\mathbf{R}_z(q_i) = \begin{pmatrix} \cos(q_i) & -sin(q_i) & 0 \\ \sin(q_i) & \cos(q_i) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{31}$$

A. Kinematic analysis

Starting from the angular velocity of the first body and following the recursive formulation, angular velocities of the bodies in the kinematic chain are

$${}_{K_i}\boldsymbol{\omega}_{I,K_i} = \mathbf{R}_{K_i,K_{i-1}}\ {}_{K_{i-1}}\boldsymbol{\omega}_{I,K_{i-1}} + {}_{K_i}\boldsymbol{\omega}_{K_{i-1},K_i}\ , \quad i = 2\ldots 6, \tag{32}$$

where

$$\begin{aligned}
{}_{K_1}\boldsymbol{\omega}_{I,K_1} &= \begin{bmatrix} 0 & 0 & \dot{q}_1 \end{bmatrix}^T,\\
{}_{K_2}\boldsymbol{\omega}_{K_1,K_2} &= \begin{bmatrix} 0 & \dot{q}_2 & 0 \end{bmatrix}^T,\\
{}_{K_3}\boldsymbol{\omega}_{K_2,K_3} &= \begin{bmatrix} 0 & \dot{q}_3 & 0 \end{bmatrix}^T,\\
{}_{K_4}\boldsymbol{\omega}_{K_3,K_4} &= \begin{bmatrix} 0 & \dot{q}_4 & 0 \end{bmatrix}^T,\\
{}_{K_5}\boldsymbol{\omega}_{K_4,K_5} &= \begin{bmatrix} 0 & 0 & \dot{q}_5 \end{bmatrix}^T,\\
{}_{K_6}\boldsymbol{\omega}_{K_5,K_6} &= \begin{bmatrix} 0 & \dot{q}_6 & 0 \end{bmatrix}^T.
\end{aligned} \tag{33}$$

With the angular velocities defined, velocity of the origin of frame $K_i xyz$, written in that frame, is

$${}_{K_i}\mathbf{v}_{K_i} = \mathbf{R}_{K_i,K_{i-1}}\ {}_{K_{i-1}}\mathbf{v}_{K_i}, \quad i = 1\ldots 6 \tag{34}$$

where

$$\begin{aligned}
&{}_{K_0}\mathbf{v}_{K_1} = \mathbf{0},\\
&{}_{K_i}\mathbf{v}_{K_{i+1}} = \mathbf{R}_{K_i,K_{i-1}}\ {}_{K_{i-1}}\mathbf{v}_{K_i} + {}_{K_i}\tilde{\boldsymbol{\omega}}_{I,K_i}\ {}_{K_i}\mathbf{r}_{K_i,K_{i+1}}\ , \quad i = 1\ldots 5;
\end{aligned} \tag{35}$$

and where the index 0 denotes inertial frame of reference. In the previous equations, vectors ${}_{K_i}r_{K_i,K_{i+1}}$, $i = 1\ldots 6$ are

$$\begin{aligned}
{}_{K_1}r_{K_1,K_2} &= \begin{bmatrix} 0 & l_{2,y} & 0 \end{bmatrix}^T,\\
{}_{K_2}r_{K_2,K_3} &= \begin{bmatrix} 0 & l_{3,y} & l_{3,z} \end{bmatrix}^T,\\
{}_{K_3}r_{K_3,K_4} &= \begin{bmatrix} 0 & 0 & l_{4,z} \end{bmatrix}^T,\\
{}_{K_4}r_{K_4,K_5} &= \begin{bmatrix} 0 & l_{5,y} & 0 \end{bmatrix}^T,\\
{}_{K_5}r_{K_5,K_6} &= \begin{bmatrix} 0 & 0 & l_{6,z} \end{bmatrix}^T.
\end{aligned} \tag{36}$$

Differentiating with respect to time (32), vectors of angular accelerations of bodies are obtained. Similarly, differentiating with respect to time (34), and taking into account both change of intensity and of direction, acceleration vectors of points K_i, $i = 1\ldots 6$ are obtained as

$${}_{K_i}\mathbf{a}_{K_i} = {}_{K_i}\dot{\mathbf{v}}_{K_i} + {}_{K_i}\tilde{\boldsymbol{\omega}}_{I,K_iK_i}\mathbf{v}_{K_i}. \tag{37}$$

B. Parameter linear form of the equations of motion

Since all elements for writing the parameter linear form of the equations of motion are known, in order to construct the regressor matrix, it is necessary to substitute them into (6), (10), and (12). However, the obtained analytical expression for the regressor is not shown. Instead, it will be assumed that the regressor matrix $\Theta(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ is known. Then, the parameter linear form of the equations of motion is

$$\Theta(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\,\mathbf{p} = \mathbf{Q}_M, \quad \Theta \in \mathbb{R}^{6,78}\ ,\ \mathbf{p} \in \mathbb{R}^{78},\ \mathbf{Q}_M \in \mathbb{R}^6. \tag{38}$$

where the parameter vector $\mathbf{p}$ is

$$\mathbf{p} = \begin{pmatrix} \mathbf{p}_T \\ \mathbf{p}_{TM} \\ \mathbf{p}_R \end{pmatrix} \in \mathbb{R}^{78}, \tag{39}$$

with its elements defined as

$$\begin{aligned}
\mathbf{p}_T &= \left(\mathbf{p}_{T1}\ldots\mathbf{p}_{T6}\right) \in \mathbb{R}^{60},\\
\mathbf{p}_{Ti} &= \left(m, m\rho_{Sx}, m\rho_{Sy}, m\rho_{Sz}, A, B, C, D, E, F\right)_i \in \mathbb{R}^{10}, \quad i = 1\ldots 6,\\
\mathbf{p}_{TM} &= \left(i^2_{G,1}C_{M,1}\ldots i^2_{G,6}C_{M,6}\right) \in \mathbb{R}^6,\\
\mathbf{p}_R &= \left(r_{v_1}\ldots r_{v_6}\ r_{c_1}\ldots r_{c_6}\right) \in \mathbb{R}^6,
\end{aligned} \tag{40}$$

where ρ_{Sx_i}, ρ_{Sy_i} and ρ_{Sz_i} are projections of the center of mass of body i onto the axis of the coordinate frame $K_i xyz$, respectively, i.e.

$${}_{K_i}r_{K_i,S_i} = \begin{bmatrix} \rho_{Sx_i} & \rho_{Sy_i} & \rho_{Sz_i} \end{bmatrix}^T, \quad i = 1\ldots 6. \tag{41}$$

Note that, since motion of the first body is described as pure rotation, only a column in the matrix $\Theta(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$, corresponding to axial moments of inertia for the axis of rotation is not zero. All other columns in that regressor are zero.

Substituting random values for vectors $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ in the matrix $\Theta(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$, and applying the QR decomposition to the

resulting matrix, results in the base parameter vector

$$\mathbf{p}_B = \begin{pmatrix} i_{G,1}^2 C_{M,1} + C_1 + C_2 + C_3 + C_4 + \\ +0.01285\,m_2 + 0.01191\,m_5 + \\ +0.01191\,m_6 + 0.2267\,m_2\,\rho_{Sy_2} \\ \hline m_2\,\rho_{Sx_2} \\ \hline 0.425\,m_3 + 0.425\,m_4 + 0.425\,m_5 + \\ +0.425\,m_6 + m_2\,\rho_{Sz_2} \\ \hline A_2 - C_2 + 0.1806\,m_3 + 0.1806\,m_4 + \\ +0.1806\,m_5 + 0.1806\,m_6 \\ \hline i_{G,2}^2 C_{M,2} + B_2 + 0.1806\,m_3 + 0.1806\,m_4 + \\ +0.1806\,m_5 + 0.1806\,m_6 \\ \hline D_2 - 0.04818\,m_3 - 0.04818\,m_4 - \\ -0.001789\,m_5 - 0.001789\,m_6 + \\ +0.425\,m_3\,\rho_{Sy_3} + 0.425\,m_4\,\rho_{Sy_4} \\ \hline \begin{array}{c} E_2 \\ F_2 \\ m_3\,\rho_{Sx_3} \\ 0.3922\,m_4 + 0.3922\,m_5 + 0.3922\,m_6 + m_3\,\rho_{Sz_3} \\ A_3 - C_3 + 0.1539\,m_4 + 0.1539\,m_5 + 0.1539\,m_6 \\ B_3 + 0.1539\,m_4 + 0.1539\,m_5 + 0.1539\,m_6 \\ D_3 + 0.04281\,m_5 + 0.04281\,m_6 + 0.3922\,m_4\,\rho_{Sy_4} \\ E_3 \\ F_3 \\ m_4\,\rho_{Sx_4} \\ 0.09465\,m_6 + m_4\,\rho_{Sz_4} + m_5\,\rho_{Sz_5} \\ A_4 + B_5 - C_4 + 0.008959\,m_6 \\ B_4 + B_5 + 0.008959\,m_6 \\ D_4 + 0.01033\,m_6 + 0.1092\,m_5\,\rho_{Sz_5} \\ E_4 \\ F_4 \\ m_5\,\rho_{Sx_5} \\ m_5\,\rho_{Sy_5} + m_6\,\rho_{Sy_6} \\ A_5 - B_5 + C_6 \\ C_5 + C_6 \\ D_5 + 0.09465\,m_6\,\rho_{Sy_6} \\ E_5 \\ F_5 \\ m_6\,\rho_{Sx_6} \\ m_6\,\rho_{Sz_6} \\ A_6 - C_6 \\ B_6 \\ D_6 \\ E_6 \\ F_6 \\ \left[i_{G,3}^2 C_{M,3} \dots i_{G,6}^2 C_{M,6}\right]^T \\ \left[r_{v_1} \dots r_{c_6}\right]^T \end{array} \end{pmatrix}. \tag{42}$$

Thus, the system of equations

$$\Theta(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\,\mathbf{p} = \mathbf{Q}_M, \quad \Theta \in \mathbb{R}^{6,78},\ \mathbf{p} \in \mathbb{R}^{78},\ \mathbf{Q}_M \in \mathbb{R}^6 \tag{43}$$

is substituted with the new system

$$\Theta_B(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\,\mathbf{p}_B = \mathbf{Q}_M, \quad \Theta_B \in \mathbb{R}^{6,52},\ \mathbf{p}_B \in \mathbb{R}^{52},\ \mathbf{Q}_M \in \mathbb{R}^6 \tag{44}$$

where all columns in the new regressor Θ_B are mutually independent. Note that elements of the vector $\mathbf{p}_B$ are linear combination of the model parameters. Also note that the zero columns from the regressor are not eliminated first, but the corresponding parameters are still not in the vector $\mathbf{p}_B$. They are eliminated by use of the QR decomposition.

Sometimes, some parameters are known to be zero or they are negligible compared to some other parameters. In that case one can chose not to identify them so the corresponding columns in matrix Θ are eliminated first and then the QR decomposition is applied to the resulting matrix. This results in a new base parameter vector.

In this work, for the identification of parameters of the UR5 manipulator, several parameters are assumed to be negligible. Namely, centrifugal moments of inertia of links are assumed to be much smaller that the axial moments of inertia and thus are not going to be identified. Furthermore, it is assumed that position of the center of mass of body i does not have all three projections on the axis of the coordinate frames $K_i xyz, i = 1 \dots 6$, but only one. The motion of the first body in the kinematic chain is pure rotation and thus only the axial moment of inertia corresponding to the rotation axis is identified. For the second body, it is assumed that the center of mass has projection only on the $K_2 z$ axis. Similarly, the center of mass of the third body is assumed to be on $K_3 z$ axis. For the forth and the sixth body in chain, it is assumed that the corresponding centers of mass are on $K_4 y$ and $K_6 y$ axis, respectively. Finally, for the fifth body, the center of mass is assumed to lie on the $K_5 z$ axis.

With the previous assumptions, the base parameter vector $\mathbf{p}_B \in \mathbb{R}^{33}$ is now

$$\mathbf{p}_B = \begin{pmatrix} i_{G,1}^2 C_{M,1} + C_1 + C_2 + C_3 + C_4 + 0.01285\,m_2 + \\ +0.01191\,m_5 + 0.01191\,m_6 \\ \hline \begin{array}{c} m_2\,\rho_{Sz_2} \\ A_2 - C_2 \\ i_{G,2}^2 C_{M,2} + B_2 \\ m_3 + m_4 + m_5 + m_6 \\ 0.3922\,m_4 + 0.3922\,m_5 + 0.3922\,m_6 + m_3\,\rho_{Sz_3} \\ A_3 - C_3 + 0.1539\,m_4 + 0.1539\,m_5 + 0.1539\,m_6 \\ B_3 + 0.1539\,m_4 + 0.1539\,m_5 + 0.1539\,m_6 \\ 0.1092\,m_5 + 0.1092\,m_6 + m_4\,\rho_{Sy_4} \\ A_4 + B_5 - C_4 + 0.008959\,m_6 \\ B_4 + B_5 + 0.008959\,m_6 \\ 0.09465\,m_6 + m_5\,\rho_{Sz_5} \\ A_5 - B_5 + C_6 \\ C_5 + C_6 \\ m_6\,\rho_{Sy_6} \\ A_6 - C_6 \\ B_6 \\ i_{G,3}^2 C_{M,3} \\ i_{G,4}^2 C_{M,4} \\ i_{G,5}^2 C_{M,5} \\ i_{G,6}^2 C_{M,6} \\ r_{v_1} \\ \vdots \\ r_{c_6} \end{array} \end{pmatrix}. \tag{45}$$

Note that, if some parameters are not going to be identified, the new base parameter vector is not obtained by simply substituting zeros for those parameters in the vector shown in (42). In what follows the base parameter vector (45) is going to be identified.

C. Identification results

For the identification of the base parameters, two persistent excitation trajectories are generated. One is used for parameter identification and the other one for validation of the obtained parameter vector. These trajectories are generated by solving the optimization problem (26), where the order of the series in (25) is 5, and where the offset $q_{2,0} = -\pi/2$ and all others are zero. The rest parameters in Fourier series are found by optimization. The identification is done on a time interval of 20 seconds, however, only first 10 seconds are shown in figures. In Fig. 2 measured angles of the excitation used for the parameter identification are shown, while Fig. 3 shows measured motor currents for the same trajectory.

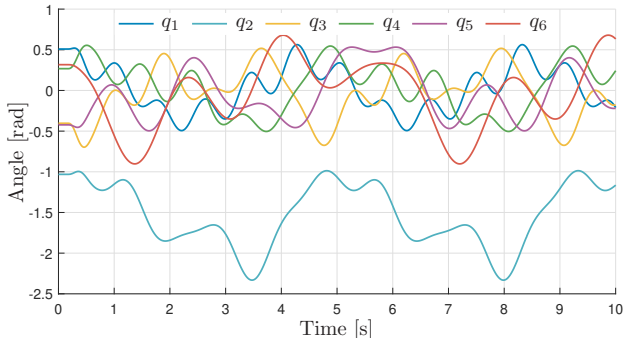


Fig. 2. Persistent excitation trajectories used for the identification

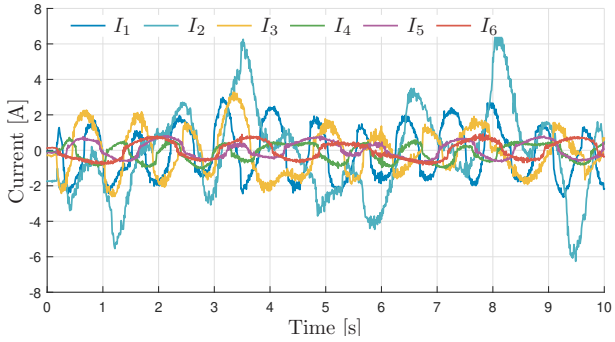


Fig. 3. Motor current

In order to calculate torques acting on bodies, each motor current is multiplied with torques constant and by gear ratio. Thus, body torques are $M_i = i_{G,i} k_i I_i$, $i = 1 \ldots 6$. On the UR5 robot, there are two types of motors, one with motor constant $k_i = 0.125$ Nm/A, $i = 1 \ldots 3$, and other with constant $k_i = 0.0922$ Nm/A, $i = 4 \ldots 6$. Also, all gears have the same gear ration, i.e. $i_G = i_{G,i} = 101$, $i = 1 \ldots 6$.

In order to form the regressor Θ_B, generalized velocities and accelerations must be calculated from the measured values of generalized coordinates. When working with the UR5 robot, generalized velocities are obtained from the controller, while generalized accelerations are calculated using filtering. The transfer function of the filter used is

$$y = \frac{s}{\frac{s}{w} + 1} u, \tag{46}$$

where s denotes the Laplace variable, $w = 2\pi f$ is angular frequency with $f = 10$ Hz being the corner frequency of the filter. The values of the corner frequency is determined by inspecting the frequency content of the measured signals. Using the filter and Matlab's "*filtfilt*" function, generalized acceleration are obtained. With the previous preparation done, one can proceed to the determination of the base parameters as described in Section III.

Following the methodology for the parameter identification, first the information matrix $\overline{\Theta}_B$ and vector $\overline{\mathbf{Q}}_M$ are formed. Then, base parameter are determined in two ways. First by using Moore-Penrose pseudoinverse from (20), and then by using numerical optimization to directly solve the optimization problem (19), together with the constraint that all base parameters are positive.

The results for the base parameters obtained by use of the pseudoinverse are shown in Fig. 4, together with the corresponding standard deviations. In Fig. 5, base parameters obtained by optimization are shown, again with the corresponding standard deviations.

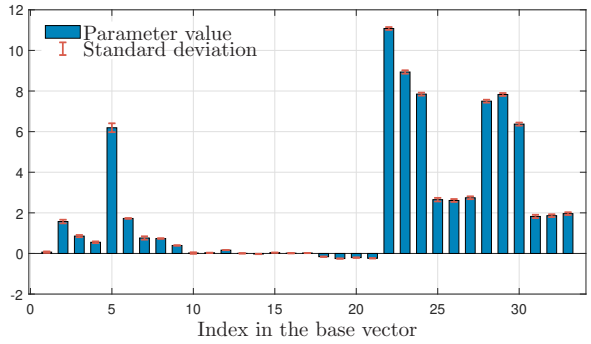


Fig. 4. Base parameters obtained by pseudoinverse

Note that the standard deviations are small, and the same in both figures.

In order to check the quality of the calculated base parameters vector, predicted torques are compared with the measured ones and the normalized error (22) is calculated. Predicted body torques, obtained using the base parameters vector obtained with the use of the pseudoinverse, are shown in Fig. 6, Fig. 7 and Fig. 8, while the normalized error reads

$$e_N = 0.0279. \tag{47}$$

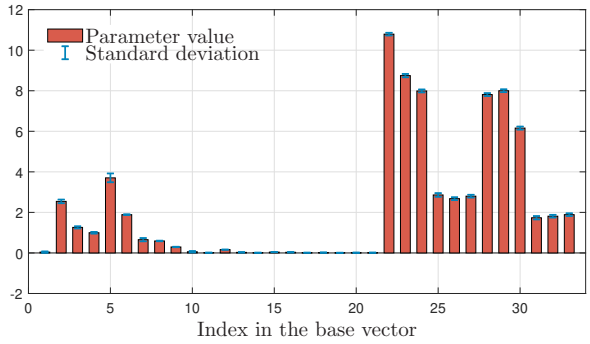


Fig. 5. Base parameters obtained by optimization

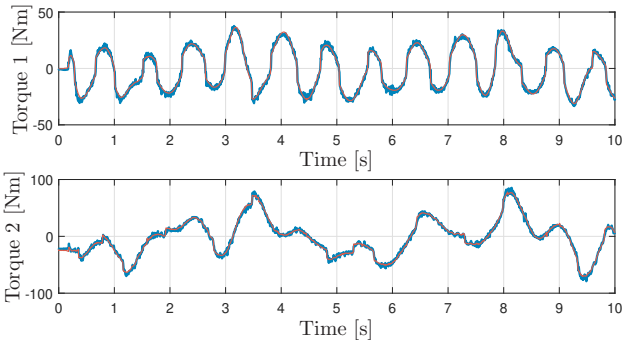


Fig. 6. Measured and predicted torques - pseudoinverse

For the parameter vector obtained using the optimization, predicted body torques are shown in Fig. 9, Fig. 10 and Fig. 11, while the normalized error for this vector is

$$e_N = 0.0301. \tag{48}$$

Next, calculated vectors of the base parameters are used for predicting torques obtained using the second excitation trajectory, shown in Fig. 12.

For the trajectory in Fig. 12, and using the base parameters obtained by pseudoinverse, prediction of torques are shown in Fig. 13, Fig. 14 and Fig. 15, while the normalized error is

$$e_N = 0.0152. \tag{49}$$

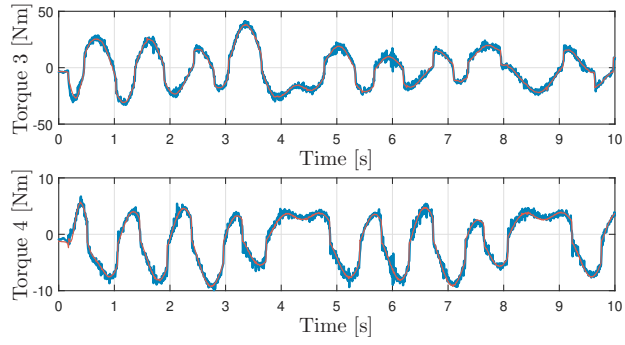


Fig. 7. Measured and predicted torques - pseudoinverse

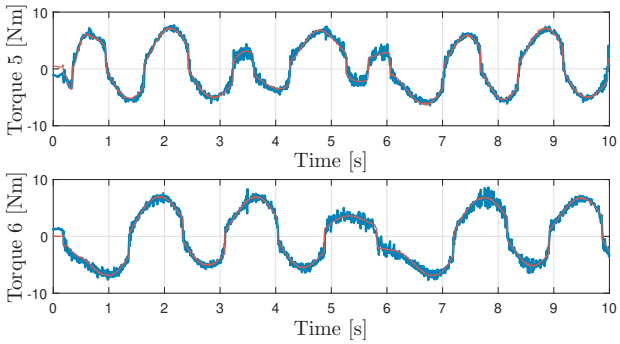


Fig. 8. Measured and predicted torques - pseudoinverse

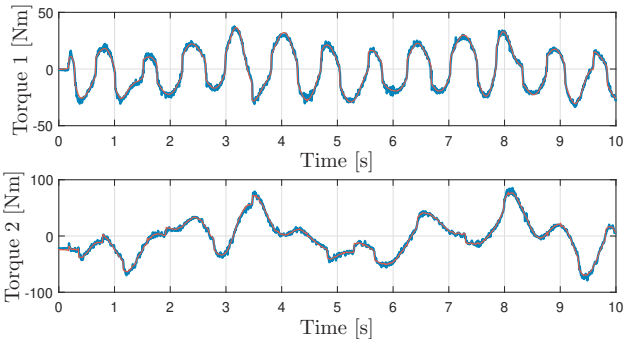


Fig. 9. Measured and predicted torques - optimization

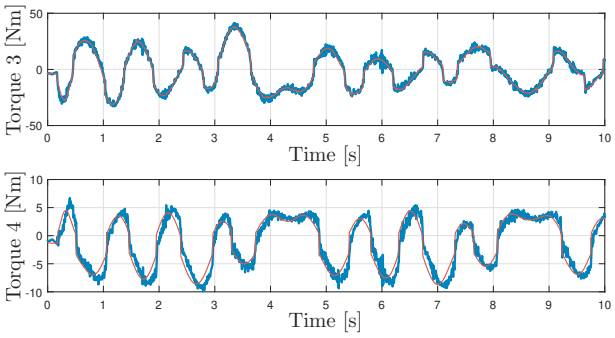


Fig. 10. Measured and predicted torques - optimization

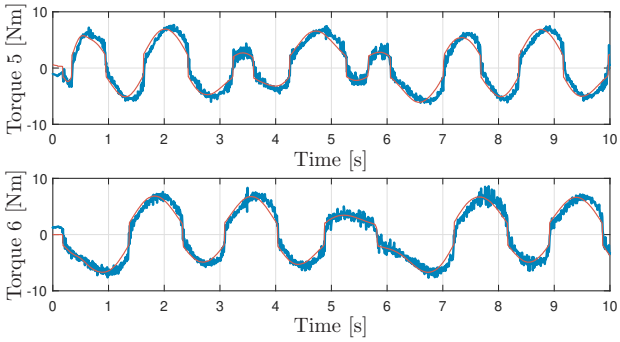


Fig. 11. Measured and predicted torques - optimization

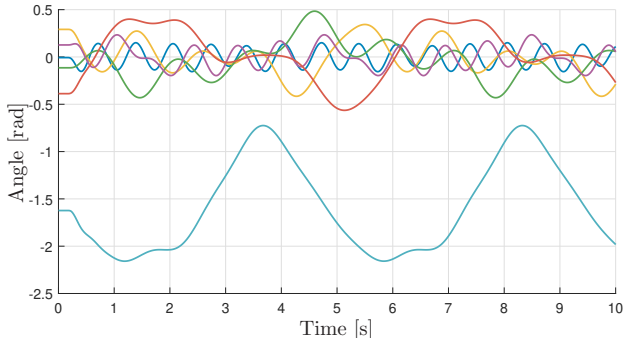


Fig. 12. Persistent excitation trajectories used for the parameter validation

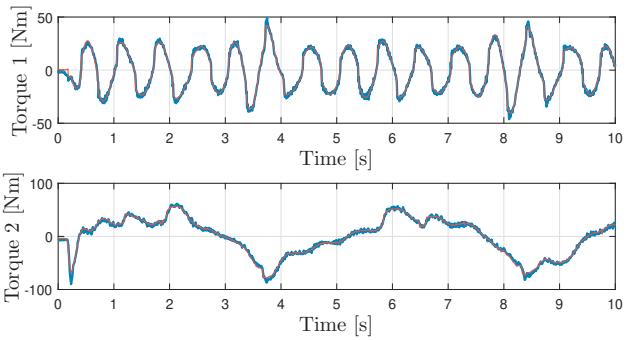


Fig. 13. Validation of the obtained base parameter vector, trajectory from Fig. 12 - pseudoinverse

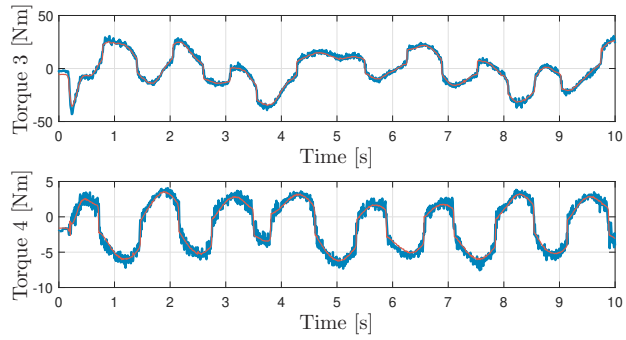


Fig. 14. Validation of the obtained base parameter vector, trajectory from Fig. 12 - pseudoinverse

Finally, torque predictions of the base parameters vector obtained by optimization, for trajectory in Fig. 12 are shown in Fig. 16, Fig. 17 and Fig. 18. The normalized error for this case is

$$e_N = 0.0163. \tag{50}$$

V. Conclusion

From the identification results several things can be seen. First, both base vectors can predict measured torques almost equally good. Although vector obtained by pseudoinverse has negative parameters corresponding to moment of inertia of the motor rotors, which is physically impossible, its torque

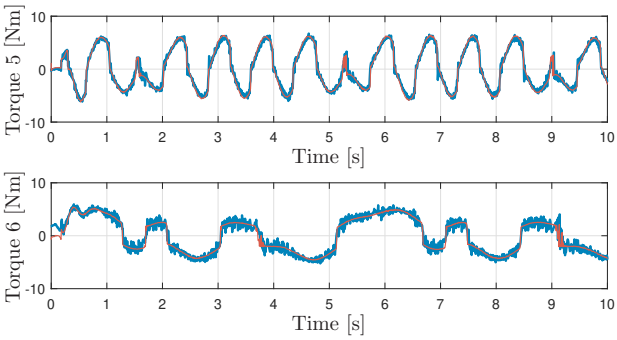


Fig. 15. Validation of the obtained base parameter vector, trajectory from Fig. 12 - pseudoinverse

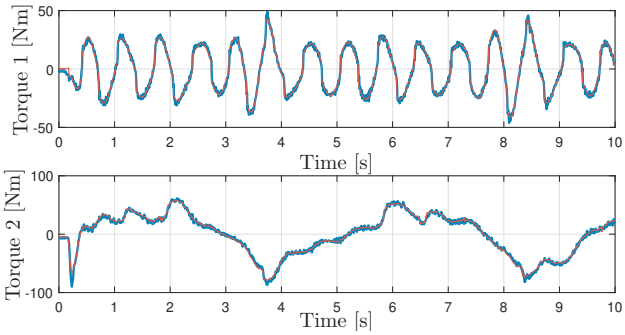


Fig. 16. Validation of the obtained base parameter vector, trajectory from Fig. 12 - optimization

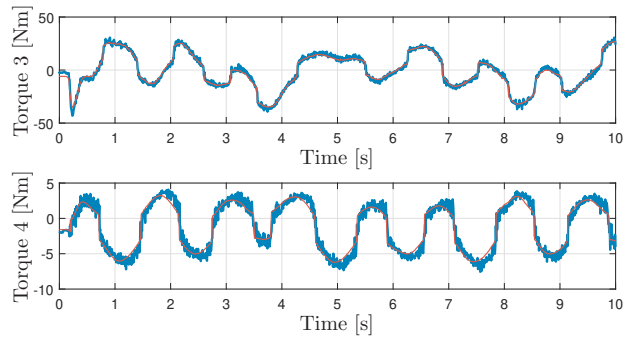


Fig. 17. Validation of the obtained base parameter vector, trajectory from Fig. 12 - optimization

predictions are a little bit better as can be seen from the corresponding normalized errors. However, the consequence of having physically impossible negative parameters is that the mass matrix is, for some robot configurations, not symmetric or negative definite and thus methods for mass matrix inversion tailored for symmetric positive definite matrices, like the Cholesky decomposition, can not be used.

At the end, note that on some figures showing torque predictions there is an error at zero time. This error is because of static friction which is greater that the here identified dynamic one.

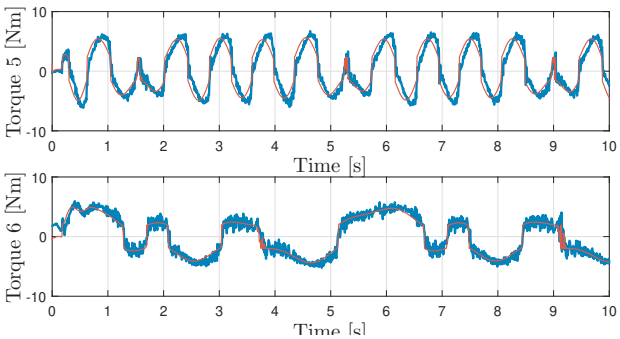


Fig. 18. Validation of the obtained base parameter vector, trajectory from Fig. 12 - optimization

ACKNOWLEDGMENT

This work was financed by DR.KORS - Dynamic reconfigurability of collaborative robot systems project, funded by FFG - Österreichische Forschungsförderungsgesellschaft, project number 864892.

REFERENCES

[1] G. Antonelli, F. Caccavale, and P. Chiacchio, "A systematic procedure for the identification of dynamic parameters of robot manipulators," *Robotica*, vol. 17, no. 4, pp. 427–435, 1999.

[2] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *The International Journal of Robotics Research*, vol. 5, no. 3, pp. 101–119, 1986.

[3] A. Codourey and E. Burdet, "A body-oriented method for finding a linear form of the dynamic equation of fully parallel robots," in *Proceedings of International Conference on Robotics and Automation*, vol. 2. IEEE, 1997, pp. 1612–1618.

[4] J. Jin and N. Gans, "Parameter identification for industrial robots with a fast and robust trajectory design approach," *Robotics and Computer-Integrated Manufacturing*, vol. 31, pp. 21–29, 2015.

[5] W. Khalil and E. Dombre, *Modeling, identification and control of robots*. Butterworth-Heinemann, 2004.

[6] K. M. Lynch and F. C. Park, *Modern Robotics: Mechanics, Planning, and Control*, 1st ed. New York, NY, USA: Cambridge University Press, 2017.

[7] M. Neubauer, H. Gattringer, and H. Bremer, "A persistent method for parameter identification of a seven-axes manipulator," *Robotica*, vol. 33, no. 5, pp. 1099–1112, 2015.

[8] C. Presse and M. Gautier, "New criteria of exciting trajectories for robot identification," in *Proceedings IEEE International Conference on Robotics and Automation*. IEEE, 1993, pp. 907–912.

[9] J. Swevers, C. Ganseman, D. B. Tukel, J. De Schutter, and H. Van Brussel, "Optimal robot excitation and identification," *IEEE transactions on robotics and automation*, vol. 13, no. 5, pp. 730–740, 1997.

[10] J. Wu, J. Wang, and Z. You, "An overview of dynamic parameter identification of robots," *Robotics and computer-integrated manufacturing*, vol. 26, no. 5, pp. 414–419, 2010.

Machine Vision for Embedded Devices: from Synthetic Object Detection to Pyramidal Stereo Matching

Daniele Evangelista[1], Marco Imperoli[2], Emanuele Menegatti[1] and Alberto Pretto[3]

***Abstract*— In this work we present an embedded and all-in-one system for machine vision in industrial settings. This system enhances the capabilities of an industrial robot providing vision and perception, e.g. deep learning based object detection and 3D reconstruction by mean of efficient and highly scalable stereo matching. To this purpose we implemented and tested innovative solutions for object detection based on synthetically trained deep networks and a novel approach for depth estimation that embeds traditional 3D stereo matching within a pyramidal framework in order to reduce the computation time. Both object detection and 3D stereo matching have been efficiently implemented on the embedded device. Results and performance of the implementations are given for publicly available datasets, in particular the T-Less dataset for texture-less object detection, Kitti Stereo and Middlebury Stereo datasets for depth estimation.**

I. INTRODUCTION

State-of-the-art industrial machine vision systems currently works with 3D sensors, sometimes coupled with a color or a gray-level camera. Traditionally, the 3D information has been acquired using *passive* stereo systems, i.e. systems composed by two or more cameras. The depth map is recovered by means of a correspondence problem: matched points projections are triangulated between pairs of sensors. Unfortunately, these systems often fail to provide an accurate 3D reconstruction for large portions of the framed scene, due to the absence of salient visual features. To overcome this limitation, *active* stereo systems have been introduced. Active vision sensors use light emitters that project a specific pattern (*Active Stereo and Structured Light* sensors) or a light with a specific wavelength (*Time-of-Flight* sensors): all these sensors modify in some way the surrounding environment (i.e., they illuminate the scene). In the first cases, the correspondence problem is solved in different ways: by performing a traditional stereo matching algorithm using visual features synthetically created by the light projector for the active stereo sensors; by searching the known pattern in the camera image (so called *pattern decoding*) for the structured light sensors.

In this work we propose an embedded and all-in-one device that integrates both active and passive stereo matching technologies. In particular, two high resolution color cameras provide the system with passive stereo capabilities, and a random pattern projector mounted at the center of the cameras baseline provides active stereo capabilities by means of structured illumination of the scene (see Fig.2 for more details). The proposed sensor also integrates a CPU and a powerful Graphical Processing Unit (GPU) specifically designed to run expansive Machine Learning algorithms (e.g. Deep Learning) and a complete Unix based Operating System. This design enables the possibility to mount the system directly on top of a robotic cell and being connected bidirectionally with the robot system without the need of any external unit. This high level of flexibility makes the system appropriate for bin picking applications where a robotic manipulator needs to be driven by a vision system to detect and accurately manipulate highly cluttered objects.

Fig. 1: The proposed sensor: renders (left column) and its realization with a really functioning prototype (right column).

To be able to perceive and accurately detect objects, vision systems rely on 2D and 3D information at the same time. For this reason we propose an efficient depth estimation method that embeds traditional 3D stereo matching techniques within a pyramidal framework in order to reduce the computation time. Moreover, on the system, we also implemented deep neural network based object detectors that were trained using synthetically generated data. This process drastically decreases the time needed for collecting data, and does not require any human intervention for annotating the data. The aforementioned perception pipeline has been tested on highly challenging task, namely texture-less objects, a very common situation in industrial settings where objects quite often do not offer any, or very poor, texture detail.

This work was supported by the European Commission under 601116-ECHORD++ (FlexSight experiment).

[1] authors are with the Department of Information Engineering (DEI), University of Padova, Italy, (evangelista, emg)@dei.unipd.it

[2] author is with the Department of Computer, Control, and Management Engineering, Sapienza University of Rome, Italy imperoli@diag.uniroma1.it

[3] author is the FlexSight s.r.l. company, Padua, Italy alberto.pretto@flexsight.eu

II. Related Works

A. Texture-less Object Detection

Object detection in images has been approached mainly in two ways: methods based on sliding window as Deformable Part Model from [1]; classification of region proposals produced with region proposal algorithms as the well known Selective Search from [2]. Thanks to the enormous increase in the research on Convolutional Neural Networks (CNNs), methods on region proposals have become prominent. R-CNN from [3] has been the first deep neural network trained for extracting features from region proposals using convolutional networks. This approach has been further improved in Faster R-CNN from [4] where the selective search region proposal algorithm is replaced with a Region Proposal Network (RPN, first time introduced with [5] and [6]) and the complete deep network is trained end-to-end for extracting the proposals and performing classification on the object's bounding box extracted using regression.

In this work we used a object detector called Single Shot Detector, from [7] that improve the quality and speed of the detection w.r.t. Faster R-CNN object detector by simultaneously producing a score for each object category in each predicted box and then classifying them. In this way the deep network is easier to train, faster, and ready to be integrated into other tasks.

B. Stereo Matching

Depth estimation from stereo is one of the most active topics in computer vision of the last 30 years. Given two rectified images, the problem is to find for each pixel in the reference image the corresponding point in the second image. Rectification reduces the correspondences' search along the same scanline. As described in [8], the main steps of stereo algorithms are: matching cost computation, cost aggregation, disparity optimization followed by a disparity refinement step. Methods can be categorized in local [9] [10] [11] [12], global [13] [14] [15] [16] or semiglobal [17] [18], depending on the techniques used to solve each step of the pipeline.

Recent works exploit the framework of PatchMatch Stereo [11] [14]. These methods exploit alternatively a random depth generation procedure and the propagation of depth, resulting in a total runtime cost of $O(W \log L)$, where W is the window size used to compute the matching cost between patches and L the number of searched disparities. The method proposed in [12], instead, strongly relies on superpixels, removing the linear dependency on on both the window size and label space. However, the superpixels's estimation requires a high computational time.

The active stereo problem has been recently addressed by exploiting efficient learning-based solutions [19] [20] [21] [22].

Recent deep learning based methods, among the others [23] [24] [25], provide very accurate results. However, these techniques usually don't generalize well to different contexts and require a fine-tuning of the CNN. Others [26] [27] [28] try to predict depth from a single image, but in practice are limited to very specific scenes.

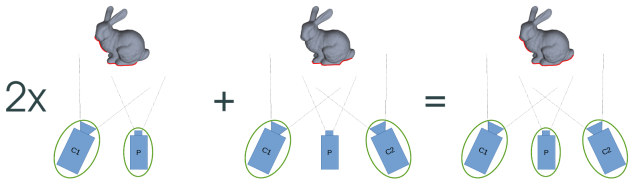


Fig. 2: The proposed system embeds multiple types of stereo vision technologies: 2 active stereo systems (C_1+P and C_2+P) and 1 passive stereo system (C_1+C_2).

III. Perception

The proposed system has been studied for industrial robotics applications where perception capabilities have a key role. Object detection and depth estimation are two of the main important tasks in this field, in the following we present more in detail the prototype we built and our custom solutions for the two applications, namely *texture-less object detection* based on deep transfer learning from synthetic data and *depth estimation* by mean of 3D Stereo matching within a pyramidal framework.

A. The Embedded Device

The proposed system (coded with the name FlexSight C1 and depicted in Fig.1) integrates both active and passive stereo matching technologies. In particular, two high resolution color cameras provide the system with passive stereo capabilities, while a random pattern projector mounted at the center of the cameras baseline provides active stereo capabilities by means of structured illumination of the scene (see Fig.2 for more details). It also integrates a CPU and a powerful Graphical Processing Unit (GPU) specifically designed to run expansive Machine Learning algorithms (e.g. Deep Learning) and a complete Unix based Operating System. This design enables the possibility to mount the system directly on top of a robotic cell and being connected bidirectionally with the robot system without the need of any external unit. This high level of flexibility makes the system appropriate for bin picking applications where a robotic manipulator needs to be driven by a vision system to detect and accurately manipulate highly cluttered objects.

B. Deep Learning Texture-less Object Detection

Data driven methods demonstrated to be very effective in detecting common textured and complex objects [30] [31], on the contrary, that is not the case with texture-less objects, e.g. mechanical parts in industrial bin picking applications. Texture-less objects do not expose so many features that a deep neural network can learn, and most often, having no texture highlights object symmetries and similarities making difficult the generalization of the task, in this way both classification and detection accuracy fall down rapidly.

The key aspect of every data driven task is the nature of the data itself, how the information encoded in the data is

(a) (b)

Fig. 3: (a) Example object classes (b) and test scenes from the *T-Less* dataset [29].

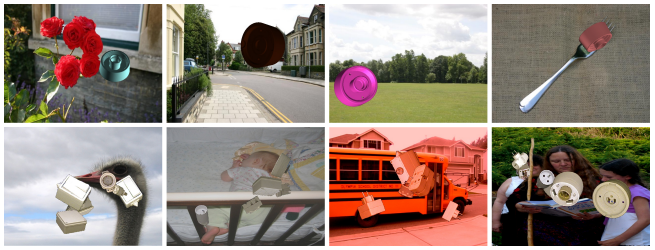


Fig. 4: Some examples from the synthetic generated dataset. Upper raw shows full synthetic example images, lower row shows semi-synthetic example images.

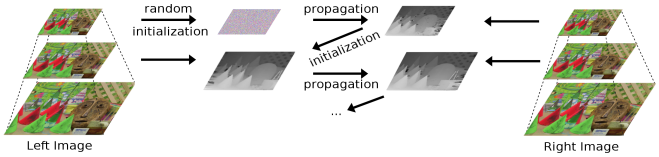


Fig. 5: Hierarchical architecture with propagation from top to bottom.

well exposed and how this can enhance the deep network capability in detect and emphasize highly generalized and heterogeneous features. Data collection is a fundamental aspect within the entire learning process and most of the time this task is done manually. Human intervention is often needed in collecting and then labeling the huge amount of data necessary for feeding the networks with enough information in order to avoid problems such as overfitting the input data. To limit, and somehow overcome, human intervention in data preparation we used synthetic data. Synthetic data is automatically generated by means of projection of the 3D object models onto random and highly generalized backgrounds. This process allows fast and accurate data collection. Without the need of manual intervention the data is generated directly ready to be used for the training of the detection model. Moreover, given the potentially infinite amount of data that can be generated, we are able to create well generalized datasets making the texture-less object detection training process more focused in learning more general features such as object shape, edges, occlusions and symmetries rather than color and appearance.

We exploited multiple CNNs architectures capable of running in inference mode on our embedded system, from accurate and efficient implementation of fully convolutional neural networks with region-based detector [32], to more fast and compact CNNs architectures such as [33] [34] [35] [7]. During training, for all our deep models, the layers responsible for feature extraction have been freezed to generic layers pre-trained on real images, and only the remaining layers are trained with our fully synthetic data. This process is also called *transfer learning* and it demonstrated to be very effective when training large and complex deep convolutional networks with pure synthetic images [36].

The system has been tested with some of the objects presented in the *T-Less Dataset* from [29]. Some examples of objects from T-Less can be seen in Fig.3.

C. Pyramidal Stereo Matching

In local stereo matching, a support window is centered on a pixel of the reference frame. In order to find the correspondence, this support window is displaced in the second image to find the point of lowest dissimilarity. Here is the implicit assumption that the pixels within the support region have a constant disparity. This does not apply to slanted surfaces, which are then reconstructed as compositions of frontal-parallel surfaces. The PatchMatch Stereo algorithm [11] overcomes this problem by estimating a 3D plane at each pixel onto which the support window is projected. As shown in [37], this technique provides very accurate disparities but it is also very slow, i. e. it is not suitable for real-time computing.

Inspired by [38] and [39], we propose to embed the PatchMatch Stereo algorithm [11] in a pyramidal framework (see Fig. 5) in order to reduce the matching time, while sensibly increasing the accuracy of the estimated disparities.

The disparity estimation of the upper levels of the pyramids (lower image resolution) is propagated on the lower levels (higher image resolution), enabling *i)* a considerable speed up of the random search step and *ii)* a reduction of the size of the support window in the lower pyramid levels.

IV. Experiments

A. Object Detection

As already anticipated, we overcome the problem of data acquisition and manual labeling by mean of synthetic data generation. In particular, starting from the 3D CAD model representation of our object, we project it onto random natural images as background, positioning the object in completely random position and orientation in the camera reference frame. Moreover, the objects are rendered using

Training Data	Obj_5	Obj_8	Obj_9	Obj_10	**Average**
Full Synthetic	0.3732	0.288	0.3179	0.2725	0.3129
Semi-Synthetic	0.5283	0.468	0.4956	0.477	**0.49225**

TABLE I: Performance on 4 of the objects' classes in the T-Less *test primesense* data. Results are given in terms of mAP@0.5 (mean Average Precision with 0.5 Intersection Over Union threshold).

random colors and illumination conditions (e.g. light intensity and position). In Fig.4 some example of the synthetic data are given. With this set of data we are able to train very deep networks for object detection, e.g. [32] [33] [34] [35] [7]. We will focus on *Single Shot Detector (SSD)* deep network as it has been the fastest in training time while achieving almost the same accuracy among all the tested networks.

A set of 10000 samples have been generated using the aforementioned procedure with random background extracted from the Microsoft Research Cambridge Object Recognition Image Database[1]. The deep model has been implemented with the TensorFlow Object Detection API[2] and trained on a machine equipped with a Nvidia GTX 1060 GPU Board.

Table I shows some quantitative results obtained using the synthetically generated data for 4 different classes of the dataset. The poor performance of this model reflects how it actually does not generalize well the task. An effective increase of performance has been obtained by training the model with *semi-synthetic* data: real images of real objects (as the ones in Fig.3 (a)) have been used instead of CAD renderings. This approach makes the transfer learning task easier, because real object images actually have more visual features, and make the network easily learn to detect and accurately distinguish among different objects on the test data.

Fig.6 shows some qualitative results obtained with the semi-synthetic approach. In particular, Fig.6 (a, b) show good detections in a dense scene, where the desired object is very similar to some other in the scene. Class similarity still remains a problem for the network, and it can be seen in the detection examples given in Fig.6 (c, d) where the desired object class is often confused with a similar one.

B. Stereo Matching

The proposed algorithm has been tested and evaluated on two popular benchmark data: Middlebury Stereo 2014 [40] and Kitti Stereo 2012 [41]. The evaluation has been performed on a i7-5700HQ CPU, 2.70GHz, and then implemented also on the embedded device, which is equipped with a ARM Cortex-A57 (quad-core), 1.73GHz, with an increase of runtime of 20%. The results in Tab. II refer to downscaled ($0.5Mpx$) version of the Middlebury training images. The evaluation on the Kitti dataset (see Tab. III), instead, has been performed using the original resolution (1242x375px) colored images. The state-of-the-art deep learning based methods [23] and [25] have been tested on a Nvidia GTX 1060 GPU using the pretrained models on Kitti-Stereo 2012 training set. As reported in Tab. III, [23] and [25] show superior performance when using fine-tuned models on the specific benchmark[3]. However, the degraded results in Tab. II show the difficulty of these techniques to generalize to completely different scenarios. The proposed method, instead, is able to generalize (the same set of parameters has been used in both evaluations), providing comparable results, in terms of bad pixel rate[4], in both benchmarks and outperforming other state-of-the-art algorithms. More specifically, the proposed method, compared to [11], is able to decrease the computational time up to 60%, while the accuracy of the disparities is improved up to 20%, demonstrating the effectiveness of the pyramidal framework.

Algorithm	bad 0.5	bad 1.0	bad 2.0	bad 4.0	Runtime
PSMNet [23]	89.4%	76.5%	57.1%	35.9%	0.7 s (GPU)
MC-CNN [25]	67.9%	40.2%	26.7%	13.9%	101 s (GPU)
ELAS [42]	67.3%	38.6%	25.9%	13.5%	0.3 s
[11]	47.2%	27.5%	15.8%	6.2%	22.3 s
Pyramidal Matching	46.3%	25.8%	12.9%	5.5%	8.7 s

TABLE II: Average performance on Middlebury training dataset [40].

Algorithm	bad 2.0	bad 3.0	Runtime
PSMNet [23]	2.4%	1.5%	0.4 s (GPU)
MC-CNN [25]	3.9%	2.4%	67 s (GPU)
ELAS [42]	10.8%	8.2%	0.2 s
[11]	8.1%	5.3%	13.1 s
Pyramidal Matching	7.4%	4.5%	5.6 s

TABLE III: Average performance on Kitti-Stereo 2012 dataset [41].

V. Conclusions and Future Works

In this work we presented an embedded system developed for industrial robots, where texture-less object detection and stereo matching are two important tasks. The system is meant for working without the need of any external computational unit, moreover it embeds vision techniques that minimize, and to some extent cancel, the human intervention in the loop. In particular, synthetic data demonstrated to have a huge potential in limiting the manual intervention in data acquisition and data annotation. The proposed synthetic pipeline is tested on a very challenging dataset, which contains low variability among the different classes of objects, most of them reflect high similarity and symmetries making the learning process difficult to generalize to unknown test data. Synthetic data may overcome the problem of generality by introducing high variability, both in terms of visual and geometric features. Further investigations must be done in order to increase the performance of pure synthetically generated data, so that to drive deep models to learn not mainly relying on the visual features, e.g. object textures, but focusing the learning process on more geometric features such as object shape and edges, which are independent from

[1] https://www.microsoft.com/en-us/download/details.aspx?id=52644

[2] https://github.com/tensorflow/models/tree/master/research/object_detection

[3] Results for [23] and [25] in Tab. III are taken from the Kitti stereo evaluation website (http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=stereo).

[4] The "bad N" metric, used in Tab. II and III, refers to the percentage of pixels whose disparity error is grater than N.

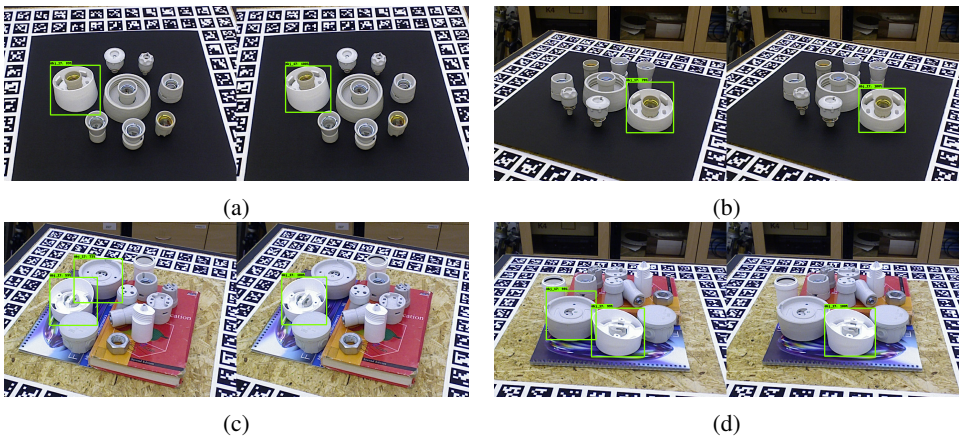


Fig. 6: Qualitative results of the texture-less object detection system. Each couple of images report the network detection *(left)* and the ground truth detection *(right)*. (a) and (b) represents positive examples of detection, with high score in terms of accuracy in the detection and *IoU (Intersection over Union)* between network detection and ground truth. On the contrary (c) and (d) depict some examples of failures.

the visual aspect of an object and more suitable for texture-less object detection tasks.

A possible solution is to include the depth information in the learning process. In this direction, we proposed a pyramidal stereo matching framework that provides accurate depth estimation that could be used in the detection pipeline. Although we showed an improvement of runtime performance compared to [11], the proposed stereo matching algorithm is still not suitable for real-time computing. In future work, we will exploit the massive parallelization capabilities of modern architectures by providing a GPU implementation that might lead to real-time performance.

REFERENCES

[1] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Discriminatively trained deformable part models, release 4," http://people.cs.uchicago.edu/ pff/latent-release4/.

[2] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 580–587.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 91–99.

[5] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2155–2162, 2014.

[6] C. Szegedy, S. E. Reed, D. Erhan, and D. Anguelov, "Scalable, high-quality object detection," *CoRR*, vol. abs/1412.1441, 2014. [Online]. Available: http://arxiv.org/abs/1412.1441

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, "SSD: single shot multibox detector," *CoRR*, vol. abs/1512.02325, 2015. [Online]. Available: http://arxiv.org/abs/1512.02325

[8] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 7–42, Apr 2002. [Online]. Available: https://doi.org/10.1023/A:1014573219977

[9] K.-J. Yoon and I.-S. Kweon, "Locally adaptive support-weight approach for visual correspondence search," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, June 2005, pp. 924–931 vol. 2.

[10] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," in *IEEE Computer Vision and Pattern Recognition*, 2011.

[11] C. R. Michael Bleyer and C. Rother, "Patchmatch stereo - stereo matching with slanted support windows," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2011, pp. 14.1–14.11, http://dx.doi.org/10.5244/C.25.14.

[12] J. Lu, H. Yang, D. Min, and M. N. Do, "Patch match filter: Efficient edge-aware filtering meets randomized search for fast correspondence field estimation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013.

[13] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient belief propagation for early vision," *International Journal of Computer Vision*, vol. 70, no. 1, pp. 41–54, Oct 2006. [Online]. Available: https://doi.org/10.1007/s11263-006-7899-4

[14] F. Besse, C. Rother, A. Fitzgibbon, and J. Kautz, "Pmbp: Patchmatch belief propagation for correspondence field estimation," *International Journal of Computer Vision*, vol. 110, 10 2013.

[15] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in Neural Information Processing Systems 24*, J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2011, pp. 109–117.

[16] Y. Li, D. Min, M. Brown, M. Do, and J. Lu, "Spm-bp: Sped-up patchmatch belief propagation for continuous mrfs," in *2015 International Conference on Computer Vision, ICCV 2015*. United States: Institute of Electrical and Electronics Engineers Inc., 2 2015, pp. 4006–4014.

[17] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, Feb 2008.

[18] M. Bleyer and M. Gelautz, "Simple but effective tree structures for dynamic programming-based stereo matching," in *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2008.

[19] Y. Zhang, S. Khamis, C. Rhemann, J. Valentin, A. Kowdle, V. Tankovich, S. Izadi, T. Funkhouser, and S. Fanello, "Activestereonet: End-to-end self-supervised learning for active stereo systems," 09 2018.

[20] S. R. Fanello, J. Valentin, C. Rhemann, A. Kowdle, V. Tankovich, P. Davidson, and S. Izadi, "UltraStereo: Efficient learning-based matching for active stereo systems," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jul 2017.

[21] S. R. Fanello, J. Valentin, A. Kowdle, C. Rhemann, V. Tankovich, C. Ciliberto, P. Davidson, and S. Izadi, "Low compute and fully

parallel computer vision with hashmatch," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 3894–3903.

[22] S. R. Fanello, C. Rhemann, V. Tankovich, A. Kowdle, S. O. Escolano, D. Kim, and S. Izadi, "Hyperdepth: Learning depth from structured light without matching," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 5441–5450.

[23] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5410–5418.

[24] Z. Liang, Y. Feng, Y. Guo, H. Liu, W. Chen, L. Qiao, L. Zhou, and J. Zhang, "Learning for disparity estimation through feature constancy," 2018.

[25] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *Journal of Machine Learning Research*, vol. 17, pp. 1–32, 2016.

[26] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, pp. 2366–2374.

[27] A. Saxena, M. Sun, and A. Y. Ng, "Make3d: Learning 3d scene structure from a single still image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 824–840, May 2009.

[28] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *CVPR*, 2017.

[29] T. Hodaň, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis, "T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects," *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017.

[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[31] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

[32] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: object detection via region-based fully convolutional networks," *CoRR*, vol. abs/1605.06409, 2016. [Online]. Available: http://arxiv.org/abs/1605.06409

[33] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015. [Online]. Available: http://arxiv.org/abs/1506.02640

[34] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," *CoRR*, vol. abs/1612.08242, 2016. [Online]. Available: http://arxiv.org/abs/1612.08242

[35] ——, "Yolov3: An incremental improvement," *CoRR*, vol. abs/1804.02767, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

[36] S. Hinterstoisser, V. Lepetit, P. Wohlhart, and K. Konolige, "On pre-trained image features and synthetic images for deep learning," *CoRR*, vol. abs/1710.10710, 2017. [Online]. Available: http://arxiv.org/abs/1710.10710

[37] B. Tippetts, D. J. Lee, K. Lillywhite, and J. Archibald, "Review of stereo vision algorithms and their suitability for resource-limited systems," *Journal of Real-Time Image Processing*, vol. 11, no. 1, pp. 5–25, Jan 2016.

[38] T. Xu, P. Cockshott, and S. Oehler, "Acceleration of stereo-matching on multi-core cpu and gpu," in *2014 IEEE Intl Conf on High Performance Computing and Communications, 2014 IEEE 6th Intl Symp on Cyberspace Safety and Security, 2014 IEEE 11th Intl Conf on Embedded Software and Syst (HPCC,CSS,ICESS)*, Aug 2014, pp. 108–115.

[39] Y. Hu, R. Song, and Y. Li, "Efficient coarse-to-fine patch match for large displacement optical flow," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 5704–5712.

[40] D. Scharstein, H. Hirschmller, Y. Kitajima, G. Krathwohl, N. Nesic, X. Wang, and P. Westling, "High-resolution stereo datasets with subpixel-accurate ground truth." in *GCPR*, ser. Lecture Notes in Computer Science, X. Jiang, J. Hornegger, and R. Koch, Eds., vol. 8753. Springer, 2014, pp. 31–42.

[41] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[42] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Asian Conference on Computer Vision*, Queenstown, New Zealand, November 2010.

Towards a flexible industrial robot system architecture

Raimund Edlinger[1], Lydia Leimer[2], Michael Zauner[3] and Roman Froschauer[4]

Abstract— **The present work deals with the recording, transmission and presentation of sensor data, which is transmitted by different sensors mounted on or in mobile robots. Complex, heterogeneous, modular robot systems require manufacturer- and user-independent standardized interfaces based on open communication standards and information models to enable interoperability and integration. Cross-system communication and data retrieval from different devices of different manufacturers is complicated by proprietary application programming interfaces (APIs). It is virtually impossible to exchange modules with devices from alternative manufacturers, which makes it difficult to integrate devices that meet the requirements. The OPC-UA communication interface is a platform-independent standard and is widely used in robotics and automation technology to connect compatible devices with different interfaces. In this paper we present the concept and implementation of a standardized communication interface for data exchange and visualization with ROS-based robot systems.**

I. INTRODUCTION

Modern robotic systems often comprise several components from different vendors to accomplish specific tasks. Resulting from this heterogeneous interfaces, communication protocols etc. prevent exchange of components (either hardware or software). Furthermore any kind of reconfiguration or reuse of components is virtually impossible. On a higher level of abstraction ROS (Robot Operating System) has been established as flexible middle-ware dealing with this problem. Unfortunately industrial robots and peripheral devices have no support for ROS and therefore limit the modular software ecosystem provided by ROS. Within the domain of industrial automation systems in the last years OPC-UA has been established as vendor-independent communication protocol. Recent developments such as OPC-UA Publish/Subscribe are dealing with loosely coupled devices at the shop floor level of automation and robot system. Therefore this paper proposes an overall system architecture for implementing flexible and intelligent robot systems featuring ROS on programming or behavioural layer and OPC-UA as core communication layer (see also [8]).

The flexibility of production that will become necessary in the future requires a high degree of cognition and independence from the automation solutions used. Within the framework of Industry 4.0, systems that integrate sensors, actuators and cognition are referred to as cyber-physical systems (CPS), which are regarded as key technologies for the production of the future [7]. The communication at CPS is not only based on pure data exchange between machines, but on the exchange between many functional units from machines to planning software to the integration of human decision makers. The specific implementations of a production system by end users require a simple and adaptable, flexible solution that enables interoperability in the overall system. Exemplary aspects to be considered in an inter-operable production system:

- Consistency of information flows
- Application of suitable methods for modelling, calculation, simulation and optimisation
- Involving people as creative actors in the global value stream
- Design of the man-machine interface

Systems currently on the market that pursue an integrated process chain strategy only consider very specific application areas and/or proprietary product or system technologies. ROS is one of the most popular frameworks for robotics researchers and manufacturers, but it does not provide the necessary security against possible cyber attacks and data theft. [1] present a secure communication channel for ROS which handles the communication between two nodes in a secure manner. [10] et al. introduce a new research tool to facilitate cyber-physical security research.

In the first chapters a short introduction to the topics OPC-UA and ROS [3] is given. Furthermore, an overview of the architecture "from sensor to user interface" and the structure of the ROS topics of a mobile recovery robot is presented. Finally, the research results will be presented.

II. STATE OF THE ART

A. ROS and ROS-Bridge v2.0

Robot Operating System (ROS) has been developed as part of the STAIR project at Stanford University. ROS is an open source software package that can be used for a variety of different applications within robotics. Currently ROS only runs on Unix based platforms. A port to Windows is basically possible, but is in the experimental stage [4]. The following objectives have been set for the development of ROS:

- ROS is designed as a peer-to-peer system: ROS makes it possible to combine several devices into one system. Instead of a central server, a peer-to-peer structure was used. The reason for this is that the entire system can be used in the general consists of devices on a mobile platform and further external computing units. Thus, computation intensive tasks can be transferred to more powerful computers. Since the majority of

*This work was supported by the Austrian Research Promotion Agency within the program "Strategy Innovative Upper Austria 2020" under grant agreement NR. 862013

[1]Raimund Edlinger, [2]Lydia Leimer, [3]Michael Zauner and [4]Roman Froschauer are with the Upper Austrian University of Applied Sciences, 4600 Wels, Austria {raimund.edlinger, lydia.leimer, michael.zauner, roman.froschauer}@fh-wels.at

the mobile robot system is only wirelessly connected to the external computers, unnecessary traffic over the already inefficient connection should be avoided. If the system was set up as/with a the form of a central server, communication between the robot modules would also have to be handled via the possibly external server. The result would be a heavy load on the wireless connection.

- ROS is open to all languages: ROS supports the programming languages C++, Python, LISP and Octave. In order to be able to use ROS on a par with other programming languages, data within ROS is represented in a neutral format.
- ROS is an open source project: All ROS source code is freely available under the BSD license. This also allows the development of commercial products with ROS without the obligation to make the developed code freely available.
- ROS has a modular structure: The functionalities of ROS are strictly packed into individual modules. This may lead to a loss of efficiency, but the stability of the system is improved. In case of faulty software fragments, the affected modules can be identified and deactivated if necessary.
- Encourage the generation of reusable code: Too strong system-specific dependencies make it difficult to extract and reuse code that has already been written. Through the implemented build system, ROS tries to encourage the reuse of existing code packages. ROS itself uses code from other open source projects [12].
- The rosbridge v2.0 server implementation makes it easy to add and modify protocol operations and decouples JSON-handling from the websockets server [2]. This allows users to arbitrarily change the specific websockets server implementation they are using.

B. OPC Unified Architecture (UA)

Based on OPC, one of the first standards established for inter-device communication in the 90s, OPC-UA was developed in order to provide platform independent functionality, supporting more complex data and systems than its predecessor [9]. The new standard, which promotes reliable, scalable and flexible communication across systems, can be used in automation as well as other areas for data transport. Using a self descriptive, extendable abstract base model, OPC-UA supports object orientation and provides security features [9]. The service oriented architecture is based on TCP/IP and covers layers 5, 6 and 7 of the OSI-Model. For use in non time-critical systems across internet and firewalls, XML-based variants have been specified. In order to support easy access for developers, the OPC Foundation provides the basic implementation as an OPC-UA Stack for different platforms, currently supporting C, C# and Java. Toolkits for C, C++, C# and Java are provided by independent vendors [5]. This makes OPC-UA particularly attractive for use in robot systems.

C. Other Communication Protocols

To support connectivity for various communication protocols and standards common interfaces should include interface implementations of a WebSocket API, OPC-UA Server and Client and an MQTT broker. Schel et.al. developed the concept and implementation of Manufacturing Service Bus (MSB) [13]. Another communication protocol is DDS (Data Distribution Service), which has its applications mainly in government and military uses. Like OPC-UA's server-client architecture, it provides data transport on a publish-subscribe basis. Due to its similar transport style, the ROS community uses DDS as communication standard for ROS 2. Unlike OPC-UA, DDS has been implemented over UDP, although some vendors provide support for TCP as well [11]. Also designed for machine to machine transport of telemetric data is MQTT (Message Queuing Telemetry Transport)[1]. It is aimed at usage scenarios with distant and/or mobile devices, where an efficient use of bandwidth is a requirement. Wireless Sensor Networks (WSNs) have been gaining increasing attention, where MQTT is used as an extension of the open publish/subscribe protocol [6].

III. Approach ROS - OPC-UA Stack

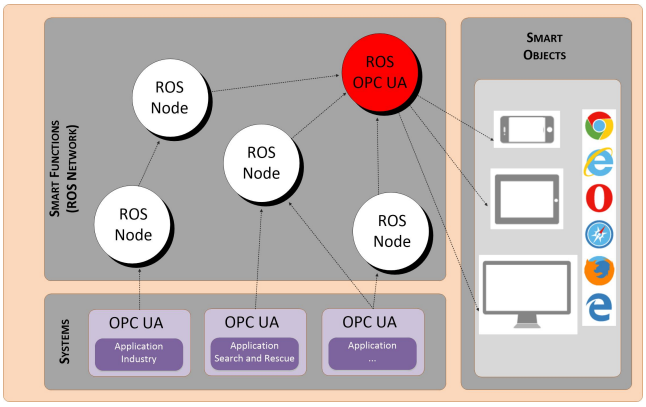


Fig. 1. ROS networks using OPC-UA Pub/Sub

Heterogeneous robot systems are robot systems that either combine several robots of the same or a different design or robots with complex machines, such as injection molding machines. Typically if mobile robots are equiped with external sensors, measuring systems, a connection to higher-level control systems and a control system are also used.

The appropriate equipment and safety systems are needed to implement the desired functions. With an increasing number of system components and possible actions, higher demands are required on the control of the overall system. Robot and component manufacturers often integrate UGV, robot arm systems, safety devices and peripherals into proprietary overall solutions. By using OPC-UA as a uniform communication standard between the system components, it is possible to make hardware and manufacturer independent components easier to integrate, exchange and thus more

[1]http://mqtt.org/

flexible in their application, in order to be able to engineer heterogeneous robot systems more quickly.

A. Data transmission from sensor to user interface

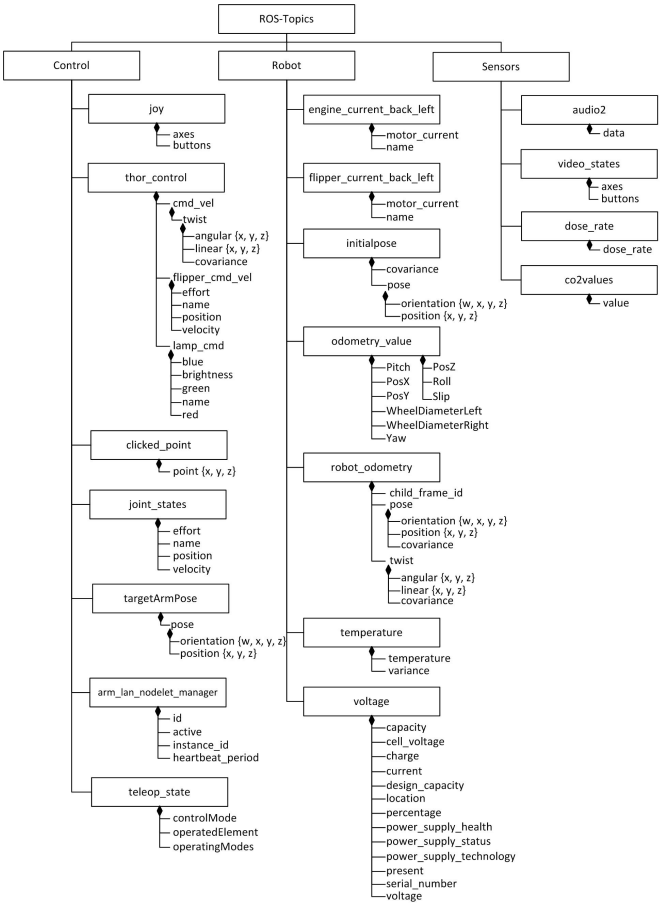


Fig. 2. ROS topics of a mobile rescue robot

In ROS operated robot systems, sensor data is forwarded from the operating system to ROS, where they can be made accessible to other applications. Sensor data can be read, visualized for users or processed, for example to create maps of the robot surroundings. Utilizing appropriate extensions like the ros_opcua_communication package, data can be provided to other applications and computers across the network. In this approach it is not the sensor itself which publishes its data and meta-data via OPC-UA, but ROS which publishes the information independently as ROS topics. These ROS topics include attributes common to all OPC-UA nodes as well as those only applicable to variable type nodes. Depending on its size and structure, data from several sensors can be merged before publishing. This is as much part of the freedom in implementation design as the naming of topics and variables. Fig. 2 shows an example of ROS topics of a mobile rescue robot. The separation into control, robot and sensors is content related and not reflected by the interface. ROS also publishes several topics required for the communication and by the software, such as for logfiles. Based on the existing ROS OPC-UA communication stack, the package was extended by a parameterization, where it is then possible to publish selected ROS topics and ROS services. This enables a robot-specific adaptation of the data to be published in relation to the wireless connection and limited bandwidth to mobile robot systems. Due to flexibility and the need for information in today's industrial world is constantly increasing and more and more process data must be processed and visualized. At the process control level, dashboards are required to give a quick overview of the most important indicators of a process. This is particularly important in order to react quickly to changes in a process.

B. Data visualization

Software for the visualization of robot data is available especially for the field of automation technology in industry. In most cases, the hardware manufacturers also develop and/or distribute the appropriate software. For service robots the offer is clearly smaller. In the following, visualization possibilities with the open source software ROS are presented. By using the OPC-UA technology, CERTEC EDV GmbH offers a universally applicable solution. Fig. 5 shows an example for a visualization of live stream, sensor and robot data. The interfaces created with Atvise[2] are made available via a web server, so that they can be accessed from any end device via a web browser. This approach also makes the interfaces useful for displaying sensor data on mobile devices.

IV. Show case

For demonstrating the approach the architecture of an existing mobile robot system has been adopted as shown in 3.

The architecture for the mobile robot documented in Fig. 3 utilizes CAN bus, USB and Ethernet for the transmission of sensor data. The sensor data of the motor controllers are transmitted to ROS via a main controller. Sensors that detect the robot's environment are often connected via USB or RJ45 to the onboard PC. Video data can also be transmitted to a ROS based system via Ethernet, which makes the live stream available via HTTP. For simple and versatile user interaction a web-based visualization tool-kit featuring OPC-UA communication has been used.

A ROS-controlled robot, see fig. 4 provides the sensor data via OPC-UA and is therefore suitable for OPC-UA capable visualization software. With the help of the visualization software Atvise, the data can be grouped and formatted on graphical user interfaces. Depending on the application and user, interfaces can be designed for different terminal devices and screen sizes. Figure 5 shows a graphical user interface which has been designed for different terminal devices and screen sizes and is designed for any browser-enabled devices such as desktop, laptop or tablet computers, PDAs or smartphones.

[2]https://www.atvise.com/en/

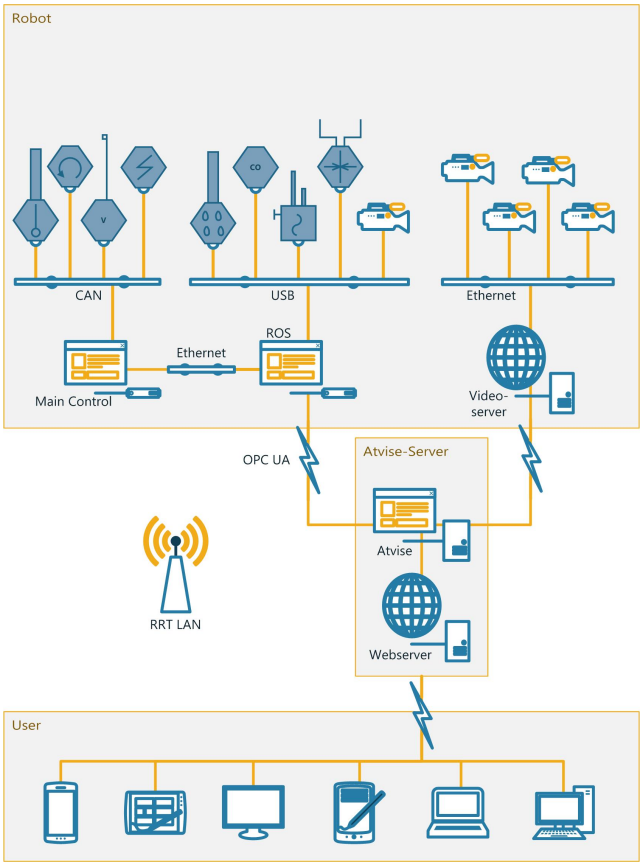


Fig. 3. Architecture: from sensor to user interface

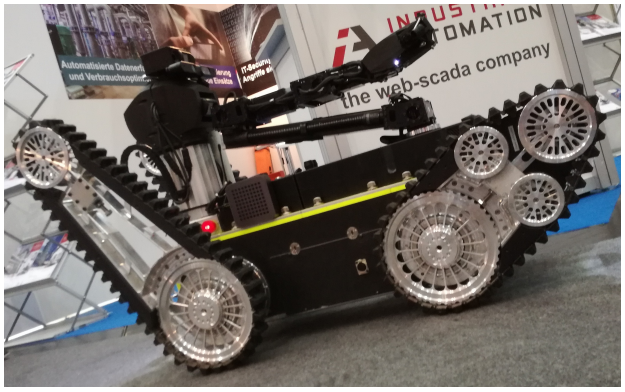


Fig. 4. Rescue Robot

V. Conclusion

The basic research focuses on OPC-UA, a platform-independent and object-oriented standard for communication between machines that has been developed [3]. OPC-UA covers the upper three layers of the OSI model. An overview graphic was created for the data transmission in which it is schematically shown that the sensors on the robot can be connected via different systems, the data is passed on via servers and prepared for the end application. The rough structure of the controller, robot and sensor data is also graphically documented, the fine structure of the so-called

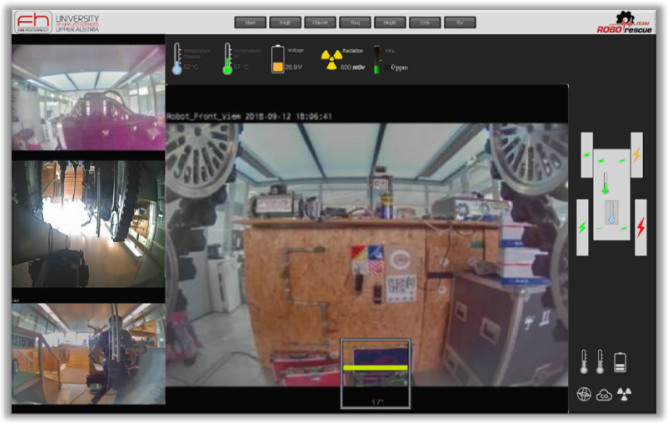


Fig. 5. Graphical user interface for prototype

nodes is recorded in the OPC-UA fundamentals. In order to provide an overview of the possible approach to visualization, three visualization tools are presented: the open source framework ROS for mobile robots, the universally applicable Atvise, and Visual Components for industrial automation. Sensor data can be visualized in different ways, there are hardly any limits to creativity. For example, distances can be plotted directly in camera images, physical quantities displayed using slide controls or dial gauges, or camera images are distorted/corrected, and can be superimposed. The good overview of the components and technologies used the graphics and explanations provided are suitable for the training of future Employees. The documentation can also be used as a basis for further developments.

References

[1] B. Breiling, B. Dieber, and P. Schartner, "Secure communication for the robot operating system," in *2017 Annual IEEE International Systems Conference (SysCon)*. IEEE, 2017, pp. 1–6.

[2] C. Crick, G. Jay, S. Osentoski, B. Pitzer, and O. C. Jenkins, "Rosbridge: Ros for non-ros users," in *Robotics Research*. Springer, 2017, pp. 493–504.

[3] O. Foundation. (Accessed: 2015-11-25) Opc foundation. opc ua in the reference architecture model rami 4.0. [Online]. Available: https://opcconnect.opcfoundation.org/2015/06/opc-ua-in-the-reference-architecture-model-rami-4-0/

[4] O. S. R. Foundation. (Accessed: 2017-04-03) Ros introduction. [Online]. Available: http://wiki.ros.org/ros/Introduction

[5] A. GmbH. (Accessed: 2018-07-18) Opc unified architecture. [Online]. Available: http://www.ascolab.com/de/unified-architecture/

[6] U. Hunkeler, H. L. Truong, and A. Stanford-Clark, "Mqtt-sa publish/subscribe protocol for wireless sensor networks," in *2008 3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE'08)*. IEEE, 2008, pp. 791–798.

[7] N. Jazdi, "Cyber physical systems in the context of industry 4.0," in *2014 IEEE international conference on automation, quality and testing, robotics*. IEEE, 2014, pp. 1–4.

[8] C. Lalancette. (Accessed: 2017-04-24) Robot operating system tutorial. [Online]. Available: http://wiki.ros.org/ROS/Tutorials/ExaminingPublisherSubscriber

[9] W. Mahnke, S.-H. Leitner, and M. Damm, *OPC unified architecture*. Springer Science & Business Media, 2009.

[10] J. McClean, C. Stull, C. Farrar, and D. Mascareñas, "A preliminary cyber-physical security assessment of the robot operating system (ros)," in *Unmanned Systems Technology XV*, vol. 8741. International Society for Optics and Photonics, 2013, p. 874110.

[11] G. Pardo-Castellote, "Omg data-distribution service: Architectural overview," in *23rd International Conference on Distributed Computing Systems Workshops, 2003. Proceedings.* IEEE, 2003, pp. 200–206.
[12] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.
[13] D. Schel, C. Henkel, D. Stock, O. Meyer, G. Rauhöft, P. Einberger, M. Stöhr, M. A. Daxer, and J. Seidelmann, "Manufacturing service bus: an implementation," in *11th CIRP Conf. Intell. Comput. Manuf. Eng.*, vol. 67, 2017, p. 6.

Flexible industrial mobile manipulation: a software perspective

Thomas Haspl[1], Benjamin Breiling[1], Bernhard Dieber[1], Marc Pichler[1], Guido Breitenhuber[1]

Abstract— **With ongoing research in robotics, some specific architectural approaches of robotic systems earn more and more interest by all kinds of industries. Mobile manipulators– robots consisting of a mobile base and a serial manipulator– provide the ability to make robotic manipulation location-independent, which will be an essential feature in future production. Such robot platforms offer a high level of flexibility and efficiency of robot applications. Especially under the aspect of modularity, mobile manipulators would provide even more flexibility by offering the possibility to exchange or extend the robot hardware for specific applications. To achieve this, modularity also has to be considered in software. In this paper, we present a software architecture for modular mobile manipulation applications. It provides mechanism for reconfigurability, easy programming, and an easy approach for adding external hardware components. Being targeted at industrial use, the architecture also considers security and software deployment aspects. These considerations will, in combination with all the other aspects, be presented by means of two modular mobile manipulation platforms and a set of representative scenarios.**

I. INTRODUCTION

Nowadays, the number of robots used in different industrial and non-industrial is steadily rising. These application areas range from manufacturing, transportation, household, up to health care, amongst others. Independent from the application area, the requirements to robotic platforms also heavily increased. Due to the demand of flexible applicability and the desire to let robots operate in unstructured environments, a significant amount of research work has been dedicated to this. One very promising type of robots for many kinds of applications are mobile manipulators. Those combinations of robot arms and mobile bases free the serial manipulator (the arm) from its location-dependence thus enabling a whole new range of applications. In industry, mobile manipulation is currently taking up speed where sensitive robots are employed working next to humans on the shop floor. Key aspects in the successful application of mobile manipulation is the flexibility in the use of the robot. This requires easy programming for end-users and system integrators, extensibility and reconfigurability of the robot hardware as well as the easy integration of the robot into the manufacturing IT infrastructure. Additionally, a higher level of modularity of robotic hard- and software inevitably requires strategies for secure communication between the robot's components in order to keep the system safe from intrusion.

While there have been many reports of mobile manipulators in research, this paper presents a software architecture that has already been proven in industrial settings by enabling a highly flexible use of robots in various scenarios. The architecture explicitly considers mobile manipulators that are modular in their nature. Thus, one key requirement for the software architecture is that it must be easily possible to integrate software components for new pieces of hardware like sensors or grippers but also to exchange core hardware elements like the robotic arm or mobile base. So, while we here mainly report on a configuration consisting of a *MiR100* mobile base and a *UR10* arm, the architecture also supports the integration of different bases and arms of other types and vendors. while the architecture focuses on the industrial use of mobile manipulators, it can also be used in other settings in- and outside of industry.

This paper is structured as follows: section II depicts the state of the art, which is relevant for our software architecture. This includes reconfigurability of hard- and software components, security in robotic applications and the issue of deploying robot software. Section III then elaborates our developed architecture. By providing necessary requirements at the beginning, this section then describes the core components and processing layers of the architecture. The topics security and deployment will finalize this section. Then, in section IV an evaluation of the architecture by means of three scenarios will be done. Before that, we present two in-house developed mobile manipulation platforms - the CHIMERA which can operate in industrial environments and a modular research platform for mobile manipulation. Both are driven by the software stack presented in this paper.

II. STATE OF THE ART

Robot software is becoming increasingly important as the focus of the development shifts from hardware capabilities to more intelligent robots. In terms of software, the *Robot Operating System (ROS)* [19] has become the predominant framework for prototyping robot applications and building intelligent robotic products. Over the years, several architectural approaches have been presented for domains with references to robotics [18], [12], [1], [14].

A. Reconfigurability of Hard- and Software

Modularity in robots has been proclaimed as one of the most promising approaches to making robots more flexible while at the same time decreasing integration times [15], [23], [20]. Modularity in hardware enables the reconfiguration or extension on a physical level by adding, removing or swapping hardware parts. In software, modularization aids re-usability and thus is aimed at minimizing time required to develop a solution.

[1]The authors are with the Institute for ROBOTICS and Mechatronics, JOANNEUM RESEARCH, Klagenfurt, Austria firstname.lastname@joanneum.at

B. Security

Also security has been addressed in our previous work. Especially in the context of *ROS*, we have focused on securing robots i) in the applications [11], ii) in the *ROS*-core [3] or iii) in their integration in larger IT infrastructures [9]. In addition to our work, contributions to ii)[22] and iii)[16] can be found as well. In general, a multi-layer approach to security is recommended in- and outside of the robotics domain (i.e., the combination of all three aforementioned security approaches) [5], [10]. Security flaws in robot systems are especially grave since recent studies have shown a very large number of robots to be publicly accessible via the internet [8], [21].

C. Deployment

Software deployment is an activity performed for or by the customer where all the customer-centric configuration and customization is done [7]. It is a process consisting of activities related to the release, installation, activation, deactivation, update, and removal of components [6]. Different generic standards for software deployment like RPM (RPM Package Manager) or DPKG (Debian Package) on Linux or OMG's (Object Management Group) deployment and configuration specification for component based distributed applications [17] exist. In the field of robotics software deployment is a challenging problem due to the complexity and variability of robots [4], [13]. Robot manufacturers tackle the problem with custom proprietary solutions. *Universal Robots* uses *URCaps* for installing additional components or system updates. *Franka Emika* has a concept of APPs, which add new functionality to the controller. In *ROS* there is `roslaunch`, which at least covers the configuration and activation step of the deployment process. Yet we found no related research for an end-to-end approach on how to deploy a robotic software from a software developer via optional system integrators and/or customers to a robot in a secure way where each of the stakeholders should be able to add or modify parts of a deployment package.
In contrast to the works reported above, the architecture presented in this paper considers all of the aspects of security, modularity and deployment within an end-to-end concept for industrial mobile manipulators.

III. ARCHITECTURE

In this section, we first present the requirements for our architecture before going into detail on its components and their interactions.

A. Requirements

Considering the hardware-related aspects and requirements as mentioned in the introduction, we can derive a number of requirements regarding the software structure of mobile manipulators from them.

1) Hardware reconfiguration: As hardware reconfiguration is key to enabling a flexible use of a robot, it is also necessary for the underlying software to support this aspect. This requires a hardware-driver model and defined communication channels to other components. This basically means that the software has to be designed in a way that its components for controlling a mobile platform and a serial manipulator are capable of communicating over defined interfaces, but can basically operate self-contained. Furthermore, these components should meet the requirement that they are independent from the applied hardware. Providing these attributes a software structure for mobile manipulation can reach a high level of flexibility and reusability.

2) Security: It is undoubted that human safety is an essential aspect of all robotic platforms and applications. When it comes to mobile manipulation this aspect earns even more interest as the typical operation area of mobile manipulators can also include interactions with humans in any kind of way. A key requirement we actually want to address with this argument is security. Security in a robotic context relates to a secure (intrusion-proof) communication structure between hardware and software components, whereas the term safety is used for human safety. Even the most excellent safety concept for a robotic application becomes worthless if security issues are not considered. Accordingly, security is strongly connected to safety and as we already stated in our previous work [2] it is even practically not possible to guarantee safety without considering security. In this sense, security is a necessary but not sufficient precondition for safety. It must be assured, that (typically software-based) safety measures are not undermined by a lack of security.

3) Extensibility: Another requirement we want to point out is extensibility. A system integrator as one important end-user of our software does not want to struggle with issues because of a complicated way of programming when it comes to extending the software setup. The whole concept should be based on structural decisions which allow a modular behavior of the application software. With this, it should not only be easy to add custom modules such as sensors or grippers in hardware, but also in software.

4) Deployment: In addition to easy programming, a simple and scalable procedure for deployment and integration is an important point. When deploying software to a robot it must be made sure that only genuine software components are deployed and used on the robot. We dedicated a particular section to this important topic in III-E.

B. Components

The core of our software consists of four components which provide the basic functionality for a mobile manipulator consisting of a mobile platform, a serial manipulator and a gripper. Additionally, the architecture provides interfaces for extending the basic application. An exemplary composition of an application using our software is visualized in figure 1. Basically, the architecture defines three layers, one for components working close to hardware, a task layer, which abstracts specific hardware functionality and exposes atomic

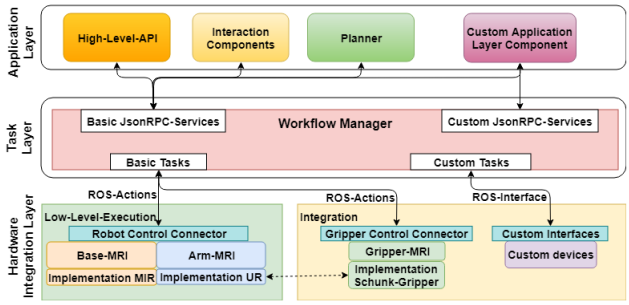


Fig. 1. Architecture of the software running on the core NUC

functions to the application layer where complex applications and the integration to outside infrastructure are realized.

C. Architecture layers

In the application layer, the implementation of a specific use case takes place. Various options are available for this, ranging from code-based implementation to teaching by demonstration. The result is always the same, namely a workflow with already parameterized tasks. A high-level *API* is maybe one of the flexible ways to create applications. Here, a system integrator can easily implement new processes for the robot platform. Basically, client libraries for any language can be implemented as long as they support *JSON-RPC* communication. Another option for creating executable workflows is a planner. Planners are basically used for more complex actions, where for instance tasks have to be dynamically reordered.

The task layer is responsible for processing a workflow task by task and to manage the status feedback to the application layer. From a workflow definition, which can be an ordered sequence or complexer structures with branches, one task after another is triggered by the *Workflow Manager*. This happens by calling the appropriate *ROS*-action server from the underlying robot control connector (or custom interfaces). A workflow also contains strategies if one or more tasks fail. As this information is required by the workflow templates, it can be assured that a workflow always terminates in a defined state, even in case of hardware execution errors.

On the hardware integration layer, the actuation of hardware components, thus, the actual execution of tasks takes place. The responsible components, namely the *Robot Control Connector* and the *Gripper Control Connector*, serve as an interface between the single tasks maintained by the *Workflow Manager* from the task layer and the robot commands used by the *Multi Robot Interface*. In particular, the control connectors wait for an *ROS*-action goal, extract the task parameters and utilize the *Multi Robot Interface* to generate and execute base- and arm commands (*Robot Control Connector*) as well as gripper commands (*Gripper Control Connector*) respectively. For integrating custom hardware, a system integrator can define and implement his own task definitions and execution components. After completion of a command, successful or not, the result is sent back to the *Workflow Manager*. In the current version of our software every task is executed in a blocking manner. Hence, parallel task execution is not yet supported.

1) Workflow Manager: The *Workflow Manager* (WFM) is used to trigger individual atomic actions (e.g. navigating the mobile base to a specific pose, move the TCP of the serial manipulator to a specific pose, move the serial manipulator to a specific joint configuration), but also supports composition of such atomic actions to complex workflows. The mobile manipulator then executes these sequences autonomously. The dynamic JSON-RPC interface of the *Workflow Manager* is extended if new hardware is integrated in the integration layer. This means that higher layer actions of this hardware are also available. On task level, the *Workflow Manager* runs ROS-action clients for sending basic operations to the *Robot Control Connector* and *Gripper Control Connector* respectively. If additional hardware needs to be integrated, the WFM can be dynamically extended by new interfaces to the hardware integration layer. For that, the source code of the interfacing components has to be stored in a predefined directory. Then, the *Workflow Manager* loads the interfacing components automatically at the next startup, while its source code remains untouched.

2) Control connectors: These components, namely the *Robot Control Connector* and the *Gripper Control Connector*, serve as an interface between the single tasks maintained by the *Workflow Manager* and the robot commands used by the *Multi Robot Interface* (see below). In particular, the control connectors maintain ROS-action servers, receiving task actions, in order to execute commands on the robot's hardware via the *Multi Robot Interface*. Like the *Workflow Manager* the source code of the control connectors is independent from the current hardware configuration. However, they must be initialized with hardware specific implementations of the related *Multi Robot Interface* components at runtime.

3) Multi Robot Interface: The *Multi Robot Interface* (MRI) is a software interface to access common functionalities of different robot classes. It provides generic command factories for serial manipulators, mobile bases as well as grippers and so abstracts the underlying hardware. This especially means that task- and application layer components are independent from the current hardware configuration and do not need to be modified when the hardware configuration and their software drivers change. As depicted in figure 1, the MRI is split into three parts; i) Base-MRI providing basic commands for controlling a mobile base, ii) Arm-MRI providing basic commands for controlling a serial manipulator, iii) Gripper-MRI providing basic commands for controlling a gripper. These interfaces can then be utilized by the *Robot Control Connector* and *Gripper Control Connector* to trigger commands. As a consequence, the source code of both control connectors is independent from the hardware configuration. As already described in III-C.2 the low level execution process is initialized with hardware specific implementations of the MRI components at runtime.

D. Security

As already indicated in the introduction, there is a gap between security requirements for the internal communication and the need of interaction with external entities. Consequently, our proposed security architecture has to work on multiple levels, in particular on network, operating system and application level. The overall network architecture of our mobile manipulator, shown in figure 3, is separated into i) an internal network for data processing and low level task execution, ii) an interaction network for the communication to external components. Particularly, the second provides secure access to an administrative interface for maintenance and configuration as well as secured interfaces to a superordinate task planning component and external interaction devices. The most important point in this setup is that there is no direct network connection between the interaction network on the right hand side and the robotic hardware on the left hand side of the figure. As a consequence, an attacker will not be able to directly aim at potentially vulnerable devices (with regards to cybersecurity), like the base or the serial manipulator. Further, we pass the information between the two networks via inter-process communication techniques, in order to prevent unauthorized access to the passed information according to the security requirements we already proposed in [9].

E. Secure Deployment

In order to deploy new software and updates to the mobile manipulation platform, the architecture has to allow for both, secure deployment as well as secure execution of the deployed code. In our work, secure deployment is achieved by introducing a signed package format, which allows configuration data to be modified by a system integrator, while still ensuring integrity of the software contained in the package. Secure execution on the other hand is accomplished by extensive sandboxing of applications, as well as by restricting access to system resources such as network or file access.

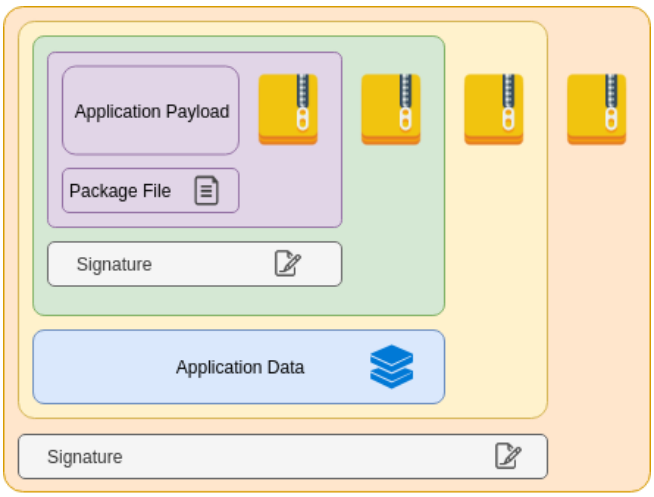


Fig. 2. Application Package

1) Package format: As seen in Figure 2, an application package consists of multiple parts. This allows for flexibility in modifications to parts of the package.

The application payload is the actual application or plug-in that is deployed to a part of the system. For actual standalone software, this is a *Docker* container, for plug-ins this might be Python modules, dynamically linked libraries, or any other payload authored by a developer. The application payload is digitally signed by the developer, and this signature is used on deployment to verify the package's integrity and origin.

The package file is a text file that contains metadata about the package itself such as name, version information and intended target container.

The application data part contains configuration files and other modifiable resources. Splitting application data and application payload in two allows for some flexibility in configuration of packages. For instance if a system integrator wants to reconfigure a software package for a specific use-case. This can be accomplished by unpacking the outer package, changing the application data, and finally re-signing the outer package with their own private-key. Since the application payload is left untouched, it is not re-signed and can be assumed to still work as intended by its developers.

2) Package broker: The package broker is an intermediary component between the interaction network and the internal network (as seen in Figure 3). Due to the necessity of deploying packages to any component of the system as well as due to security requirements, it exists as a component that receives packages from the interaction network and distributes them to either the interaction network or the internal network. The package broker is responsible for package-verification and deployment to the intended target package container within the system.

3) Package containers: A package container registers with the broker to receive packages since the broker acts as the central instance for package distribution inside the system. Once a package arrives at the broker, the broker inspects the signature and extracts the intended target package container from the package file. It then verifies that the creator of the package and source files has the permissions to deploy to the package container in question. After performing these tasks, the broker hands the package off to the package container for further processing.

Once the package arrives at the package container, the container starts unpacking the package, does some additional verification, and finally runs the package container-specific update procedure. When done, it reports the package as installed to the broker, providing feedback to the user, and giving the broker the authority to start, stop or uninstall the application in question.

4) Additional considerations: In order to make the packaging procedure easier, a separate application for packaging and signing is provided. In addition to this, a user interface connected to the package broker allows for starting and stopping running applications on package containers, as well as for installing and uninstalling packages, hiding the inner

workings of the deployment and maintenance procedures.

IV. EVALUATION

In this section we evaluate our software architecture against the requirements stated in III-A. For that, we first introduce two mobile manipulators able of running our software. Then, we describe different scenarios, how our software architecture supports i) the exchange of the hardware modules, ii) the integration of custom hardware and iii) the deployment of new software packages.

A. Platforms

As target platforms for our software architecture we aim at two mobile manipulation platforms with certain characteristics. One manipulator is used in industrial settings and has been proven in real factories already. The second model is a research platform composed of exchangeable modules. The cross-cutting assumption for this in the software architecture is to have a robot with a mobile base and a serial manipulator attached.

1) CHIMERA: The CHIMERA consists of a *MiR100* including its internal hardware parts, the network with router, PC (Intel® NUC) and PLC, as well as an UR controller for an *UR10* robot arm and an additional PC (Intel® NUC) for integrating additional hardware and software. Figure 3 shows the overall network architecture of the CHIMERA, which is shown in figure 4.

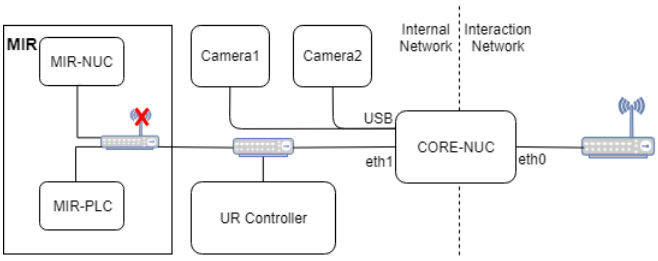


Fig. 3. Network architecture

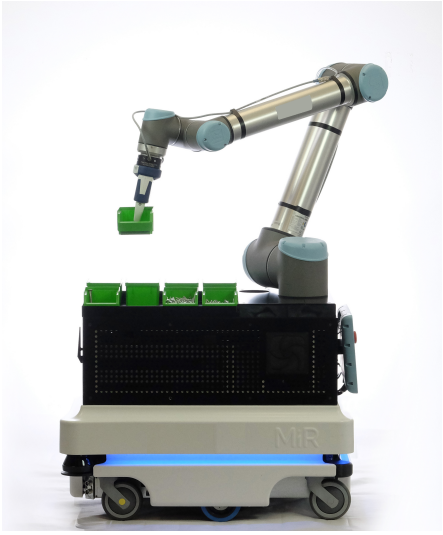


Fig. 4. CHIMERA

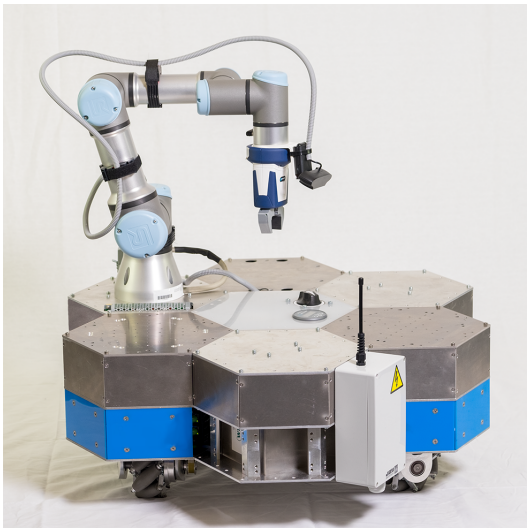


Fig. 5. Research Platform for modular mobile Manipulation

From the software point of view, our goal is to combine the functionalities of the *MiR100* and the *UR10* in order to execute customizable mobile manipulation workflows. At the same time we keep our software as generic as possible, which means that we can exchange the base, the arm or both without changing the major part of our software.

2) Research platform for modular mobile manipulation: The specific characteristic of our wheeled mobile manipulation research platform is its modular mobile base. It can be assembled with an arbitrary number of hexagonally shaped modules in order to optimize it for specific applications. Thereby it is possible to create the most suitable robot for a given problem. The hexagonal modules can be equipped with different wheel types, other modules can be used for power supply, the robot arm or other hardware components. A computing module contains an *Intel® NUC* PC, which is responsible for running all software parts. At startup, a configuration file, that contains information about the physical configuration of the mobile base, is parsed. The gripping module includes an *UR3* as well as its controller. A composition of such a wheeled mobile manipulator can be seen in figure 5. The mobile base in this figure consists of three driving modules with omni-directional wheels, two battery modules, a gripping module with an *UR3* and a computing module.

B. Scenarios

In this section, we iterate through various situations in the lifecycle of our two platforms and explain how the software architecture will support this.

1) Integration of custom hardware: In the following scenario, we describe how a system integrator (SI) can integrate a new sensor and utilize it on the task layer in order to include it into a custom workflow. First, the SI needs to deploy a driver software for the sensor using the secure deployment mechanism. The driver can then be started in an

own process. In order to use the sensor within a customized workflow, the *Workflow Manager* then needs to be extended with a sensor specific task definition and an interface to the newly integrated sensor. To make the sensor specific workflow available for an application layer component the system integrator also has to provided a related JSON-RPC service.

2) Deployment of a new software package: In this example, we outline how the SI deploys the necessary sensor driver required for the previous scenario using the secure deployment mechanism. In a first step, the sensor driver has to be implemented using a programming language of the SI's choice. Furthermore, an application setup-script to install necessary dependencies has to be created. After implementation, the driver is packaged using the packaging application which generates a *Docker* container based on the setup-script, the docker container is then packaged with the package file and is subsequently signed by the SI. The SI then logs in and pushes the package to the *package broker*, which will then perform the deployment as described in Section III-E.

3) Exchange of basic hardware: Let's suppose, we successfully executed an use-case on the research platform and want to run it on the CHIMERA. To do so, we need a *MiR100* implementation of the *Base-MRI* interface, replacing the research platform implementation. Further we have to change the configuration of the UR implementation in order to use the *UR10* instead of an *UR3*. The *Robot Control Connector* is then initialized with the *MiR100* implementation at startup. The software architecture shown in figure 1 is unmodified, including the workflows which need to be executed. Only the driver implementations for hardware components must be exchanged. Of course transferring applications to different hardware does not guarantee that it can be executed successfully out-of-the box due to physical differences like reachability of the robotic arm or maneuverability of the mobile base.

V. CONCLUSIONS

In this paper we have described a flexible software architecture for modular mobile manipulators that is suitable (and has been tested) in industrial scenarios. We have shown that it can be used on both, platforms for industrial and research use. Thus, the architecture supports the required flexibility very well.

In future work, we plan to do more experiments with different kinds of hardware in order to strengthen the robustness of the architecture. We also plan to advance tool support for developers and system integrators to simplify development and deployment of software components.

ACKNOWLEDGMENT

This work has been supported by the Austrian Research Promotion Agency in the program "ICT of the Future" funded project FlexIFF (grant no. 861264)

REFERENCES

[1] R. Alami, R. Chatila, S. Fleury, M. Ghallab, and F. Ingrand, "An architecture for autonomy," The International Journal of Robotics Research, vol. 17, no. 4, pp. 315–337, 1998.

[2] B. Breiling, B. Dieber, B. Reiterer, A. Schlotzhauer, and S. Taurer, "Safety nicht ohne security in der kollaborativen robotik."

[3] B. Breiling, B. Dieber, and P. Schartner, "Secure communication for the robot operating system," in 11th Annual IEEE International Systems Conference, SysCon 2017 - Proceedings, 2017.

[4] D. Brugali, A. Agah, B. MacDonald, I. A. Nesnas, and W. D. Smart, "Trends in robot software domain engineering," in Software Engineering for Experimental Robotics. Springer, 2007, pp. 3–8.

[5] E. Byres, P. E. Dr, and D. Hoffman, "The myths and facts behind cyber security risks for industrial control systems," in In Proc. of VDE Kongress, 2004.

[6] A. Carzaniga, A. Fuggetta, R. S. Hall, D. Heimbigner, A. Van Der Hoek, and A. L. Wolf, "A characterization framework for software deployment technologies," COLORADO STATE UNIV FORT COLLINS DEPT OF COMPUTER SCIENCE, Tech. Rep., 1998.

[7] A. Dearle, "Software deployment, past, present and future," in Future of Software Engineering (FOSE'07). IEEE, 2007, pp. 269–284.

[8] N. DeMarinis, S. Tellex, V. Kemerlis, G. Konidaris, and R. Fonseca, "Scanning the Internet for ROS: A View of Security in Robotics Research," arXiv preprint arXiv:1808.03322, 2018.

[9] B. Dieber and B. Breiling, "Security considerations in modular mobile manipulation," in Proceedings of the 3rd International Conference on Robotic Computing. Naples, Italy: IEEE, Feb. 2019, pp. 70–77.

[10] B. Dieber, B. Breiling, S. Taurer, S. Kacianka, S. Rass, and P. Schartner, "Security for the Robot Operating System," Robotics and Autonomous Systems, vol. 98, pp. 192–203, 2017.

[11] B. Dieber, S. Kacianka, S. Rass, and P. Schartner, "Application-level security for ROS-based applications," in IEEE International Conference on Intelligent Robots and Systems, vol. 2016-Novem, 2016, pp. 4477–4482.

[12] B. Hayes-Roth, K. Pfleger, P. Lalanda, P. Morignot, and M. Balabanovic, "A domain-specific software architecture for adaptive intelligent systems," IEEE Transactions on software engineering, vol. 21, no. 4, pp. 288–301, 1995.

[13] N. Hochgeschwender, L. Gherardi, A. Shakhirmardanov, G. K. Kraetzschmar, D. Brugali, and H. Bruyninckx, "A model-based approach to software deployment in robotics," in 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2013, pp. 3907–3914.

[14] A. Koubaa, "Service-oriented software architecture for cloud robotics," arXiv preprint arXiv:1901.08173, 2019.

[15] V. Mayoral, A. Hernndez, R. Kojcev, I. Muguruza, I. Zamalloa, A. Bilbao, and L. Usategi, "The shift in the robotics paradigm the hardware robot operating system (h-ros); an infrastructure to create interoperable robot components," in 2017 NASA/ESA Conference on Adaptive Hardware and Systems (AHS), July 2017, pp. 229–236.

[16] M. Mukhandi, D. Portugal, S. Pereira, and M. Couceiro, "A novel solution for securing robot communications based on the MQTT protocol and ROS," in Proceedings of the 2019 IEEE/SICE International Symposium on System Integrations (SII 2019), 2019.

[17] OMG, Deployment and Configuration of Component-based Distributed Applications Specification Version 4.0, 01 2006.

[18] M. Quigley, E. Berger, A. Y. Ng, et al., "Stair: Hardware and software architecture," in AAAI 2007 Robotics Workshop, Vancouver, BC, 2007, pp. 31–37.

[19] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in ICRA workshop on open source software, vol. 3, no. 3.2, 2009, p. 5.

[20] K. Roufas and M. Yim, "Software architecture for modular self-reconfigurable robots," in IROS, 2001.

[21] V. M. Vilches, G. O. Mendia, X. P. Baskaran, A. H. Cordero, L. U. S. Juan, E. Gil-Uriarte, O. O. S. de Urabain, and L. A. Kirschgens, "aztarna, a footprinting tool for robots," arXiv preprint arXiv:1812.09490, 2018.

[22] R. White, H. Christensen, and M. Quigley, "SROS: Securing ROS over the wire, in the graph, and through the kernel," in Proceedings of the IEEE-RAS International Conference on Humanoid Robots (HUMANOIDS)., 2016.

[23] M. Yim, D. G. Duff, and K. D. Roufas, "Polybot: a modular reconfigurable robot," in ICRA, 2000, pp. 514–520.

HolisafeHRC: Holistic Safety Concepts in Human-Robot Collaboration

Naresh Chowdary Chitturi[1*], Markus Ganglbauer[1], Matthias Plasch[1], Pavel Kulha[1] and Andreas Pichler[1]

***Abstract*— The success of human-robot collaboration (HRC) systems is currently facing problems related to unsolved issues in terms of safety. Standards have been established that provide a framework for implementation of such systems, but the actual safety assessment is still very difficult due to the overall complexity of HRC systems. This creates barriers for potential users and system integrators, which is a limiting factor in terms of industrial exploitation. The HolisafeMRK project addresses the safety issues in HRC and aims to develop a method for risk assessment analysis. This paper presents an overview of HolisafeMRK, a methodology for risk assessment analysis, and intermediate results.**

I. INTRODUCTION

Collaborative Robots (cobots) unlike industrial robots, work in close proximity with human co-workers, sharing common work space. Safety is vital when it comes to fulfilling collaborative operations. The safety requirements for industrial robots are described in ISO 10218-1:2011 and ISO 10218-2:2011 [7][8], which also include standards for collaborative robots. The standards state that identification of hazards is necessary and adequate measures have to be taken, in order to eliminate or reduce the risk associated with collaborative operations.

In the literature there are many approaches dealing with safety aspects in human-robot collaboration. The work in [11] presents a collision model to evaluate the safety in human-robot collaboration, which is useful in finding design parameters. The design metric for the assessment of the severity of a transient physical contact between a robot and human body region is presented in [15]. The other safety approaches are based on safe designing of shared work places, where collaborative tasks are to be carried out [12][5]. The experimental results based on ISO/TS 15066 standards for safe human-robot collaboration are presented in [13]. Most of the research work is focusing either on designing of safe manipulators or finding the impact forces on dummy human subjects. In contrast to the existing approaches, the project HolisafeMRK deals with safety aspects at application level.

The main goals of the HolisafeMRK project are:

- A simulation software called CASA tool (Computer Aided Safety Assessment) is being developed, which simulates the models that represent the manufacturing process to determine safety-relevant critical areas and possible collisions. Thus allowing the robot integrators to make the risk analysis beforehand, giving an opportunity to take proper actions before deployment of the applications in a real world scenario.
- As robots and operators work in close vicinity, tactile sensors are key elements in HRC systems [4]. Therefore, application specific tactile sensors are developed along with illuminated interfaces.
- Conventionally, human co-workers interact with the robot either with the help of teach pendants or Graphical User Interfaces (GUIs). The HolisafeMRK project introduces new interaction modalities by integrating tactile sensors directly on the robot that allow for a more fluent interaction with the robot.
- Furthermore, the developed technologies are tested on a pneumatic robot.

The contributions of this paper are as follows:

- First, a methodology for an assessment of the safety in a human robot collaboration process is presented.
- Then, the implementation concept of the described methodology against an industrial use case is provided.
- Finally, the initial results of the risk assessment by calculating the collision forces and developed capacitive sensors are presented.

The remaining part of the paper is organized as follows: In Section II, the methodology of the proposed approach is described together with the contact forces and the transferred energy relations, and the tactile sensors. The industrial use case is given in Section III. Section IV describes the implementation details and the initial results. Finally, some concluding remarks and the planned future work are given in Section V.

II. METHODOLOGY

Figure 1 depicts the work flow of the CASA tool. The simulations are performed with pre-defined trajectories. Every use-case defines trajectories that are specific to the application. Initially the robot model as well as the robot trajectories are loaded into the simulation. Then the critical points are identified during the simulation with respect to the robot pose. The associated risks are assessed by comparing the computed forces with safety limits proposed in the standards. If the collision forces are within the safety limits, then use case is safe to test it real scenario. Otherwise, various simulations are carried out to alter the robot trajectories and safety values to bring the collision forces to the safe limit. Thus safety of the human co-worker is ensured by foreseeing the potential risks and taking corresponding measures before actual integration takes place.

[1]All authors are with Profactor GmbH, Im Stadtgut A2, A-4407 Steyr-Gleink, Austria <firstname.lastname>@profactor.at

[*]Corresponding author: Naresh Chowdary Chitturi naresh.chitturi@profactor.at

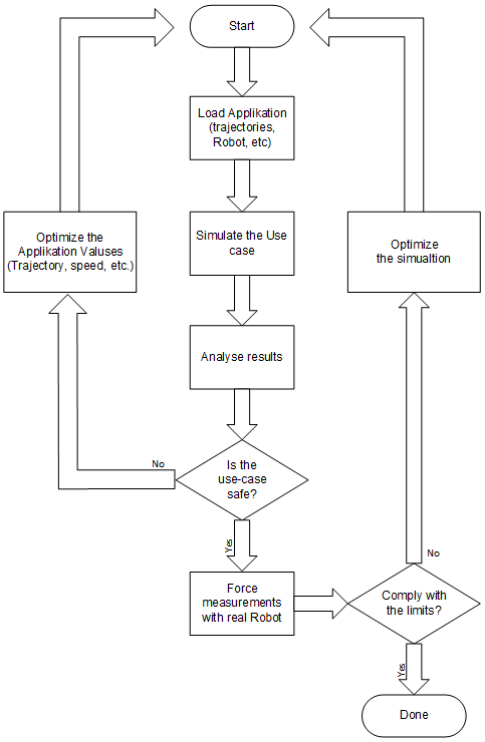


Fig. 1. CASA Tool: work flow

A. Collision Forces and Transferred Energy

A six-DOF robot arm is considered for the simulation as it is widely used. Let l_1 to l_6 be the link lengths of the manipulator; m_1 to m_6 be the masses; I_1 to I_6 be the moments of inertia; l_{c1} to l_{c6} be the distances from the each link to its center of mass; r_1 to r_6 be the radius of the cylindrical links; q_1 to q_6 be the joints angles; $\dot{q}_1$ to $\dot{q}_6$ be the velocities at the corresponding joints; u is the direction vector in which the robot end-effector is moving. Given these parameters, now the effective mass m_R of the robot is given in the Equation 1 [10].

$$m^{-1}{}_R = u^T[(J(q)M^{-1}(q)J^T(q))]u \tag{1}$$

Where $J(q)$ and $M(q)$ are the Jacobian matrix and inertia matrix of the manipulator, q is the joint angle vector, and u is the direction vector. The relation between transferred energy (E) and the collision force F is given in the following Equation 2 [9].

$$E = \frac{F^2}{2K} = \frac{1}{2}\mu v_{rel}^2 \tag{2}$$

Where K is the effective spring constant of the specific human body region. v_{rel} is the relative speed between robot and human body region. μ is the reduced mass between two-body system, which is given in the Equation 3.

$$\mu = \left(\frac{1}{m_H} + \frac{1}{m_R}\right)^{-1} \tag{3}$$

Where m_H is the effective mass of the human body region and m_R is the effective mass of the robot. The relative speed

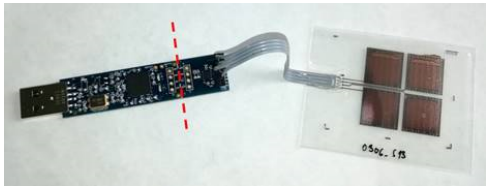


Fig. 2. Capacitive flexible sensors with four electrodes printed on PET connected to the evaluation board with TI FDC1004 capacitive interface circuit

(v_{rel}) between robot and human body region is given in Equation 4.

$$v_{rel} = \frac{F}{\sqrt{\mu k}} = \frac{pA}{\sqrt{\mu k}} \tag{4}$$

where p is the maximum allowed pressure value [9]. Effective masses and spring constants of different human body regions are given in Table I.

TABLE I
EFFECTIVE MASSES AND SPRING CONSTANTS OF HUMAN BODY REGIONS [6] [14]

Body region	Effective spring constant (k)(N/mm)	Effective Mass (M_H) (kg)
skull and forehead	150	4.4
Face	75	4.4
Neck	50	1.2
Back and shoulders	35	40
chest	25	40
Abdomen	10	40
Pelvis	25	40
Upper arm and elbow joints	30	3
Lower arm and wrist joints	40	2
Hands and fingers	75	0.6
Thighs and Knees	50	75
Lower legs	60	75

B. Capacitive Sensors

The capacitive sensors enable new methods of interaction with robots. For example, restarting an existing application without need of using a teach pendant. The work was focused on the design of a capacitive structure for proximity and touch detection, which can be applied to a robot for preliminary testing. PET (Polyethylene terephthalate) substrate was chosen due to its flexibility and as a relatively easy processable material. Jetting optimization for two different silver inks has been done with a resolution of printed lines down to 200 µm. In parallel, printing tests of conductive and insulation inks on different kinds of *AIRSKIN*® [1] have been performed. The capacitive sensor is shown in Figure 2.

III. USE-CASE DEFINITION

The industrial use case of palletizing application called Flexpalletizer from Haba Packaging GmbH is shown in Figure 3.

The set up consists of a robot (UR10), a conveyor belt to carry pancake boxes, and the pallets on which the pancake

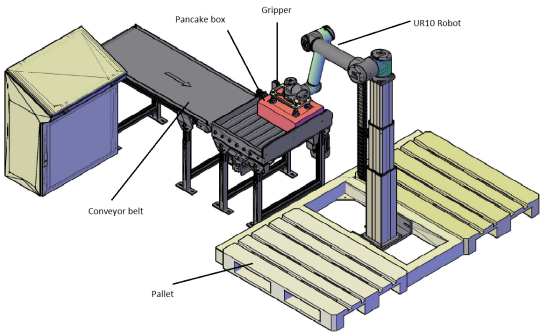


Fig. 3. Flexpalletizer Use Case: conception of robot picking pancake box from conveyor belt [3]

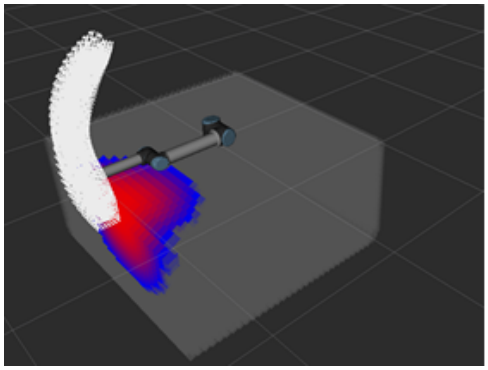


Fig. 4. Robot Visualization in Rviz with critical points (red color) along defined trajectory (white color)

boxes to be stacked. The goal of the use case is a collaborative process, where the robot picks the pancake boxes from the conveyer belt and places them on the pallets. The worker then labels them accordingly. Given the process that requires human and the robot work in close proximity, safety of the human worker is crucial. In order to perform the risk assessment, the complete process is divided into following sub tasks; 1. Robot moving from its home position to pick position 2. Picking the box with gripper manipulation. 3. Moving from pick position to place position. 4. Manipulating the gripper to place the object. 5. Moving back to home position or back to pick position. 6. Worker labeling the stickers on the box. Tasks 2 and 4 are not relevant concerning safety, as robot stays stationary. The remaining robot movement tasks could have a potential collision possibility with human co-worker sharing the workspace.

The task that is performed by human co-worker, for example, if the the human worker is trying to take an object from the pallet and the robot approaches the pallet at the same time, could lead to a potential collision scenario. Some examples of such collision scenarios could include:

- Scenario1: Robot approaching from above; could result in a head-on collision
- Scenario 2: Robot approaching from sideways; could result in collision with the arm.
- Scenario 3: Robot placing a box while worker is labeling the boxes; could press the hand against other boxes

IV. Implementation and Initial Results

A. CASA Tool

A simulation environment of ROS is used together with Kinematics and Dynamics Library(KDL). An URDF (Unified Robot Description Format) model is used to define the geometric relations between robot links and dynamics.

The visualization of the robot in Rviz is depicted in Figure 4. It shows the simplified scenario with a robot moving along the trajectory (white overlay). As the end-effector approaches to close vicinity of the work table, it is identified as potential collision area (red color). The force computations for quasi-static contact are considered only for

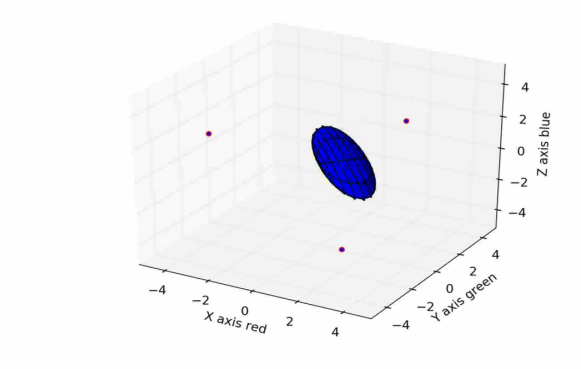


Fig. 5. Robot end-effector velocity ellipsoid

the potential collision areas. The velocity ellipsoid of an end-effector at an arbitrary pose is seen in Figure 5. The robot moves fast along the long axis of the ellipsoid compared to the short axis of the ellipsoid. Therefore, possible collision forces are greater along the large axes.

The results obtained from the simulation will get compared to the results of the dynamic force measurement device (KMG 500 KOLROBOT) for the validation. It has a spring constant of 75 N/mm, which is equal to human hand region according to the ISO/TS 15066 [9]. The comparison of the estimated forces from the robot with dynamic force measurement is given in Figure 6. The duration of the collision is 180 ms. The collision force in impact direction i.e. z direction stayed approximately at 100 N from 25 ms to 135 ms, which is with in the safety limits. After this, the robot reacted to the collision and retracted from the collision point and the collision force becomes zero.

Figure 7 shows the collision test results of UR10 robot with configured parameters such as different velocities and the force value of 150 N at tool center point. The collision tests with a speed of 50 mm/sec and 100 mm/sec have impact forces within the safety limits as specified in the standards. At other speeds, safety limits are violated indicating that a maximum velocity of 100 mm/sec is allowed for that robot end-effector pose in order to meet safety criteria.

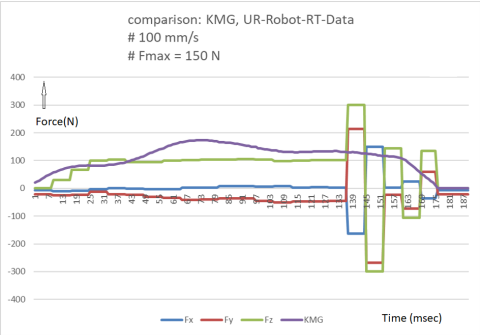


Fig. 6. Force Measurements: Dynamic force measurement device vs estimated forces from robot simulation

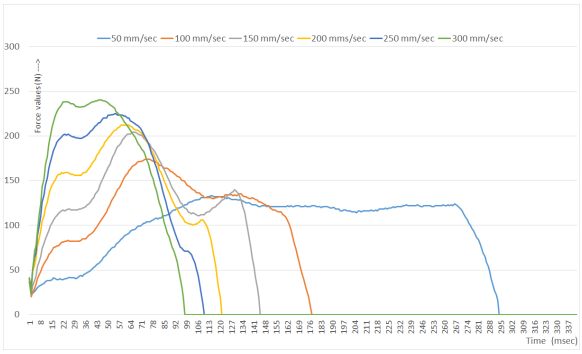


Fig. 7. Collision force measurements

B. Capacitive Sensors

Capacitive elements were printed on PET substrates using LP50 printer and industrial printing head Dimatix Spectra-128. Conductive layers were laminated with a thin PET foil, which serves as a protection layer and also as a dielectric layer for touch sensing. The capacitive structure is divided into four sections of each 15 mm x 15 mm in size, the capacitance of each section can be evaluated separately. The FDC1004 [2] was used for sensor evaluation. It is a high-resolution, 4-channel capacitance-to-digital converter for implementing capacitive sensing solutions. Each channel has a full-scale range of ± 15 pF. The FDC1004 also includes shield drivers for sensor shields, which can reduce EMI interference and help focus the sensing direction of a capacitive sensor. The electrode size allows detection of the human hand from a distance of 10 cm and reliable touch detection as is depicted in Figure 8.

V. Conclusions and Future Work

This paper presents a concept for assessment of the safety for human-robot collaborative process. The presented results are verified with real time collision tests, which are promising.

As part of the future work, the foreseeable tasks are; First, the complete scenario of the palletizing use case will be taken into the scene for identification of critical areas and computation of possible collision forces. The simulations are extended for different human body regions by taking into account of different spring constants. The tactile sensors will

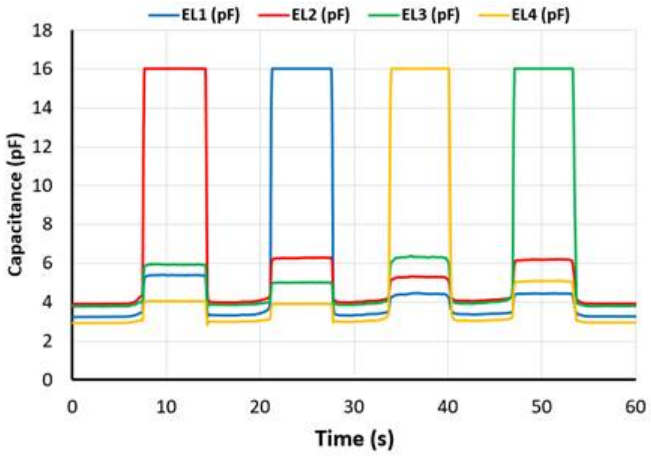


Fig. 8. Record of measurement of touch detection on segments (EL1, EL2, EL3, EL4) of the printed capacitive sensor.

be printed and tested on different materials in order to study the capacitance characteristics. Then capacitive sensors will be integrated on the robot which enable a set of functionality, for example switching of the applications, which in turn lays the pavement for the new methods of interaction. Finally, the entire application will be tested in a real scenario.

Acknowledgment

The HoliSafeMRK project is funded by österreichische Forschungsförderungsgesellschaft (FFG) under the research programme Produktion der Zukunft (PdZ).

References

[1] "Blue danube robotics," https://www.bluedanuberobotics.com/product/airskinfor-universal-ro-bots-ur5/, accessed: 27-02-2019.
[2] "Fdc1004," http://www.ti.com/lit/ds/symlink/fdc1004.pdf, accessed: 26-02-2019.
[3] "Haba packaging gmbh," https://www.haba.at, accessed: 27-02-2019.
[4] M. Fritzsche, N. Elkmann, and E. Schulenburg, "Tactile sensing: A key technology for safe physical human robot interaction," in *Proceedings of the 6th International Conference on Human-robot Interaction*, ser. HRI '11. New York, NY, USA: ACM, 2011, pp. 139–140. [Online]. Available: http://doi.acm.org/10.1145/1957656.1957700
[5] V. Gopinath, F. Ore, S. Grahn, and K. Johansen, "Safety-focussed design of collaborative assembly station with large industrial robots," *Procedia Manufacturing*, vol. 25, pp. 503–510, 2018.
[6] J. Haley *et al.*, "Anthropometry and mass distribution for human analogues," *Harry G. Armstrong Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, OH, Report No. AAMRL-TR-88-010*, pp. 33–38, 1988.
[7] ISO, "10218-1: 2011: Robots and robotic devices - safety requirements for industrial robots - part 1: Robots," *Geneva, Switzerland: International Organization for Standardization*, 2011.
[8] ——, "10218-2: 2011: Robots and robotic devices–safety requirements for industrial robots–part 2: Robot systems and integration," *Geneva, Switzerland: International Organization for Standardization*, 2011.
[9] ——, *ISO/TS 15066 - Robots and robotic devices - Collaborative robots*, ISO ISO/TS 15 066, 2016.
[10] O. Khatib, "Inertial properties in robotic manipulation: An object-level framework," *The International Journal of Robotics Research*, vol. 14, no. 1, pp. 19–36, 1995. [Online]. Available: https://doi.org/10.1177/027836499501400103
[11] S.-D. Lee, B.-S. Kim, and J.-B. Song, "Human–robot collision model with effective mass and manipulability for design of a spatial manipulator," *Advanced Robotics*, vol. 27, no. 3, pp. 189–198, 2013.

[12] N. Pedrocchi, F. Vicentini, M. Malosio, and L. Molinari Tosatti, "Safe human-robot cooperation in an industrial environment," *International Journal of Advanced Robotic Systems*, vol. 10, pp. 1–13, 01 2012.
[13] M. Rosenstrauch and J. Kruger, "Safe human-robot-collaboration-introduction and experiment using iso/ts 15066," 04 2017, pp. 740–744.
[14] D. G. Unfallversicherung, "Bg/bgia risk assessment recommendations according to machinery directive: Design of workplaces with collaborative robots," *BGIA—Institute for Occupational Safety and Health of the German Social Accident Insurance, Sankt Augustin*, 2009.
[15] B. Vemula, B. Matthias, and A. Ahmad, "A design metric for safety assessment of industrial robot design suitable for power- and force-limited collaborative operation," *International Journal of Intelligent Robotics and Applications*, vol. 2, no. 2, pp. 226–234, Jun 2018.

RNN-based Human Pose Prediction for Human-Robot Interaction

Chris Torkar[1,2], Saeed Yahyanejad[1], Horst Pichler[1], Michael Hofbaur[1], Bernhard Rinner[2]

Abstract— **In human-robot collaborative scenarios human workers operate alongside and with robots to jointly perform allocated tasks within a shared work environment. One of the basic requirements in these scenarios is to ensure safety. This can be significantly improved when the robot is able to predict and prevent potential hazards, like imminent collisions. In this paper, we apply a recurrent neural network (RNN) to model and learn human joint positions and movements in order to predict their future trajectories. Existing human motion prediction techniques have been explored in a pseudo scenario to predict human motions during task execution. Building upon previous work, we examined their applicability to our own recorded dataset, representing a more industrial-oriented scenario. We used one second of motion data to predict one second ahead. For better performance we modified the existing architecture by introducing a different output-layer, as opposed to common structures in recurrent neuronal networks. Finally, we evaluated the artificial neuronal network performance by providing absolute positional errors. Using our method we were able to predict joint motion over a one second period with less than 10 cm mean error.**

I. INTRODUCTION

Robots are widely used in different fields due to their precision, reliability, strength, and speed. Initially, in industrial applications, robots operated separated from humans in isolated areas. With advances in technology and the necessity for coexistence of robots and humans (e.g., medical application, service robots, collaborative production lines), a new era of human-robot interaction has emerged. Human-robot collaboration is one of the key aspects of future industrial manufacturing. When humans work closely with robots, the safety of the human becomes an important issue [19].

No matter how accurate and safe a system is designed, continuous monitoring is still required to ensure safety in collaborative scenarios. Naturally, perception plays an important role, identifying hazardous scenarios by using various types of advanced sensors. For instance, robot-integral force and torque sensors are used to detect collisions in order to immediately stop the robot. In addition, tactile and touch sensors can be used to detect collision areas, to further improve a robot's reactive capability. Quite recent research showed how proximity sensors even can detect an approaching object at closer ranges and prevent a collision in advance [16], [18]. Similarly, many other sensors such as conventional cameras, RGB-D cameras, time-of-flight cameras, and laser scanners can be used to monitor the environment allowing for prevention of hazardous situations. However, performing sensory data acquisition, analysis, and fusion is a computationally expensive task. Therefore, in presence of fast robot and human motions, the robot may still need extra time for a reliable perception of the environment. By predicting human and robot motions in advance, the robot can foresee hazardous situations and adjust the robot movement trajectory or speed according to safety standards [8].

Advances in machine learning and deep neural networks brought new forms of scene understanding. Miseikis et al. [14], [15] showed how to find a robot and localize its 3D joint positions using a single image applying a convolutional neural network. With an increasing number of available datasets containing a variety of human motions [7], the momentum of applying machine learning to human motion related tasks increases. For instance it is possible to extract the 2D joint positions of multiple humans from an image or video in a timely manner [1]. By using recurrent neural network topology, [13] showed that a short term human motion prediction is possible based on observed human motion patterns. Even more recent findings [3] show smoothly synthesized long-term motion and short-term prediction of human motion. As shown by [4], accurate motion predictions have been accomplished by using action-specific motion patterns to train a time-series-aware neural network to predict human motion.

These methods build the foundation to incorporate motion prediction in robotic systems and enable them to plan their own movement ahead of time and avoid risky situations such as collisions with other humans or robots in collaborative scenarios. In this paper, we investigate if existing human motion prediction, using a RNN, is in principal applicable to collaborative human-robot scenarios. The goal is to evaluate the prediction accuracy of human joint motion for up to one second. To achieve this we adapted an existing RNN topology and evaluated the performance on our dataset representing a simple industrial assembly task.

II. RELATED WORK

Here we provide an overview of the most relevant literature. Our work builds upon prior research on human-robot collaboration and human-motion modelling using artificial neural networks.

Human-robot collaboration - Previous works in the field of safe robot operation in conjunction with human interaction e.g. [5], [10], have led the consideration and characterization of safety requirements. Safety is considered as the most important design criterion in drafting a new robotic system or implementing an industrial robot in a manufacturing process. Furthermore, it is desired to account for more convenient robot interaction with humans and more efficient operation.

[1] Joanneum Research - Robotics first.lastname@joanneum.at
[2] Alpen-Adria-Universität Klagenfurt first.lastname@aau.at

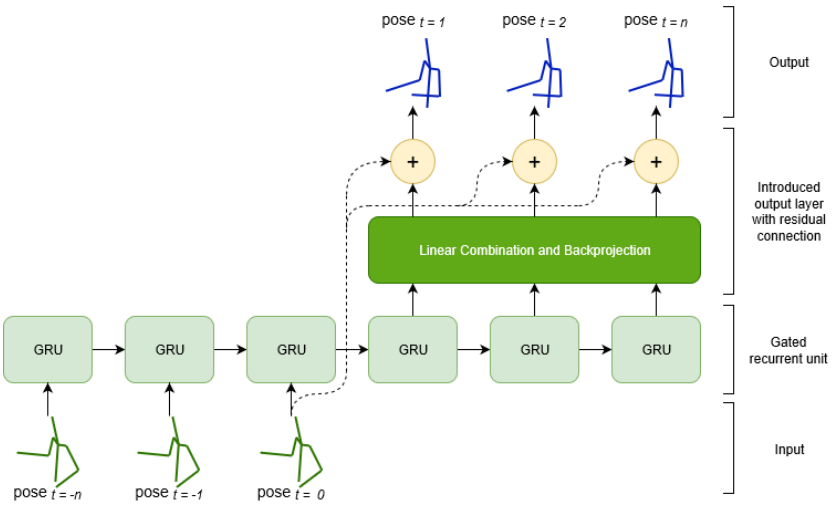


Fig. 1. Proposed architecture with extended back projection layer (green box). Green skeletons represent the input sequence and blue skeletons represent the predicted output. The dotted line show the residual connection added to the final pose output.

Similar related work has been published by [11], [12] who propose human motion prediction methods for human-robot collaborative tasks. Being able to foresee human motion in a short time span, prior information can be used to avoid hazardous situations up front.

Liu and Wang [11] intended to model a product assembly operation as a sequence of tasks. Decomposing an assembly operation in different tasks with specific motion patterns allowed them to map their assembly problem to a Hidden Markov Model. Hidden Markov Models are well suited for discrete sequence models. They showed the feasibility of applying a Markov Chain to human motion on a task level view to predict the next likely task. Lacking detailed positional joint information in their approach, [12] proposed a framework that allows robots and humans to operate safely in close proximity. Using the Gaussian Mixture Model representation of human motion they where able to predict the upcoming work space occupancy for a certain time span. Applying the existing trajectory optimization methods, they showed the practicability using a work space occupancy map to compute collision free trajectories. Providing a framework to offer real time motion recognition and prediction they assessed the frameworks capabilities by conducting experiments to measure the human interference during task execution.

Human motion modeling using artificial neural networks - Application of artificial neural network in human motion modeling and estimation has grown drastically with the introduction of datasets containing annotated human motion and pose samples [7]. Commonly used by [2], [3], [4], [13], they all focused on the task of human motion prediction and long-term synthesis. By pursuing the achievements in human motion prediction using Deep RNN, [13] introduced simple and scalable RNN architecture. Using a single Gated Recurrent Unit (GRU) rather than concatenated Long Term Short Memory cells [3] they showed that the much simpler architecture is capable of achieving similar performance. Simplifying the architecture increases performance and allows faster training. Applying ideas from [6] they where able to improve first frame discontinuities significantly. Rather than modeling human poses and motion, they added a residual connection to the architecture which forced it to model joint velocities instead. Gui et al. [4] showed, most recent improvements in the area of human motion modelling and prediction. They used a similar approach as [13], but introduced a global discriminator which examine the quality of the prediction. Inspired by Generative Adversarial Networks (GAN) they adapted this concept to the motion prediction domain and resulted in a motion GAN. GANs have shown great progress in sequence generation problems since the predicted sequence is judged from a global perspective. Jointly optimizing the discriminator together with the predictor, the predictor's performance is not only measured by the loss function, but further the predicted sequence is rated quality-wise from a global perspective.

Although state of the art showed promising results by applying RNNs in the field of human motion prediction, RNNs have not been extensively exploited in the field of human-robot collaboration. Due to the lack of available datasets containing actions similar to assembly tasks in an industrial environment, we conducted experiments using our own data samples.

III. METHOD

Recent work in human pose prediction mainly focuses on casual tasks, e.g walking, discussions, eating and etc. It showed promising results for motion prediction in a short-term manner and qualitative smooth motion synthesis in a long-term scenario. However, for the distinct scenario of motion prediction in an industrial environment, we did not find publicly available datasets which can be used for human-robot collaboration. Further more, we saw the potential to improve predictions on our own dataset by

using a more complex back projection layer in the network architecture.

Our approach - As Figure 1 shows, we apply a similar structure as [13], who based their proposed method on the advantage of modelling velocities in the RNN [6]. We focused on the importance of high accuracy in the first frames to accommodate for continuous prediction. Applying a RNN in a sequence prediction task requires the output to have the same dimensionality as the input. This means in our case the input is given by a sequence of observed body poses in the joint angle domain (green skeletons) and the output is a continued sequence of the input (blue skeletons). The input size represents the number of joint angles we consider for reconstruction of the body skeleton, see Figure 3. The output size is dependent on the number of units in the GRU. Using a higher number of internal units allow to extract more features from the input, although it mismatches the desired output-size, e.g. using 1024 outputs instead of 15 output angles. To apply RNN to sequence prediction problems, the output size is projected from the GRUs output units to the original dimensionality. Commonly this is done by a linear layer which back-projects frame by frame. Temporal information is only contained in the RNN and the linear back-projection is only used for size adaptation. We saw potential in expanding the back-projection layer to incorporate temporal information. This was done by feeding all GRU output information to a fully connected layer. Dependent on the sequence, the fully connected layer is able to combine all available information and fuse this to the output sequence. All information fusing in one layer (Linear Combination and Backprojection) allows this layer to weight the extracted information from the GRU in a second instance. Applying a residual connection forces the overall structure to model joint velocities rather than poses. Due to the changes in the network structure, the same residual architecture as used by [13] was no longer possible and is changed to use the last pose of the input sequence as input for the residual connection (dotted line).

Setup - We conducted our experiments with our own recorded dataset to evaluate the applicability of the proposed method in a human-robot collaboration scenario. A simple rectangular object with one screw in each corner was used to simulate the assembly procedure. Each screw was tightened by reaching for the screwdriver, tightening and placing the screwdriver again in the rest position, see Figure 2. The recorded dataset contains joint positions in $[x, y, z]$ format for 9 joints on the upper human body. The recorded joint positions resemble a simplified human torso skeleton, see Figure 3. A sequence of 120 seconds of motion capture footage, using OptiTrack motion capture system [17], was recorded at 120 Hz.
The recorded data was pre-processed applying error removal and smoothing. After this, the pre-processed raw data was fed to a skeleton generator to convert joint positions to the joint angle space. Our proposed architecture was trained on the converted dataset.

Fig. 2. Schematic representation of the setup.

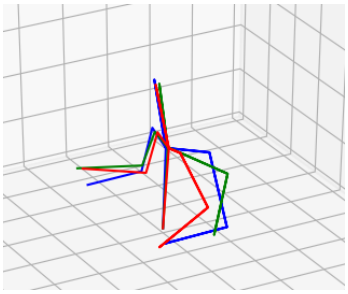


Fig. 3. Computed torso skeleton from joint angle space.

Implementation Details The results presented in this paper where obtained using the proposed method for an input sequence of 120 frames followed by 120 predicted frames. Experiments with the structure showed that 512 units in the GRU are sufficient for our requirements. The recurrent neural network is implemented using Tensorflow and was trained for 12h on a Nvidia GTX 1050Ti GPU.

IV. RESULTS

Finally we present the achieved results of our proposed method for motion prediction. Due to the small dataset we used, it was not feasible to split the dataset into test and train data. However, the results should be representative enough to evaluate the applicability of the proposed method. We considered one second as input sequence to predict one second ahead. We measured the error as mean angle error as it has been done by [4], [13] to be able to compare it to their results. Direct comparison to their results is not possible since they consider full body motion compared to torso motion in our tests. Figure 5 shows the development of the mean angle error over a time span of one second. It indicates low error for the first frames of prediction and increases as time evolves. Showing a faster error development in the first 200 milliseconds, error develops slower and linear after 200

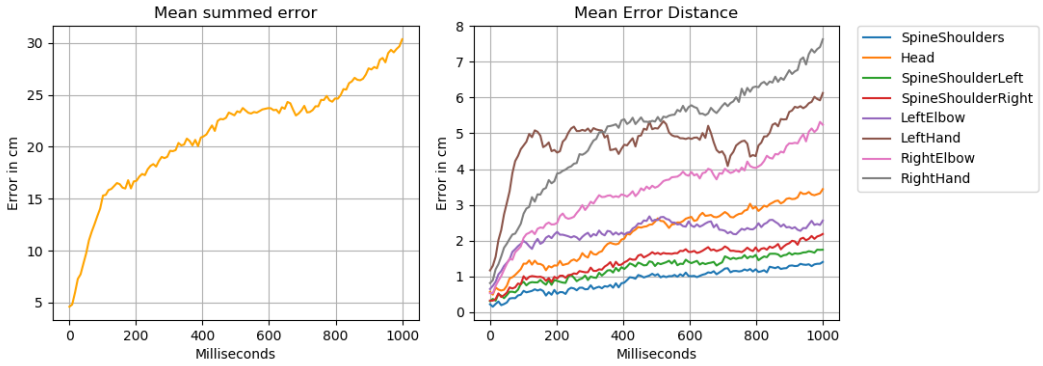


Fig. 4. Absolute errors considering a average human person with 180 cm body height. Left: Shows the summed error of all joints. Right: Shows the error of each joint individually. The results represent the average performance of 60 randomly selected samples.

milliseconds.

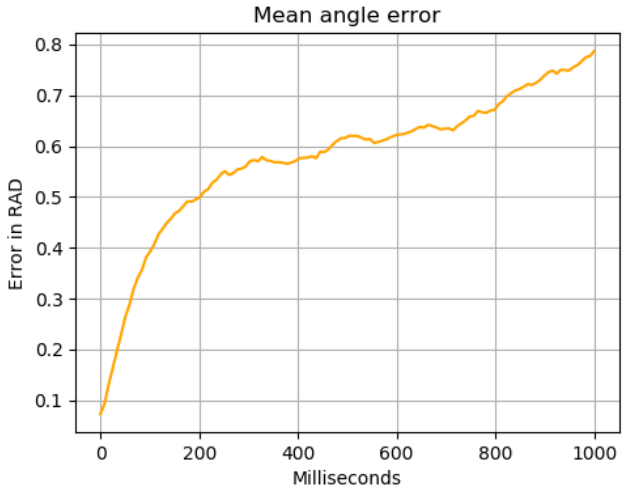


Fig. 5. Mean angle error plot of the proposed method as commonly used by previous work.

Figure 4 was used to asses the achieved performance of the proposed method. The left plots validate the results from Figure 5 by showing a fast progressing error up to 200 milliseconds and slower development after. The right plot reveals highest error for hand joint prediction. Still the maximum error is less than 8 cm at its peak. The higher error for the hand joints can be explained by assuming the highest velocity for hand joints. [3], [4], [13] commonly provide their results in mean angle error, for more easy comparison Figure 5 is provided.

V. CONCLUSIONS

We demonstrated the potential of sequence to sequence artificial neural networks for human robot collaboration scenarios. The obtained results are still in a early stage, but support our vision to use the proposed architecture as scalable human motion prediction solution in human-robot collaborative tasks. However, it allows to assess this technology for usability and shows promising performance. The introduced architecture offers sufficient accuracy with low first frame errors, allowing for motion prediction on continuous data streams. Continuous prediction results can be used to dynamically adjust motion trajectories of robots for collision avoidance. Due to the scarcity of industrial datasets, the next planned steps include to improve the dataset and represent a more general motion model. Additionally, we have an ongoing work to expand our current dataset and split it into test and train sequences. This would allow to verify the presented performance and asses the relevance for industrial integration. Furthermore, for safe industrial integration, it would be valuable to provide a certainty measure [9]. The certainty of a predicted sequence could give a rough estimate of how probable the predicted sequence is. We plan to cover these points in our future work.

ACKNOWLEDGMENT

This work has been supported by the Austrian Federal Ministry for Transport, Innovation and Technology (BMVIT) programme ICT of the Future, managed by the Austrian Research Promotion Agency (FFG), under grant no. 861264.

REFERENCES

[1] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1302–1310, 2017.

[2] K. Fragkiadaki, S. Levine, and J. Malik, "Recurrent network models for kinematic tracking," *CoRR*, vol. abs/1508.00271, 2015. [Online]. Available: http://arxiv.org/abs/1508.00271

[3] P. Ghosh, J. Song, E. Aksan, and O. Hilliges, "Learning human motion models for long-term predictions," *CoRR*, vol. abs/1704.02827, 2017. [Online]. Available: http://arxiv.org/abs/1704.02827

[4] L.-Y. Gui, K. Zhang, Y. Wang, X. Liang, J. M. F. Moura, and M. Veloso, "Teaching robots to predict human motion," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, October 2018.

[5] S. Haddadin, A. Albu-Scheffer, and G. Hirzinger, "Requirements for safe robots: measurements, analysis and new insights," *The International Journal of Robotics Research The International Journal of Robotics Research*, vol. 28, pp. 11–12, 01 2009.

[6] J. N. Ingram, K. P. Körding, I. S. Howard, and D. M. Wolpert, "The statistics of natural hand movements," *Exp Brain Res*, vol. 188, no. 2, Jun 2008, 18369608[pmid]. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/18369608

[7] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1325–1339, 2014. [Online]. Available: https://dblp.uni-trier.de/db/journals/pami/pami36.html

[8] ISO, *ISO/TS 15066:2016: Robots and robotic devices – Collaborative robots*. Geneva, Switzerland: International Organization for Standardization, Feb. 2016.

[9] V. Kuleshov, N. Fenner, and S. Ermon, "Accurate uncertainties for deep learning using calibrated regression," *CoRR*, vol. abs/1807.00263, 2018. [Online]. Available: http://arxiv.org/abs/1807.00263

[10] D. Kulic and E. Croft, "Pre-collision safety strategies for human-robot interaction," *Auton. Robots*, vol. 22, pp. 149–164, 01 2007.

[11] H. Liu and L. Wang, "Human motion prediction for human-robot collaboration," *Journal of Manufacturing Systems*, vol. 44, pp. 287–294, 08 2017.

[12] J. Mainprice and D. Berenson, "Human-robot collaborative manipulation planning using early prediction of human motion," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 299–306.

[13] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," *CoRR*, vol. abs/1705.02445, 2017. [Online]. Available: http://arxiv.org/abs/1705.02445

[14] J. Miseikis, I. Brijacak, S. Yahyanejad, K. Glette, O. J. Elle, and J. Tørresen, "Transfer learning for unseen robot detection and joint estimation on a multi-objective convolutional neural network," *CoRR*, vol. abs/1805.11849, 2018. [Online]. Available: http://arxiv.org/abs/1805.11849

[15] J. Miseikis, P. Knöbelreiter, I. Brijacak, S. Yahyanejad, K. Glette, O. J. Elle, and J. Tørresen, "Robot localisation and 3d position estimation using a free-moving camera and cascaded convolutional neural networks," *CoRR*, vol. abs/1801.02025, 2018. [Online]. Available: http://arxiv.org/abs/1801.02025

[16] S. Mhlbacher-Karrer, M. Brandsttter, D. Schett, and H. Zangl, "Contactless Control of a Kinematically Redundant Serial Manipulator Using Tomographic Sensors," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 562–569, Apr. 2017.

[17] NaturalPoint, Inc., "Optitrack," *[Online]. Available: http://www.naturalpoint.com/optitrack. [Accessed: Feb. 2019]*.

[18] S. E. Navarro, S. Koch, and B. Hein, "3d contour following for a cylindrical end-effector using capacitive proximity sensors," in *IROS*. IEEE, 2016, pp. 82–89.

[19] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human-srobot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, 03 2018.

Adaptive Loading Station for High-Mix Production Systems

Alexander Raschendorfer[1], Michael Hofmann[2], Florian Pauker[3], Marcel Fuschlberger[1], Martin Kunz[1]

***Abstract*— This paper presents a loading station for high-mix production systems in a production shift without factory workers. A UR10e is used to load and unload a pallet, fitted on an Autonomous Guided Vehicle (AGV). The robot is supported by the software XRob and cameras for visual detection of the raw parts as well as the raw part trays. Additionally the cameras are used to correct the position of the AGV. The process itself is orchestrated by the workflow engine centurio.work.**

I. INTRODUCTION

The motivation behind creating an adaptive, automated loading station lies in two main fields of research. Firstly creating a system that can handle a production during a factory shift without workers and a limited supply of trays for handling raw parts. Thus limiting and reducing the amount of trays for the part handling on the shop floor level. Reduced amounts of trays required for work piece handling lower the costs of the workflow as well as the environmental footprint. Less storage areas are required and the trays can easily be reused for similar raw parts. Pallets and handling equipment for raw parts is often highly specialized and only serves the single purpose to bring the piece to the first process stage in the manufacturing area. The trays used in this use case carry the raw parts as well as the finished product after machining. Secondly using the centurio.work engine to orchestrate the loading of a modular pallet with trays for individual raw parts in combination with a visual detection of the raw parts by cameras mounted onto the robot.

For demonstration reasons three parts were defined for the first stage of the use case. The three parts are chess pieces (Bishop, Rook and Pawn) and are all turned on an EMCO MaxxTurn 45. The raw parts are cylinder of varying diameter and height.

II. DESCRIPTION OF THE LOADING STATION

This section describes the hard- and software used for the loading station.

A. Layout and Hardware

The basic layout, as seen in Fig. 1 of the use-case consists of an UR10e-Robot by Universal Robots, a Neobotix AGV and Trays, that can be fitted on a special pallet, that is moved by the AGV and used within the production of the ”Pilotfabrik 4.0” of the Vienna University of Technology. The UR10 of the e-Series is equipped with a range of built-in safety functions as well as various control signals to or from the electrical interface, to connect to other machines and devices [12]. The gripper used is a Robotiq 2F for Universal Robots. It is an electric, adaptive two-finger gripper with a maximum clamping force of 125 N.

The Neobotix MP-400 is an Autonomous Guided Vehicle (AGV) by the company Neobotix. Its maximum payload is up to 150 kg. For the AGV to work, no guidewires in the floor for steering are necessary. Instead the stations and roads between these stations are virtually designed within the Neobotix software. The whole environment can easily be designed within the software provided by Neobotix using the AGV's integrated vision system. Once everything is set up the AGV can move freely between predefined stations, utilizing the routes and maneuvering autonomously around obstacles in its way [5]. The AGV carries a pallet, that is used within the production of the ”Pilotfabrik 4.0”, and that can be equipped with individual trays for holding the raw parts as well as the finished product.

The trays are designed to fit several needs. At first the trays need to safely carry the raw parts and the finished product. Additionally the trays need a handle, so the UR10e can grip and position the trays on the pallet. Depending on the diameter of the raw part an individual tray, fitting that diameter, is used. Lastly the tray needs to contain information about the product as well as information about the positioning of the tray so the UR10 and the connected vision systems can locate the trays. The trays are created using the FDM 3D-printing method.

B. XRob

The XRob software framework[9] enables the creation of complex robot applications within fewer minutes. It builds on unique, easy-to-use features that significantly speed up commissioning and make the operation more costefficient and flexible than common programming methods. The special software architecture allows easy and intuitive creation of processes and configuration of the components of a robot system via a single user interface. Fig. 2 provides an overview on the software components within the XRob framework.

Object Recognition: The aim of 3D Object Recognition is to localize the pose and position of an object of interest in the scene. Given the 3D model of the object, the goal is to find a correct transformation (6DOF) of the 3D model in the point cloud reflecting the current scene. A 3D model can be obtained either by 3D reconstruction or based on

[1]Alexander Raschendorfer, Marcel Fuschlberger and Martin Kunz are with the Austrian Center for Digital Production, Seestadtstrasse 27, A-1220 Vienna, Austria a.raschendorfer@acdp.at

[2]Michael Hofmann is with PROFACTOR GmbH, A-4407 Steyr-Gleink, Austria michael.hofmann@profactor.at

[3]Florian Pauker is with University of Vienna, Faculty of Computer Science Workflow Systems and Technology Group, Waehringerstrasse 29, A-1090 Vienna, Austria florian.pauker@univie.ac.at

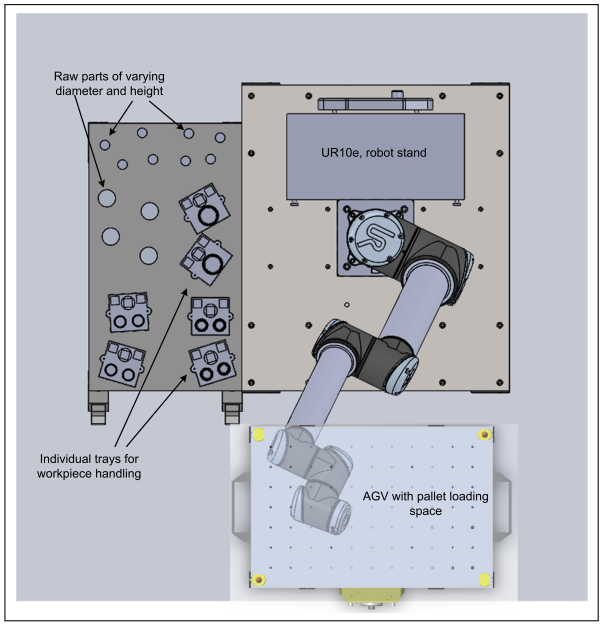


Fig. 1. Layout of the loading station

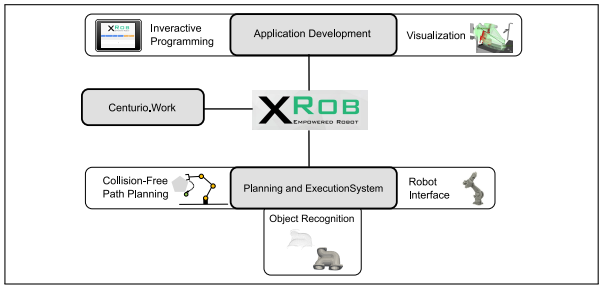


Fig. 2. Overview on the software components within the XRob framework.

the CAD model of the object, which is transformed into a point cloud during configuration. The 3D Object Recognition module is based on the Randomized Global Object Localization Algorithm (RANGO) [3], which extends the Random Sampling Algorithm [6]. Resulting object detections are used to plan collision-free robot movement paths for object manipulation. The accuracy of the 3D object recognition approach described above greatly depends on the sensor data quality and on the sizes of the objects of interest. Especially for very small objects, e.g. screws, 3D object recognition does not lead to reasonable detection results. To overcome these limitations, the XRob software framework supports 2D template matching and the detection of matrix markers [2] [1]. Matching templates can be configured on runtime, by positioning the sensor accordingly to record a template image.

Collision-Free Path Planning: The results of the Object Recognition and Localization system are used to plan and calculate collision-free robot manipulation paths [10] to enable handling of the detected objects. Based on predefined grasp as well as deposit points on the CAD model of the objects, the manipulation planner determines how the object can be grasped. All object localizations as well as the available workspace environment data are considered for collision checks. The path planning system is based on the Open Motion Planning Library (OMPL) [11] and follows rapidly-exploring random trees (RRT) [4] approach.

Robot Interfaces: To facilitate communication with the robotic system, the XRob framework provides a uniform communication interface, which can be extended in a Plug-In like fashion to support robotic systems of different vendors.

Application Development: The XRob software framework provides an intuitive user interface for application development, which includes an interactive programming environment, and software modules to simulate and visualize robotic movement paths as well as data acquisition via sensors.

C. Adaptive Loading Station for High-Mix Production Systems

The in XRob included process planner as described in II-B is one of the modules of XRob which are not easily reconfigured via programming. Instead it needs pre defined grasp points, the robot kinematics and a collision scene of the environment surrounding workcell. As a change of to be handled parts can not be easily programmed directly at the station a remote reconfiguration has to be done. The reconfiguration has to consider the consistency of the grasping points and the object model to be detected.

D. centurio.work

centurio.work is a process based framework for realizing a integration platform for manufacturing scenarios. Processes are widely used in companies for describing the workflow of business matters. In production environment they are not used explicit. The idea is using processes as unified context in a factory for orchestration bringing the flexibility from business down to shopfloor. Fig 3 shows the decomposition of the automation pyramid and how process can be used for connecting different functions on different layers.

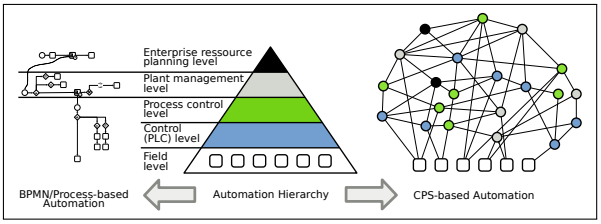


Fig. 3. Process based automation [8]

The architecture of centurio.work is shown in figure 4. It consists of several independent containers with defined information flow. Basically the framework differentiate between static and dynamic information. In the **Ressources** context all necessary information for process execution is stored. In this case the workflows and the robot programs, and information for XRob. The **Orchestration** context is important during runtime. The Process Execution as well as the Data Provisioner are located in this container. The Data Provisioner realizes the connectivity to Data Sources

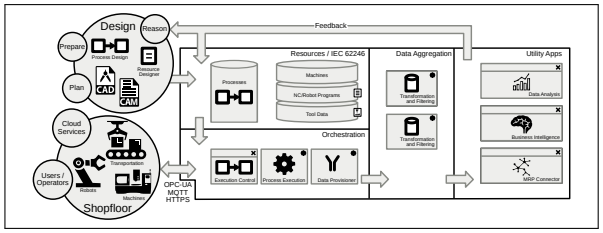


Fig. 4. centurio.work architecture [8]

like robots from different manufacturers. **Data Aggregation** container offers different solutions for storing the generated data during process execution in different forms. The idea is, that every application in the **Utility Apps** container has its own storage so no transformation is needed. Additionally a **Feedback** loop is integrated to improve static information during design.

III. Workflow and Communication

The workflows for orchestration are based on the approach presented in [7]. Atomic functions of the robot are modelled as single processes. The main processes are displayed in figure 5. The process itself is designed linear since the process steps necessary each require a successful step beforehand.

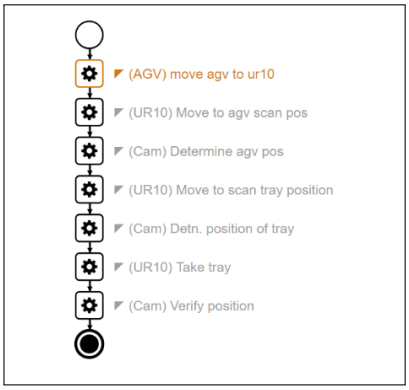


Fig. 5. centurio workflow of the loading process

Centurio.work starts with sending move orders to the AGV. The AGV has a fixed parking position right next to the loading station. The AGV needs to position itself within a small tolerance so the next processes can run on a stable basis. The process on the centurio.work level is finished once the AGV reports its final parking position. Next the UR10e is directed to move the mounted cameras over the pallet transported by the AGV. In the following step the camera determines the exact position of the pallet to get a matrix with correction values for the placing of the trays. This correction needs to be only done once for each loading or unloading process since the AGV does not move in the process. Once the position is determined the workflow engine moves the UR10 and its cameras to the area where the trays are located. The correct tray is selected and the exact position scanned. XRob provides the collision free paths so a safe picking of the tray is guaranteed. Centurio.work gives the order to execute the movements. The tray is placed on the is pallet. Once the trays are placed the same procedure starts with the raw parts. centurio.work saves the position of the trays on the pallet and so can easily place the right raw parts into the trays on the pallet.

In order to achieve a high flexibility, functions have been implemented that allow several raw parts to be handled by the softwarepieces. Centurio.work can send a container, containing a CAD file with predefined grip positions, to XRob.That way XRob can detect any form of raw part as long as it is within certain predefined boundaries. This allows for a very high flexibility on the shop floor and lot size one production. Changing raw parts can simply be pushed by centurio.work to the XRob visual recognition of the software with no worker interaction needed.

The communication between centurio.work and XRob is handled by REST (Representational State Transfer) calls. The corresponding module for communication in centurio.work is the Data Provisioner. One example is the ”GET /AGVPos” call. XRob returns the position of the AGV in Euler coordinates (rx, ry, rz, x, y, z) to the workflow engine.

IV. Conclusion and Future Work

The hughe flexibility in the system is a huge benefit of the presented use case. There are however a few design problmes of the whole system that shall be addressed. Currently the loading as well as the unloading process requires the AGV to be parked next to the loading station. This creates a dead time for the AGV itself without any productivity. The AGV cannot be used for any other tasks within the factory. Within a next iteration the loading station should be adapted to load and unload the pallet on a dedicated space without the AGV needed.

V. Acknowledgment

This work has been partially supported and funded by the Austrian Research Promotion Agency (FFG) via the Austrian Competence Center for Digital Production (CDP) under the contract number 854187.

References

[1] F. Bergamasco, A. Albarelli, and A. Torsello, “Pi-tag: a fast image-space marker design based on projective invariants,” *Machine vision and applications*, vol. 24, no. 6, pp. 1295–1310, 2013.

[2] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, “Automatic generation and detection of highly reliable fiducial markers under occlusion,” *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.

[3] P. Gmbh, “Candelor: A computer vision library for 3d scene interpretation,” 2017, http://candelor.com.

[4] J. J. Kuffner Jr and S. M. LaValle, “Rrt-connect: An efficient approach to single-query path planning,” in *ICRA*, vol. 2, 2000.

[5] *Neobotix Bedienungsanleitung MP_400*, NEOBOTIX, 04.2017.

[6] C. Papazov and D. Burschka, “An efficient ransac for 3d object recognition in noisy and occluded scenes,” in *Asian Conference on Computer Vision*. Springer, 2010, pp. 135–148.

[7] F. Pauker, I. Ayatollahi, and B. Kittl, “Service orchestration for flexible manufacturing systems using sequential functional charts and opc ua,” *Dubrovnik*, vol. 9, pp. 11–09, 2015.

[8] F. Pauker, J. Mangler, S. Rinderle-Ma, and C. Pollak, “centurio. work-modular secure manufacturing orchestration,” 2018.

[9] A. Pichler, S. C. Akkaladevi, M. Ikeda, M. Hofmann, M. Plasch, C. Wgerer, and G. Fritz, "Towards shared autonomy for robotic tasks in manufacturing," *Procedia Manufacturing*, vol. 11, pp. 72 – 82, 2017, 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2351978917303438

[10] A. Pichler and C. Wögerer, "Towards robot systems for small batch manufacturing," in *2011 IEEE International Symposium on Assembly and Manufacturing (ISAM)*. IEEE, 2011, pp. 1–6.

[11] I. A. Sucan, M. Moll, and L. E. Kavraki, "The open motion planning library," *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 72–82, 2012.

[12] *Universal Robots e-Series User Manual*, Universal Robots, 2018.

Workflow-based programming of human-robot interaction for collaborative assembly stations*

Roman Froschauer[1] and René Lindorfer[2]

Abstract— **In certain domains manual assembly of products is still a key success factor considering quality and flexibility. Especially when thinking of flexibility traditional, fully automated assembly using specialized robot stations is mostly not feasible for small lot sizes due to high costs for programming and mechanical adaptations. In the last years collaborative robots (cobots) entered the market to broaden the way for robot-assisted manual assembly. The idea was to use the robot for small repetitive tasks at the manual assembly station and keep the human factor for dealing with flexibility. Unfortunately most of the new cobots came with the same programming system as their ancient relatives. Thinking of human-robot collaboration these traditional approaches do not consider the human factor at the assembly station. Therefore, this paper presents a new approach, called *Human Robot Time and Motion* (HRTM) providing a modeling language providing generic basic elements which can be performed by a human worker or a robot. Correspondingly a workflow-oriented programming model and a prototypical development environment featuring BPMN and MQTT is presented.**

I. Introduction

The number of different variants of products offered by companies is increasing [12]. Therefore, the typical lot size decreases resulting in a demand for more flexible and adaptable production facilities [15]. Mass production companies get the same problem as SMEs (small and medium enterprises), many products with a small lot size down to one [13]. The use of industrial robots in such a flexible production is hardly or not at all possible with today's programming methods [25],[9]. Therefore, human-robot-collaborations (HRC) are getting more and more important [3]. At present working steps of human workers and robots can only be modeled separately. A possible method for human workers is MTM (Methods Time Measurement) [4]. Typically, working steps of robots are not modeled at all, but more programmed using offline programming methods. Nevertheless working steps of robots may also be modeled with RTM (Robot Time and Motion) [24]. Generally, working steps of human workers can be much more complex than those of robots. As a result of this MTM provides much more possible movements than RTM. In case of collaborative robots a human worker and a robot perform actions simultaneously. Therefore, the interaction between both should be modeled [5]. Such collaboration between humans and robots has already been researched through simulations [21]. Additionally, a distinction between a work instruction for a human worker or a robot has to be made. In order to enable comprehensive modeling of working steps in an intuitive way a graphical representation is necessary, which also can be used as work instruction itself to guide human workers through their working steps [11]. A robot can be programmed in different ways but the core elements are always a point and orientation which have to be reached [7], [29]. In order to enable unified modeling of human and robot movements, these two different approaches of modeling a sequence of working steps have to be consolidated. Furthermore the communication between all involved workers (human or robots) has to be considered within the workflow.

The visualization of a workflow is just as important as its model. Different styles are common to visualize a workflow or a process for different fields of application [17]. Due to their visualization style, BPMN (Business Process Model and Notation), AD (Activity Diagram) and EPC (Event driven Process Chain) are appropriate for a detailed check. Additional requirements must be met to visualize HRC workflows.

This paper is organized as follows. In the next chapter we discuss relevant state of the art focusing on workflow modeling for industrial assembly purpose. In chapter III we propose a complete modeling hierarchy featuring a meta-model for collaborative assembly workflows and a specific example. In chapter IV a corresponding development environment is presented using a simple Pick&Place scenario. Finally open issues and future work are discussed.

II. State of the art

Common languages and methods describing workflows can be divided into two groups: (1) The first group of methods deals with the analysis of the workflow itself and focuses on the detailed mapping of the workflow so that the timing/scheduling can be optimized accordingly. (2) The second group focuses on the general visualization of the workflow, whereas timing is only of secondary importance. Those, which have future potential for a workflow description and are well-established in research or industry, are described in this section.

A. Workflow analysis

To describe arbitrary operations of human workers the *MTM* can be used. Its modeling elements are divided into basic movement elements which are applicable for a manufacturing workflow. As a result, each motion sequence

*This work was supported by the Austrian Research Promotion Agency within the 6th COIN - Cooperation and Innovation programme under grant agreement nr. 856362

[1]Roman Froschauer is with the School of Engineering, University of Applied Sciences Upper Austria, 4600 Wels, Austria roman.froschauer@fh-wels.at

[2]René Lindorfer is with the School of Engineering, University of Applied Sciences Upper Austria, 4600 Wels, Austria rene.lindorfer@fh-wels.at

can be precisely described. By setting different attributes, a basic movement element can be adopted to specific requirements/scenarios. These attributes influence the standard duration time and the difficulty of each basic element. Therefore, all workflows can be accurately analyzed and optimized. *MTM* has been continuously developed further. Thus, several versions have been designed which have been optimized for different application domains [1], [4].

Robots are not able to perform the same motions as human workers. In contrast, they have mechanical constraints, limited workspace and intelligence. As a result, the *RTM* method was developed. *MTM* was used as a basis and got accordingly adapted and extended. Consequently, it is possible to precisely model the motions of a robot and compare the robot's and human's performance time. Similar to *MTM*, attributes influence the standard duration time and the difficulty of each basic movement element. Since the publication of *RTM* it was constantly updated. The latest version was extended with mobility elements [23], [24], [20]. Collaborative modeling elements are still missing.

B. Workflow visualization

The Unified Modeling Language (UML) is mainly used in the field of software development. In UML2.0 different diagram styles are defined, the Activity Diagram (AD) is used to visualize how a system realize a particular behavior. The main elements of an activity in the AD are *Action* elements which are connected with *Control Flow* elements. But the granularity of an activity is not defined, therefore no standard *Action* elements are described. Due to this universal description, AD can also be used to model workflows. However, the main focus is on the visualization for a following implementation of the process [17], [26], [27].

In contrast to AD, the method Business Process Model and Notation (BPMN) is specialized in describing business processes. BPMN is currently available in v2.0.2. It is a widely accepted standard and understandable for all involved parties (drafting, implementation and monitoring) of a business process. The elements of BPMN are very similar to AD. But by using BPMN, the focus is on the modeling of the process for describing the process run through the company and not on a software implementation [17], [10], [19].

With the use of an Event Driven Process Chain (EPC), the focus is placed on occurring events which trigger a function that generates an output event again. Thus, EPCs provide a dynamic view of processes. In EPC only three elements are defined: Functions, Events and Logical connectors. Due to this simple definition of semantic and syntax the models are very flexible, but it also leaves room for interpretation [17], [31], [18], [14].

C. Discussion

In order to model a workflow for a human-robot-collaboration, the method used must satisfy certain requirements. Firstly, it must provide a set of basic elements performable by humans and robots. This ensures that the worker, human being or robot, can also carry out the work instructions. These basic elements must be abstracted to such an extent that they are independent of whether the worker is a robot or a human being. Furthermore, the sequence and duration of the individual working steps must be definable. In order to carry out a single working step, human workers and robots need different specifications. Therefore, it is also necessary to be able to define the attributes of a working step flexibly. If different variants of a product can be represented in one workflow, an on-the-fly variant-change of the current assembly process shall be possible. It is therefore necessary to be able to define decision points in the workflow. Finally, the method has to support collaborative working steps. Therefore, parallel workflows and communication between these workflows must be possible. Now the methods mentioned above are evaluated on the basis of these requirements.

None of the methods fulfill all requirements. *MTM* and *RTM* have been specially developed for the analysis and optimization of human and robot workflows [4], [20]. For this reason, they provide basic elements and the sequence of the processes is determined by a tabular recording. The lead time of a working step is determined by the definition of particular attributes. Further attributes added do not affect the lead time. *AD*, *BPMN* and *EPC* were developed to model software, business or event processes [17]. These modeling languages are characterized by their high flexibility. This means that attributes can be added flexibly, but that they do not provide any basic elements like *MTM* and *RTM*. The different process paths can be defined with the help of decision points. The sequence of the steps is also specified during modeling, but it is not planned to define a lead time.

Subsequently, we merged MTM with RTM [28] and combined it with a universal modeling approach called ADAPT [16]. The universal modeling approach is intended to allow a shift of programming complexity from the end user to a modeling expert, which is only required for the first modeling of a new domain.

III. Approach

Our generic approach focuses on two major parts of modeling discrete processes (such as robot movements): (1) Type of action be accomplished and (2) the required data for each action. Thus we defined a generic model hierarchy enabling custom-specific modeling of actions and assets. This hierarchy illustrated in Fig. 1 features a Meta-Meta-Model defining the basic elements and their relations. The Meta-Model on top enables process- or domain-specific definition of action elements (e.g. using MTM/RTM elements). The final model finally represents a specific workflow using the domain-specific elements.

A. Meta-Meta-Model: ADAPT

As shown in Fig. 1 the Meta-Meta-Model consists of the two major elements called ***Action***, ***Asset*** and three supporting elements, *Decision*, *Relationship* and *Property*.

- *Action* – This element is used to model any kind of task to be accomplished by a machine or a human. Actions can consist of multiple sub-actions. This is shown in

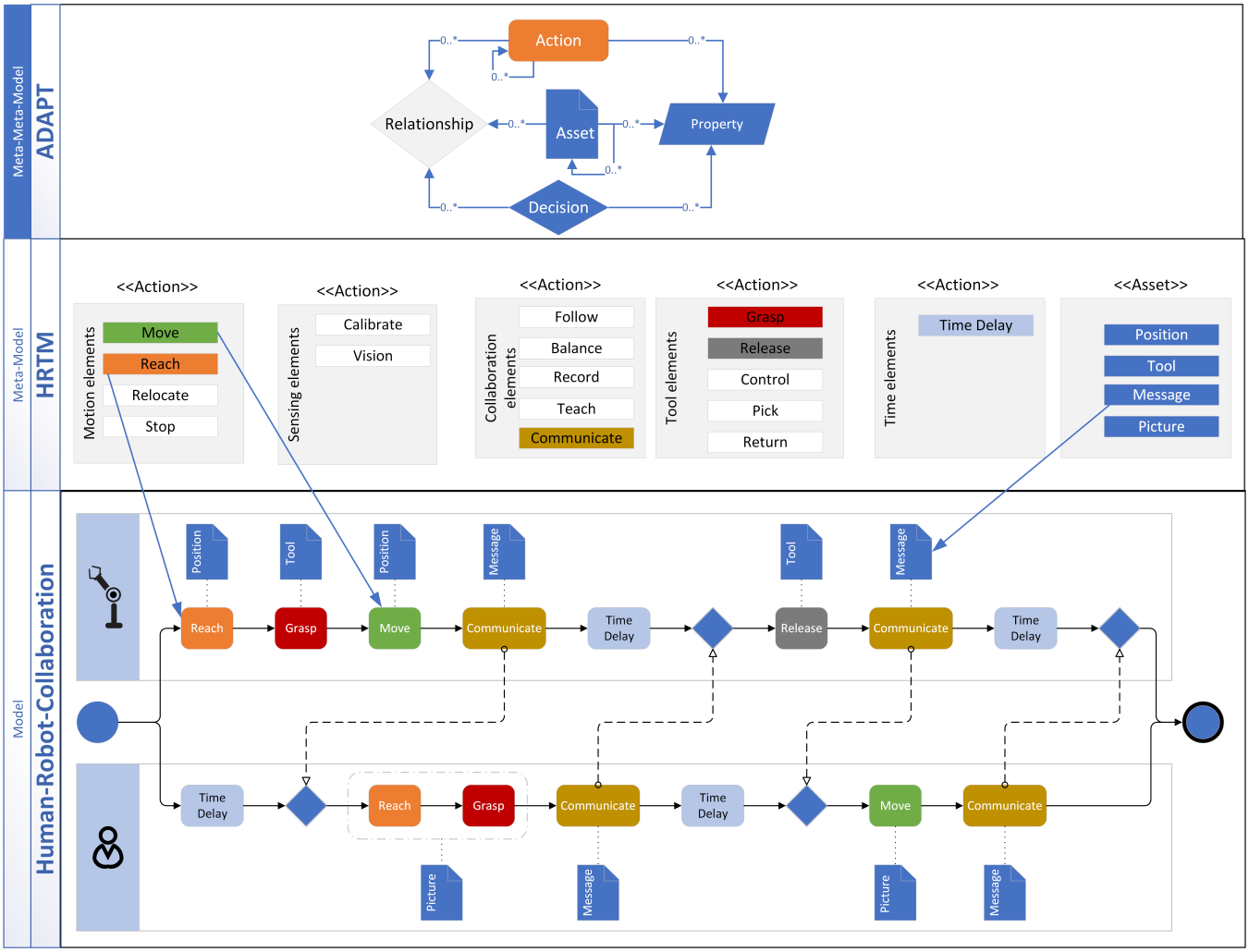


Fig. 1. Model hierarchy (For a better overview relationships, decisions and properties are not shown in full detail)

Fig. 1 with an aggregation of an *Action* to itself. For modeling actions in a generic and reusable manner we use a combination of MTM and RTM methods. An *Action* can be instantiated e.g. as Reach, Grasp, Move, Place, Release, Drill or Screw elements in the Domain Meta-Model defining a set of allowed working tasks. With this Meta-Model it becomes possible to model domain specific models. With corresponding mappings the door is opened to generate platform specific models with implementations for e.g. Reach, Grasp, Move, Place, Release, Drill or Screw (e.g. PLCopen function blocks such as *MC_MoveAbsolute*, *MC_SetPosition*, etc.).

- *Asset* – An *Asset* is a container for any type of information and must include properties. *Assets* can include other *Assets*. This is shown in Fig. 1 with an aggregation of an *Asset* to itself. Typical assets are function blocks containing IEC61131/IEC61499-code (e.g. PLCopen Motion Control), HMI-elements (e.g. buttons, displays, etc.), robot grip positions or work instructions.
- *Decision* – This element models the beginning or fork point of conditional workflows on the basis of offline product configuration data or data gathered from assets at runtime. So it is feasible to model different workflows for various product variants or multiple execution paths. As a result *Decisions* can consume trigger information from any *Asset* (e.g. Camera, Button, Microphone, etc.) for modeling live reactions with various runtime decisions. Depending on this trigger information the *validIf* condition of Relationships described later can be evaluated.
- *Relationship* – Relationship elements can be used to model e.g. aggregation, specialization, predecessor or successor dependencies, can be expanded as desired and consist of a source and a target definition. A relationship element always includes a so called Validity Condition (*validIf*) as presented in [8]. This condition is used to express a kind of variability on the basis of boolean expressions evaluated at design-time or runtime. These boolean expressions are created using boolean variables provided by assets, which are representing logical input

from product configurators or physical input from the modeled process (e.g. Cameras, Sensors or Human-Inputs). If the expression is evaluated true the connected actions are executed or assets are included. This means a conditional action flow on the basis of asset-provided data is possible.
- *Property* – The elements *Decision*, *Asset* and *Action* include *Properties* that describe them in more detail.

Using this core Meta-Meta-Model we now can implement a Meta-Model to be used for modeling collaborative assembly operations.

B. Meta-Model: Human-Robot-Time and Motion

With the previous mentioned ADAPT Meta-Meta-Model the decision-based meta-modeling of collaborative assembly and manufacturing tasks is possible. Therefore, the elements of MTM and RTM are merged and extended by collaborative elements and additional tool elements. In Fig. 1 the 17 defined or instantiated Actions of the so called Human-Robot-Time&Motion approach are shown on the Meta-Model level. In contrast to MTM and RTM method, the HRTM elements do not stipulate a coding or timing assignment. Instead, freely definable Properties can be added to the HRTM elements (i.e. Actions).

C. Modeling elements

Combining MTM/RTM and extending them with collaborative elements results in 17 elements that can be organized in five groups. (1) *Motion elements* represent the movements of a robot or a worker. These include a motion of the arm and a position change, but also the stop of a motion. To describe an action which needs a tool, (2) *Tool elements* are used. In the HRTM approach a tool is defined as an object, which enhance the abilities of the worker or allows the worker to perform a special action. It could be an active tool, like a power screwdriver, or a passive tool, like a stamp. If an interaction with the environment is required, which is not a result of another element, the (3) *Sensing elements* are used. With the help of HRTM collaborative workflows can be modeled. Therefore, (4) *Collaboration elements* are needed for an interaction between two workers. To be able to model a continuous work flow, (5) *Time elements* are used to describe waiting times. The transitions between these action elements are modeled with *Decisions*. In addition, *Actions* as well as *Assets* can be constrained with optional or mandatory relationships, e.g. a Grasp action may also require a Release action.

D. Model: Pick&Place Workflow

For easier modeling of workflows with our HRTM model we use a BPMN-based view model and a corresponding editor. The elements of HRTM are represented with the BPMN element *Activity* and are linked with *Sequence Flow* elements. As defined in BPMN, each workflow starts with a *Start event* and the last element is an *End event*. Additional information can be linked with an *Association* to the inserted HRTM element. Therefore, it is possible to define all necessary data for an action regardless of whether a human worker or a robot executes it.

Action	Group	Description
Reach	Motion	Motion of the hand-arm-system without load
Move	Motion	Move handling of a load
Stop	Motion	Stop of a motion
Relocate	Tool	Repositioning of the worker/robot in space
Control	Tool	Controlling a tool
Grasp	Tool	Indicates that the gripper of a robot or the hand of human worker holds an object
Release	Tool	The gripper of a robot or the hand of human worker is free
Pick	Tool	Pick up a tool from a tool magazine
Return	Tool	Return of a tool into a tool magazine
Calibrate	Sensing	Robot must be calibrated at its current position or a human worker has to orientate himself at the working place
Vision	Sensing	Automated recognizing, identifying, localizing objects
Follow	Collaboration	Mirroring of other movements in general
Balance	Collaboration	Hold the position within the current circumstances
Record	Collaboration	Recording a path, e.g. during teaching (special case of Follow)
Teach	Collaboration	Information exchange from the senior human worker to the novice/robot
Communicate	Collaboration	Information exchange between two or more workers/robots
Time Delay	Time	Wait while the other partner performs an action

TABLE I
HRTM MODELING ELEMENTS REPRESENTED AS ADAPT ACTIONS

Considering the example in chapter IV, the view model of a simple "Pick&Place" workflow is shown in a BPMN-based editor (see Fig. 3). The goal of the example workflow is to place a part on a defined position. Decomposing of a typical human workflow results in a *Reach* element to determine the current position of the part. After reaching the part, it has to be grasped using a *Grasp* element and subsequently placed to the target position with the help of the *Move* element and finally released using a *Release* element. In order to model start and end of a task corresponding *Start-* and *End-events* have to be inserted. Due to the use of sequence flows, the sequencing of the four elements is defined arbitrary.
After modeling the basic sequence of actions the required data to perform these actions has to be assigned by adding source positions, work instruction, etc. Typically human workers are not thinking in HRTM elements, but are automatically combining them to complex tasks. Therefore, our approach supports aggregation of elements in order to create complex tasks and enables a reuse of them. Certain actions such as the *Reach* and *Move* element needs a position and the *Grasp* and *Release* element the corresponding tool. This required information is modeled as Assets and appended to the elements with *Association* lines.

IV. Implementation

This section presents an universal workflow modeling environment WORM implementing the ADAPT-Meta-Meta model. It enables the definition of custom meta-models and supports the specification of custom model elements and their relationships, i.e. custom domain specific languages for graphical modeling. The environment features (1) a tree-based meta-modeling editor, (2) a BPMN-based model/workflow editor and (3) a corresponding runtime-engine, which is able to execute the created workflow model.

A. Meta-Modeling Editor

The meta-modeling editor, called Architect, enables the creation of an ADAPT-based, customer-specific modeling language. In our presented example we created a HRTM model featuring 17 actions and corresponding relationships and assets. Each action now has several relationships such as ”Reach includes Picture”, ”Grasp includes Picture”, etc. On the basis of the rules (i.e. grammar) defined within the Architect, several models can be implemented and verified.

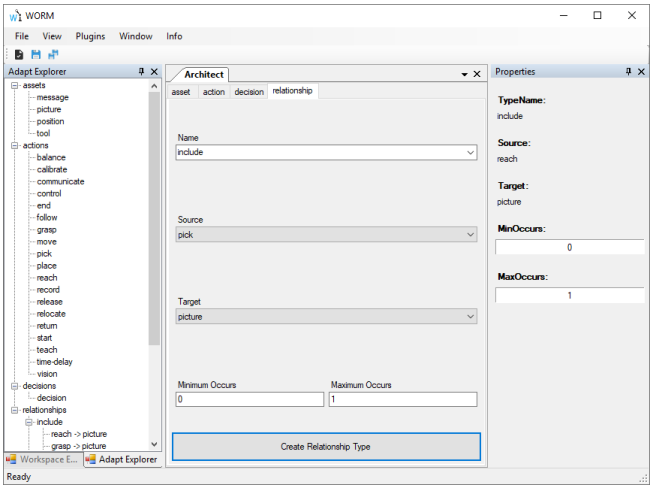


Fig. 2. Creation of the HRTM based Meta-Model.

B. Modeling Editor

In order to create collaborative workflows the user has to model the workflow itself in a sequential way, whereas the user can choose from the HRTM actions modeled in the Meta-Model. In a next step each action may have optional or mandatory relationships to other actions or assets. For example a grasp action requires either a position or in case of an installed vision system a picture of the part to be grasped. The vision system itself may also be linked to the asset picture etc. Continuing this process a complete workflow including all necessary data, e.g. positions, tools, step files, pictures and even worker profiles, is modeled. Thinking of worker profiles the approach enables to assign differently skilled worker or robots to each action. In case of a robot is assigned to an action all assets may be sent directly to the robot by the runtime system (explained below).

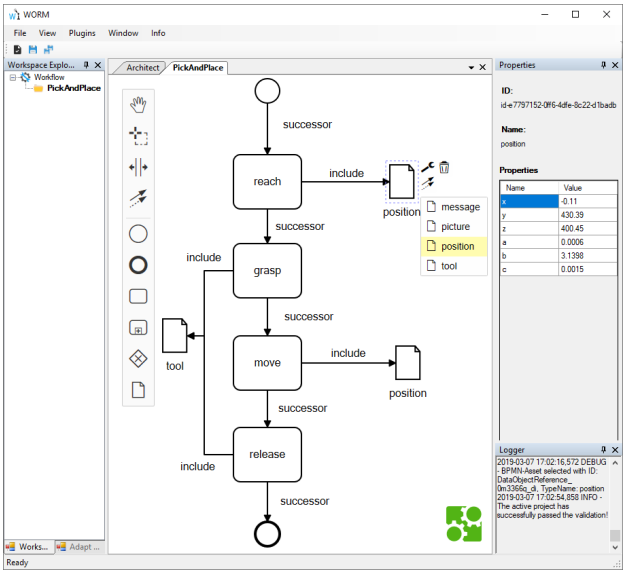


Fig. 3. Creation of a HRTM-based workflow model.

C. Runtime Engine

In order to execute workflows created with our modeling tool a first prototypic runtime engine has been implemented. The workflow, modeled by the Workflow Modeler, can be exported into an XML-representation which is then loaded by the runtime engine. The engine steps through the actions and loads the associated assets from a local asset server (via URL). The assets are then pushed into corresponding MQTT channels [2], to which the devices (i.e. in our showcase the UR10) are subscribed. By this means the robot is supplied with target positions and commands by the runtime engine. As a result one or more robots or any kind of device (such as PLCs with a corresponding listening application) can be programmed using high level workflow modeling. The engine itself is highly customizable in that way which kind of reaction shall be generated during executing the workflow. For our showcase we implemented a MQTT publisher, whereas an OPC UA [22] connector is currently under work.

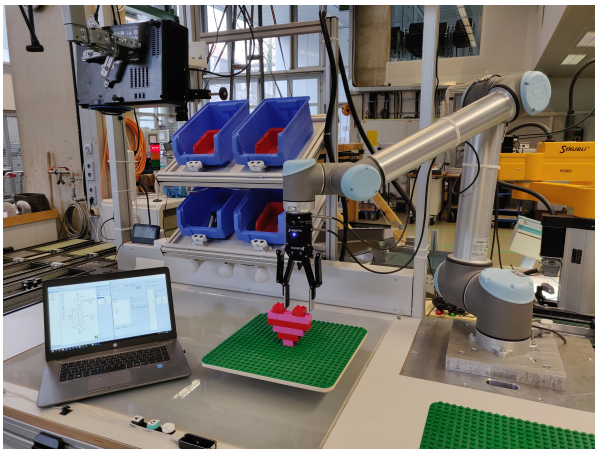


Fig. 4. Demonstration setup.

V. Conclusion and future Work

With the presented modeling approach and its corresponding engineering environment it is possible to model collaborative workflows for humans and robots. As shown in chapter IV, by associating appropriate data to the individual tasks it is possible to create a complete set of data required for programming a robot. For more complex workflows all elements of the HRTM approach as defined in Table I are required. One of the next steps is to define a corresponding minimum set of data (e.g. Assets) for each HRTM Action. This is to ensure that every Action is executable and it is able to perform product specific tasks, depending on constraints such as product dimensions, etc.

When modeling a collaborative workflow, the decision points between the actions must be modeled explicitly in order to support a seamless interaction between humans and robot workers. A human worker automatically switches to the next step as soon as the current step is completed. However, a robot needs specific trigger information to perform the corresponding command at the right time. These triggers must be defined so that the execution of a workflow works correctly on different systems.

For a created HRTM workflow to become reusable, it must be stored in an appropriate way. Currently the model is stored in our ADAPT model hierarchy. In order to enable direct interaction with upcoming toolsets an XML-based transformation into AutomationML may be investigated in future [6]. Furthermore, on the basis of approaches for autonomous assembly program derivation as presented by Thomas [30], this approach may be extended with the possibility of deriving HRTM models out of CAD data and bills of materials.

Finally, with the help of the presented approach an universally readable workflow can be modeled and executed on various devices.

References

[1] W. Antis, J. Honeycutt, and E. Koch, *The basic motions of MTM*. Maynard Foundation, 1973.

[2] A. Banks and R. Gupta, "MQTT version 3.1.1," *OASIS Standard*, 2014. [Online]. Available: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html

[3] K. Beumelburg, "Fähigkeitsorientierte montageablaufplanung in der direkten mensch-roboter-kooperation," Ph.D. dissertation, Universität Stuttgart, 2005.

[4] R. Bokranz and K. Landau, *Produktivitätsmanagement von Arbeitssystemen: MTM-Handbuch*. Schäffer Poeschel Stuttgart., 2006.

[5] L. G. Christiernin, "How to describe interaction with a collaborative robot," in *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction - HRI '17*. Association for Computing Machinery (ACM), 2017.

[6] Common Working Group of AutomationML e.V and eCl@ss e.V., "Whitepaper: AutomationML and eCl@ss Integration," 2017, V 1.0.1.

[7] B. Denkena, H. Wörn, R. Apitz, R. Bischoff, B. Hein, P. Kowalski, D. Mages, and H. Schuler, "Roboterprogrammierung in der fertigung," *Springer*, vol. 9, pp. 656–660, 2005.

[8] D. Dhungana, P. Grünbacher, and R. Rabiser, "The dopler metatool for decision-oriented variability modeling: a multiple case study," *Automated Software Engineering*, vol. 18, no. 1, pp. 77–114, Mar 2011.

[9] P. R. Engelhardt, "System für die rfid-gestützte situationsbasierte produktionssteuerung in der auftragsbezogenen fertigung und montage," Ph.D. dissertation, Technische Universität MÜnchen, 2015.

[10] O. M. Group, *Business Process Model and Notation (BPMN), Version 2.0.2*, Object Management Group Std., Rev. 2.0.2, January 2014.

[11] A. Gupta, D. Fox, B. Curless, and M. Cohen, "DuploTrack: A Real-time System for Authoring and Guiding Duplo Block Assembly," in *Proceedings of the 25th annual ACM symposium on User interface software and technology - UIST '12*. ACM Press, 2012.

[12] S. Hu, X. Zhu, H. Wang, and Y. Koren, "Product variety and manufacturing complexity in assembly systems and supply chains," *CIRP Annals*, vol. 57, no. 1, pp. 45 – 48, 2008.

[13] N. Keddis, G. Kainz, A. Zoitl, and A. Knoll, "Modeling production workflows in a mass customization era," in *2015 IEEE International Conference on Industrial Technology (ICIT)*, March 2015, pp. 1901–1906.

[14] G. Keller, M. Nüttgens, and A. Scheer, *Semantische Prozessmodellierung auf der Grundlage "Ereignisgesteuerter Prozessketten (EPK)"*, ser. Institut für Wirtschaftsinformatik Saarbrücken: Veröffentlichungen des Instituts für Wirtschaftsinformatik. Institut für Wirtschaftsinformatik, 1992.

[15] J. Krüger, T. Lien, and A. Verl, "Cooperation of human and machines in assembly lines," *CIRP Annals - Manufacturing Technology*, vol. 58, no. 2, pp. 628–646, 2009.

[16] R. Lindorfer, R. Froschauer, and G. Schwarz, "Adapt - a decision-model-based approach for modeling collaborative assembly and manufacturing tasks," in *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, July 2018, pp. 559–564.

[17] B. List and B. Korherr, "An evaluation of conceptual business process modelling languages," in *Proceedings of the 2006 ACM Symposium on Applied Computing*, ser. SAC '06. New York, NY, USA: ACM, 2006, pp. 1532–1539.

[18] P. Loos and T. Allweyer, "Object-orientation in business process modeling through applying event driven process chains (epc) in uml," in *Proceedings Second International Enterprise Distributed Object Computing (Cat. No.98EX244)*, Nov 1998, pp. 102–112.

[19] J. Mendling and M. Weidlich, Eds., *Business Process Model and Notation - 4th International Workshop*. Springer, 2012.

[20] S. Y. Nof, *Handbook of Industrial Robotics*, 2nd ed. John Wiley & Sons, Inc., 1999.

[21] J. Novikova, L. Watts, and T. Inamura, "Modeling human-robot collaboration in a simulated environment," in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts - HRI'15 Extended Abstracts*. Association for Computing Machinery (ACM), 2015.

[22] "OPC UA Specification," OPC Foundation, 2015. [Online]. Available: https://opcfoundation.org/developer-tools/specifications-unified-architecture

[23] R. L. Paul and S. Y. Nof, *Human and Robot Task Performance*. Boston, MA: Springer US, 1979, ch. 2, pp. 23–50.

[24] R. P. Paul and S. Y. Nof, "Work methods measurement—a comparison between robot and human task performance," *International Journal of Production Research*, vol. 17, no. 3, pp. 277–303, 1979.

[25] M. Rickert and A. Perzylo, "Industrieroboter für KMU - Flexible und intuitive Prozessbeschreibung," *Industrie 4.0 Management*, pp. 46–49, 2016.

[26] C. Rupp, J. Hahn, S. Queins, M. Jeckle, and B. Zengler, *UML 2 glasklar*. Hanser Fachbuchverlag, 2005.

[27] N. Russell, W. M. P. van der Aalst, A. H. M. ter Hofstede, and P. Wohed, "On the Suitability of UML 2.0 Activity Diagrams for Business Process Modelling," in *Proceedings of the 3rd Asia-Pacific Conference on Conceptual Modelling - Volume 53*, ser. APCCM '06. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2006, pp. 95–104.

[28] D. Schönberger, R. Lindorfer, and R. Froschauer, "Modeling workflows for industrial robots considering human-robot-collaboration," in *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, July 2018, pp. 400–405.

[29] H. Siegert and S. Bocionek, *Robotik: Programmierung intelligenter Roboter: Programmierung intelligenter Roboter*, ser. Springer-Lehrbuch. Springer Berlin Heidelberg, 2013.

[30] U. Thomas, *Automatisierte Programmierung von Robotern für Montageaufgaben*, Prof. Dr.-Ing. Friedrich M. Wahl, Ed. Shaker Verlag, 2008.

[31] W. van der Aalst, "Formalization and verification of event-driven process chains," *Information and Software Technology*, vol. 41, no. 10, pp. 639 – 650, 1999.

A Dynamical System for Governing Continuous, Sequential and Reactive Behaviors

Raphael Deimel

***Abstract*— In interaction with humans or movable objects, robots not only need to react to surprising information quickly, but they also need to synchronize their motions with the world, which can be done by introducing decision points (discrete state transitions), or by continuously adjusting the execution velocity. We present a novel dynamical system based on stable heteroclinic channel networks that can represent static, markovian states as well as continuous transitions between states in a compact and consistent state vector. This so-called phase-state machine can implement regular state machine semantics, but it additionally has the built-in capability to provide and adjust phases and blend consecutive movement primitives for smooth operation. In this paper, we investigate the dynamic properties, present examples for programming specific state machine semantics, and demonstrate the sequencing and mixing of continuous movement primitives.**

I. Introduction

Behaviors involving interaction (e.g. human-robot object handover [14], reactive manipulation strategies [3]) can often be described by directed graphs that sequence simpler behavior generation systems. The most prominent formalism used is the hybrid automaton [15], which is a state machine that activates and deactivates specific controllers and trajectories. For fluid interactions though, we often want behaviors that violate the strict temporal separation between motion generators. We want consecutive motions to blend into each other. We want to preserve estimates by control (e.g. the weight of a tool) across control switches. Or we want to mix motions to communicate uncertainty to an interaction partner. None of these behaviors are impossible to implement using hybrid automata, but require us to fragment and obfuscate the state graph by introducing many additional states and transitions to handle deviating or unclear situations. In addition, to achieve reactivity in real systems it is common for controllers to bypass the state machine abstraction all together by exchanging information with other controllers or perception directly.

We believe that these unwieldy solutions are the symptom of a limitation of the conventional discrete state machine, which cannot represent transitions with nonzero duration between its states. Introducing continuous, time-extended, and nonexclusive (non-Markovian) transitions in between (Markovian) states provides an elegant compromise between providing a notion of time, and independent, separate segments at the same time. The former enables synchronization and blending, while the latter divides complex behaviors in to independent segments that are easier to learn, optimize, and reason with. System evolution becomes continuous at any point in time, while conventional state graphs can be recovered by ignoring transitions.

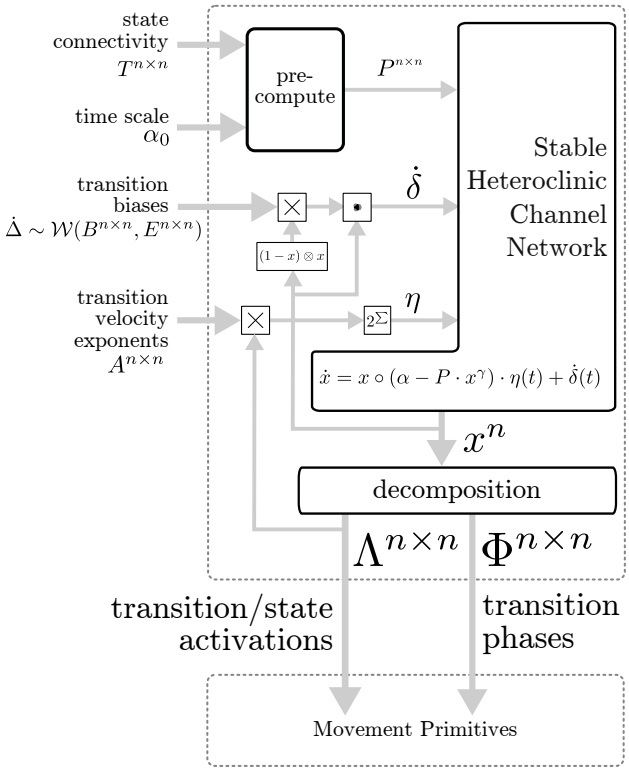


Fig. 1: By decomposing the system state into activation and phase values, dynamical systems with stable heteroclinic channels [4] can orchestrate phase-based movement primitives.

In order to provide continuous, time-extended, nonexclusive transitions, we could extend the hybrid automaton to provide explicit "transition states" in between control states. Though, one would need to add a state for each possible combination of transition and regular state, and for any combinations of transitions sharing the same predecessor state, as those could be active concurrently. In this paper, we investigate an entirely different approach: we implement a discrete state machine using a dynamical system instead of implementing a dynamical system using a discrete state machine. We achieve this by constructing a dynamical system whose attractor consists of a network of "stable heteroclinic channels" (SHC) [11], connected by saddle points which

The author is with the Control Systems Laboratory, Technische Universität Berlin, Germany

We gratefully acknowledge financial support for the project MTI-engAge (16SV7109) by BMBF .

can be interpreted as transitions and states respectively. In previous research, stable heteroclinic channels have been treated as discrete transitions [4], [11], the transitions are continuous in their nature and extended over time though. The SHC formalism provides a straightforward method to construct states (saddle points) and arbitrary transitions (channels) between those states. This paper extends the formalisms to assign an activation value for any possible state and any possible transition as well as to assign a phase variable for each transition. We further will demonstrate how to use those activation and phase values to adjust periods of each transition individually and how to integrate (perceptual) information continuously and context-dependent on the current state.

The activation and phase values can also be used directly to govern phase-parameterized movement primitives such as DMPs [13] or ProMPs [9], which we will demonstrate for a robotic arm.

II. Related Work

The problem domain addressed by the phase-state machine has traditionally been addressed with hybrid dynamical systems [15], and more specifically with hybrid automata. The main difference of phase-state machines w.r.t. hybrid automata is, that states are not necessarily digital, transitions are not discrete, and transitions with a common predecessor can stay in (unstable) superposition. Confusion may arise from the difference in semantics for state and state transition: In a hybrid automaton, trajectories or primitives happen during a *state*, while in the proposed phase-state machine trajectories happen during a *state transition*. Conversely, in hybrid automata synchronization barriers and delays are implemented by *state transitions* (guarded jumps), while in the the phase-state machine, it is implemented by dwelling in *states*.

The work on the phase-state machine builds on prior work that proposed a straightforward method to "program" states and transitions [11], [4]. Their systems have been used as central pattern generators [4], controllers were associated with states and not transitions, though, emulating a hybrid automaton. Phase-state machines are intended to govern a set of motion primitives such as ProMPs. [9], [6], DMPs [2] or others [8] to synthesize actual robot behavior. A recent publication on ProMPs proposes a library of actions and associated transition triggers to concatenate actions [5], albeit it focuses on segmenting and organizing primitives, and not on sequencing and synchronizing their execution.

III. Method

Fig. 1 shows all components of a phase-state machine. The core consists of a Lotka-Volterra-type differential equation, which evolves according to the equation:

$$\dot{x} = x \circ (\alpha - P \cdot x^{\gamma}) \cdot \eta(t) + \dot{\delta}(t) \tag{1}$$

where x is an n-dimensional state vector, vector α and matrix $P^{n \times n}$ are parameters defining the attractor landscape, and $\circ$ denotes element-wise multiplication (Hadamard product). The terms η and $\dot{\delta}$ are control inputs and used to modify the system behavior and will be explained in detail further down. Vector α is called the growth rate parameter, as it influences how fast the value of a state variable grows during a transition. P is a matrix that sets the excitation or inhibition between the state variables so that one saddle point on each coordinate axis is created to represent a discrete state. Further we assume $\dot{\delta} \sim \mathscr{W}(\mu, \sigma)$ to be generated by a Wiener process. The system is therefore numerically integrated using the Euler-Maruyama integration scheme:

$$x(t+\Delta t) = x(t) + E[\dot{x}(t)] \cdot \Delta t + \frac{\sqrt{\Delta t}}{\Delta t} \cdot \mathscr{N}(0, \sigma) \tag{2}$$

a) Inputs to influence system behavior: In order to influence the evolution of the dynamical system, we provide two inputs to the differential equation system: vectors $\dot{\delta}$ and the scalar η. Departing a state (i.e. a saddle point) happens by pushing the system in the direction of the desired subsequent state with $\dot{\delta}$ of Eq. 1. Dwell time is dependent on the magnitude of $\dot{\delta}$. The scalar η adjusts the speed at which the system evolves.

b) Construction of P: The matrix P is constructed using the rules published by Horchler et al. [4] (Eq. 5):

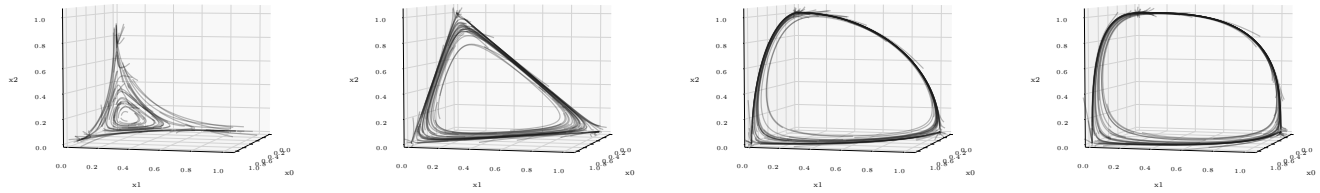
$$P_{ji} = \begin{cases} \alpha_i / \beta_j & \text{if i=j} \\ \frac{\alpha_i - \alpha_j / \nu_j}{\beta_j} & \text{if } T_{ji} = 1 \\ \alpha_i + \alpha_j / \beta_j & \text{otherwise} \end{cases} \tag{3}$$

based on the desired state connectivity matrix T, where $T_{ji} = 1$ indicates an edge from state i to j, otherwise $T_{ji} = 0$. For simplicity, we fix some parameters of the SHC-system: $\alpha_i = \alpha_0$ (uniform growth rates), $\beta_i = 1.0$ (unit state magnitudes), and $\nu_i = 1.0$ (symmetric channels). The scalar α_0 determines how quickly the system evolves, i.e. it parameterizes the time scale. For examples in this paper, $\alpha_0 = 30$.

c) Parameter γ: The γ parameter can be used to modify the shape of the heteroclinic channels. In previous work [4], $\gamma = 1$. Fig. 2 shows the effect of other values. With $\gamma = 2$ channel direction is orthogonal to the current state axis, which simplifies interpretation of input $\dot{\delta}$, which is why $\gamma = 2$ is chosen for all subsequent systems.

A. Extensions to the SHC system

The key difference between the state-machine like behavior of the SHC system and the proposed phase-state-machine is the notion of continuous transition phases. The state vector x sparsely encodes discrete states (one state per dimension), but it also implicitly encodes which transition happens and what the progress (its phase) of the transitions is. In order to specify the behavior of each state and transition independently of each other, we compute two sparse matrices that organize state and transition activations (matrix Λ), and the phases of all transition (matrix Φ). The nonlinear differential equations of the original SHC formulation [4] are further augmented with the signals $A(t)$, $B(t)$ and $E(t)$.



(a) $\gamma=0.5$ (b) $\gamma=1$ (c) $\gamma=2$ (d) $\gamma=3$

Fig. 2: Effect of γ on the attractor shape for a three-state cycle.

a) State and Transition Activation Matrix Λ*:* In order to sparsely encode which transition or state currently is active, we expand the state vector x into a matrix:

$$\Lambda^{\text{transitions}} = \frac{16 \cdot x \otimes x \cdot |x^2|}{(x \otimes 1 + 1 \otimes x)^4 + |x|^4} \circ T \tag{4}$$

The equation is chosen such that $\Lambda_{ji}=1.0$ when the system is close to the plane spanned by states i and j, which is where a channel from saddle point i to j may be located. The outer product $x \otimes x$ forces $\Lambda_{ji}=0$ when x is close to the predecessor's or successor's state axis. Multiplication with T adds information about the *direction* of the transition by assigning the activation to either the upper or the lower triangle of the matrix. State activation is computed as the residual of transition activation, i.e. states are only active if no transition is:

$$\lambda^{\text{states}} = \frac{x^2}{\sum x^2} \cdot \left(1 - \sum \Lambda^{\text{transitions}}\right) \tag{5}$$

As the diagonal of $\Lambda^{\text{transitions}}$ does not have meaningful values, we can conveniently combine all activations into a single matrix Λ:

$$\Lambda_{ji} = \begin{cases} \Lambda_{ji}^{\text{transitions}} & j \neq i \\ \lambda_i^{\text{states}} & j = i \end{cases} \tag{6}$$

State activation is scaled so that $\sum \Lambda = 1.0$. This property allows us to interpret Λ as a set of coefficients for linear combination of motion generators.

b) Transition Phases Matrix Φ*:* The state vector sparsely encodes the discrete states (one state per dimension), but it also implicitly encodes the progress of transitions. Due to the attractor landscape, transitions happen in the plane spanned by the preceding and succeeding state vector. We therefore can compute the progress of a transition from state i to j by simply subtracting the values of the involved state axes:

$$\Phi_{ji} = \frac{|x_j|}{|x_i| + |x_j|} \tag{7}$$

which yields phase variables that grow from 0 to 1 during their associated transitions.

B. Secondary Extensions to the SHC System

Based on Λ and x we can implement behaviors that are not possible with plain SHC networks.

a) Transition velocity adjustment: Modifying transition velocities with α as proposed in [4] is quite limited: we need to ensure that $\alpha_j < \cdot \alpha_i \cdot v_i$ to maintain stability of the heteroclinic from state i to j. This severely limits the available range of variation. We therefore implement a different approach: the growth rate is scaled uniformly by the scalar η in Eq. 1, but only *during* transitions to maintain the stability properties close to the meta-stable saddle points. Λ allows us to specify η for each transition and state independently, using a $n \times n$ matrix A:

$$\eta = 2^{\sum \Lambda \circ A} \tag{8}$$

If $A_{ji}=0$, then transition $i \to j$ will happen with "regular" speed as set by α_0. Positive values speed up exponentially, negative values slow down. The unmodified behavior can be recovered by setting $A=0$

b) Transition Biases: In order to integrate information from perception and higher-level control, we can use $\dot{\delta}$ (Eq. 1) to either push the system away from a meta-stable state, or to stabilize the state. This results in triggering or delaying the start of a transition respectively. In most practical applications though, we will want to specify the bias *towards a specific transition* instead of towards a specific successor state, as in the latter values are dependent on the current state. For example, when in state i, the bias towards itself (the ith component) should be zero to avoid shifting the saddle point along coordinate i, while at the same time we may want it to be nonzero during its predecessor state. In order to achieve independence of the biasing input from the current state, we define a matrix of stochastic biases $\dot{\Delta} \sim \mathscr{W}(B,E)$ which is parameterized by two $n \times n$ matrices that specify the mean value B_{ji} and noise value E_{ji} for each transition $i \to j$ individually. Due to the orthogonality of state vectors we can aggregate using x:

$$\dot{\delta} = \left(\dot{\Delta} \circ ((1-x) \otimes x)\right) \cdot x \tag{9}$$

The mask $(1-x)$ ensures that saddle points are not shifted by $\dot{\Delta}$ accidentally. The behavior as in [4] can be recovered by setting $B=0$ and $E=\varepsilon \otimes 1$.

c) Complete system: When Eq. 1, 8 and 9 are merged, we get the following equations for computing the phase-state machine:

$$E[\dot{x}(t)] = x \circ (\alpha_0 - P \cdot x^\gamma) \cdot 2^{\sum \Lambda(x) \circ A} + (B \circ ((1-x) \otimes x)) \cdot x \tag{10}$$

$$\sigma = (E \circ ((1-x) \otimes x)) \cdot x \tag{11}$$

C. Visualization

Visualizing an n-dimensional continuous state vector intuitively already is challenging, visualizing the evolution of two n^2-dimensional matrices over time even more so. Projection into a two-dimensional subspace does not capture the system's behavior completely, while at the same time even the smallest working examples already are four-dimensional (three state dimensions plus time). In order to achieve an intuitive visualization even with many dimensions, we adapt the concept of UML timing diagrams [12] instead, and adapt it to work with continuous activations and phases. The diagram consists of a number of lines along a time dimension. The input data for a diagram are the elements of the matrices Φ and Λ, referenced by the row and column indices j and $i \in \mathbb{Z}$. For each tuple (j,i) we can compute the function $y_{ji}(t) = i + (j-i) \cdot \Phi_{ji}(t)$, assign a corresponding line width $l_{ji}(t) = l_0 \cdot \Lambda_{ji}(t)$, and draw the resulting lines in a single diagram. In such a diagram, active states (i.e. $i = j$) are visualized as horizontal lines placed at their index value, while active transitions are visualized as continuous curves rising from preceeding state value to the succeeding state's value. The vast majority of states and transitions at any time is inactive and therefore assigned zero line width to make them invisible, avoiding visual clutter. Line width further indicates relative activation when several transitions and states are active at once. Colors are used to aid in distinguishing individual transitions. This visualization method is used throughout the rest of the paper. Important points in time are indicated by gray vertical lines.

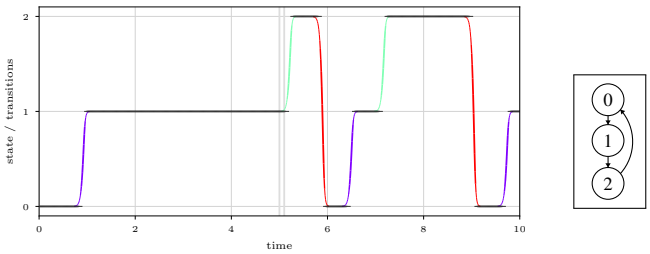
IV. Demonstrations

In this section, we will demonstrate, how various behavior patterns of state machines can be implemented with the proposed system. Some basic patterns (cycling, branching, delays) have already been demonstrated in previous work [4], [11]. We replicate those and additionally demonstrate how to implement common control patterns such as terminal states, error states, exceptions and resets. We will then demonstrate special capabilities intrinsic to the phase-state machine, such as probabilistic decisions, adjustment of transition velocity, and smooth blending of probabilistic movement primitives.

A. Staying in a State and Leaving a State

The system can be stopped from leaving a state by applying a negative $\dot{\delta}$ (Eq. 1) to all successor states. Vice versa, applying a positive $\dot{\delta}$ to a state's successor state will push the system to leave the state. In interaction strategies this is can be used e.g. to synchronize the onset of a motion.

Fig. 3 shows the state and phase evolution of a system with three states and a cyclic network ($0 \to 1$, $1 \to 2$, $2 \to 0$), and $E = 10^{-9}$. First it is locked into state 1 by setting $B_{31} = -10^{-7}$ (and all other biases zero). Then, at 5.0 s B_{31} is set to 0.1 for 0.1 s and to 0 thereafter. The applied pulse causes the system to leave the state almost immediately, which then continues to cycle through the states.



(a) (b)

Fig. 3: Example of stopping in a state, and leaving a state. By adding a negative input bias (first segment) the system stops at this state, adding a non-negative pulse at $t = 5s$ triggers the continuation of the state sequence.

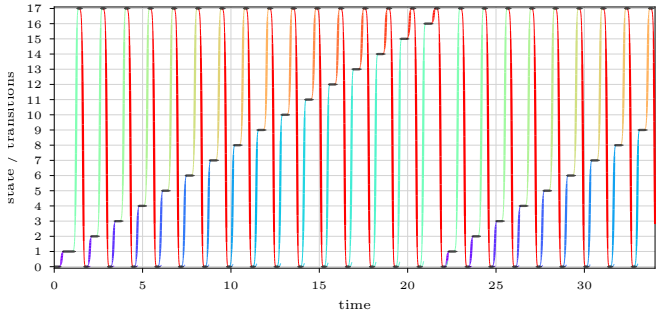


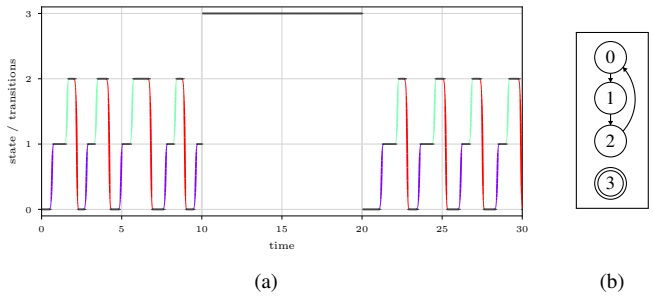
Fig. 4: Example of a system designed to branch from state 0 into states 1 to 16 and to merge back into state 17. The desired branch is selected in each iteration by setting a small $\dot{x}_{\text{bias}}$ vector.

B. Branch Out and Aggregation

Work on systems with stable heteroclinic channels already demonstrated that they can implement arbitrarily complex state machines [1]. In Fig. 4 we replicate this capability with a large number of states. State connectivity matrix T is set up so that from state 0 the state machine branches out into states 1 to 17 (fan-out of 16), which in turn all lead to state 18 (fan-in of 16) before cycling back to state 0. At the start of each cycle, B is set to prefer a different successor state. Figure 4 shows that the dynamical system has no problems implementing large fan-ins and fan-outs.

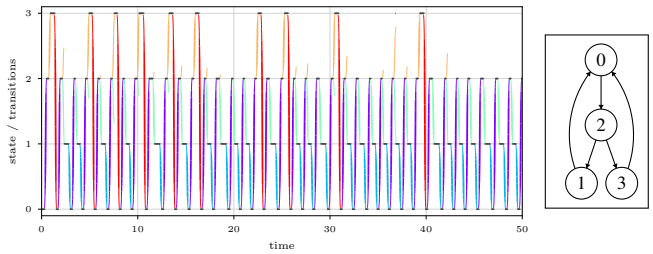
C. Excepting To Error States and Resets

Even though undesired state transitions are repelling, we can still coerce the system to transition to any state at any time by applying a large enough pulse to B. This can be used to implement resets and to except into states not reachable during normal operation (e.g. error states). Fig. 5 demonstrates this ability with a system that has an unreachable error state (state 3). By applying a pulse of $\Delta t = 20$ ms length with total area of 10.0 s to B_{30}, B_{31}, and B_{32}, we transition to this error state at $t = 10.0$ s. Likewise, we can reset the system to state 0 regardless of the state graph by applying another pulse at $t = 20.0$ s without an explicitly programmed state transition.



(a) (b)

Fig. 5: Example of excepting to error state 3 ($t=10.0$), and resetting the system to state 0 ($t=20.0$)



(a) (b)

Fig. 6: Example of implementing probabilistic decisions via bias noise. State 3 is traversed 25% of the cycles.

D. Probabilistic Decisions

An interesting aspect of the phase-state machine is its built-in capability to select transitions probabilistically. The original publications of SHC networks use stochastic noise to destabilize the saddle point equilibria over time to achieve a finite dwell time at each state [4]. But we can also use noise injected by the B input to select transitions probabilistically. Fig. 6 shows a system which branches from state 2 into states 1 and 3. The input matrix B and $E=\varepsilon\otimes 1$ are set to:

$$B=\begin{bmatrix} 0 & 10^{-9} & 0 & 10^{-9} \\ 0 & 0 & 0 & 0 \\ 10^{-4} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{12}$$

$$\varepsilon=\begin{bmatrix} 0 & 3\cdot 10^{-4} & 0 & 10^{-4} \end{bmatrix} \tag{13}$$

The resulting system behavior is shown in Fig. 6. The system visits state 1 on average three times more often than state 3, due to the noise ratio $\varepsilon_1/\varepsilon_3$.

E. Slowing Down and Speeding Up Transitions

A key distinguishing feature of the phase-state machine system w.r.t. previous SHC networks and regular (discrete) state machines are the transitions of non-negligible duration. Transition periods (or phase velocities) can be adjusted for each transition individually via B. Fig. 8 shows the behavior of a system with a three-state cycle. After 3 s with $A=0$ we apply $A_{10}=-4$, $A_{21}=-7$, and $A_{02}=5$. This speeds up transition $2\to 0$ by a factor of 2^5, and slows down $0\to 1$ and $1\to 2$ by factor 2^{-4} and 2^{-7} respectively. At 20 s, 21 s and 22 s, A_{21} is set to -6, -5 and -8 respectively to demonstrate how a transition could be continuously synchronized with

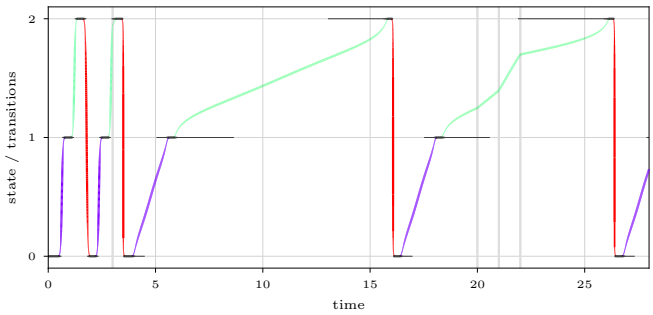


Fig. 7: Timing diagram

Fig. 8: Example of adjusting transition velocities by more than three orders of magnitude ($t=3$s), and adjusting transition velocities during a transition ($t=20$s,21s,22s)

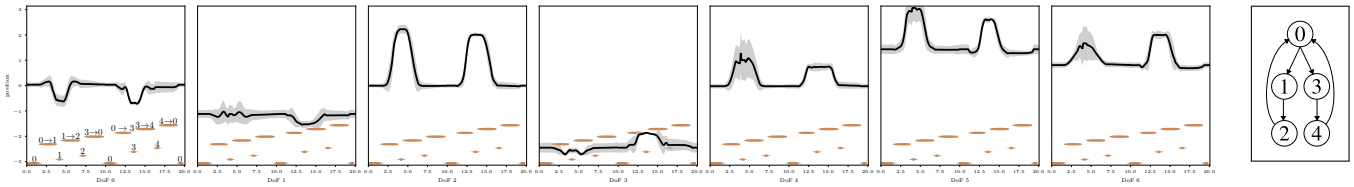
perceptual data. The system maintains stability even when transition durations span more than 3 orders of magnitude.

F. Sequencing ProMPs

Finally, we demonstrate how the phase-state machine can be applied to motion synthesis by combining it with probabilistic movement primitives (ProMP) [10], [7]. The ProMP framework is especially suited as it provides phase-parameterized trajectories, but it also provides a method to mix several concurrently active ProMPs depending on an activation value. Both phases and activations are provided by the phase-state machine. We used a Panda 7-DoF robot arm (Franka Emika) to demonstrate trajectories from an initial pose to two distinct pointing poses (10 examples each), and translated them into two probabilistic movement primitives (ProMPs [10]). For returning to the initial pose, we use the two "pointing" motions but reverse the time-phase relationship. This results in four distinct movement primitives that can be associated with specific transitions and states of a phase-state machine. We use a state graph as shown in Fig. 9b, $\alpha=20$, $E=10^{-8}$, $A=-4$, and relatively large biases to leave states quickly:

$$B=\begin{bmatrix} 0 & 0 & 10^{-3} & 0 & 10^{-3} \\ b_1 & 0 & 0 & 0 & 0 \\ 0 & 10^{-3} & 0 & 0 & 0 \\ b_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^{-3} & 0 \end{bmatrix} \tag{14}$$

In the experiment, $b_1=0$, $b_3=10^{-7}$ in the first five seconds and $b_1=10^{-7}$, $b_3=0$ afterwards to make the system traverse both branches. Transitions $0\to 1$ and $0\to 2$ are associated with one "pointing to" ProMP each. Transitions $1\to 2\to 0$ and $3\to 4\to 0$ are associated with ProMPs to return back to the waiting position (state 0). The return motion is split across two transitions, with the former implementing 90% of the motion and the latter the remaining 10%. This is necessary as the phase-state machine cannot implement bidirectional edges and therefore requires at least three transitions to implement a cycle. When a state is active, then its desired joint pose is determined by averaging the desired poses of all adjacent ProMPs. Fig. 9 shows the resulting behavior of the system, where the robot arm first points left



(a) Desired joint poses. (b)

Fig. 9: 7-DoF robotic arm trajectory generated the phase-state machine using ProMPs

(state 1) and then points right (state 3). Blending between subsequent ProMPs works smoothly, even when they are not matched perfectly together, e.g. when blending from state 0 to transition $0 \rightarrow 3$.

V. Discussion

The phase-state machine formalism proposed in this paper provides behaviors that could – with enough effort – also be implemented with hybrid automata. We do not *need* to switch instantly between control modes though, which makes it easy to blend motions and guarantee smooth execution. We also do not need to decide *when exactly* to switch, as the decision process too is extended over time. Further, interventions such as adaptation of execution speed can be incorporated into the system state at any time, while maintaining a consistent and complete system state, especially w.r.t. to activation and phase values. There are also some drawbacks. Transitions and states have to alternate, so "concatenating" two continuous trajectories is not possible without inserting a somewhat superfluous state in between. Also, two states cannot be mutual successors, i.e. cycling between two states is not possible. While this limitation can be circumvented by inserting a state (as done in the paper's example), it would be preferable to find a system formulation capable of implementing bidirectional edges. Another open problem is the relationship to probabilistic formalisms, specifically optimal control and bayesian inference. The system is capable of representing probabilistic policies, and it can also accumulate uncertain evidence over time. It is unclear how input signals (e.g. elements of B) can be related to conditional probabilities and hence, how Bayes-optimal decision processes can be implemented.

VI. Conclusion

The paper proposed a novel state vector decomposition method for SHC networks [4], that computes a consistent set of transition phases and activation signals. This enables a simple and effective integration with existing motion generation frameworks such as ProMPs [9]. Further, the ability to continuously modify state- and transition-specific parameters simplifies online-adaptation of desired behaviors. We demonstrated how the resulting *phase-state machine* can be used to represent discrete, markovian states as well as provide continuous phases and activation values. We demonstrated how to implement discrete behavior such as branching, error states and resets. We demonstrated the ability to adjust transition duration online and by more than three orders of magnitude, which is not possible with previous SHC networks [4]. We showed how to enact probabilistic decisions and finally, we also demonstrated how the phase-state machine can be used to govern the blending of movement primitives to create continuous, smooth motion.

References

[1] P. Ashwin and C. Postlethwaite, "On designing heteroclinic networks from graphs," *Physica D: Nonlin. Phen.*, vol. 265, pp. 26–39, 2013.

[2] J. Buchli, F. Stulp, E. Theodorou, and S. Schaal, "Learning variable impedance control," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 820–833, 2011.

[3] C. Eppner, S. Höfer, R. Jonschkowski, R. Martín-Martín, A. Sieverling, V. Wall, and O. Brock, "Lessons from the Amazon Picking Challenge: Four Aspects of Building Robotic Systems," in *Proceedings of Robotics: Science and Systems*, 2016.

[4] A. D. Horchler, K. A. Daltorio, H. J. Chiel, and R. D. Quinn, "Designing responsive pattern generators: stable heteroclinic channel cycles for modeling and control," *Bioinspiration & Biomimetics*, vol. 10, no. 2, p. 026001, 2015.

[5] R. Lioutikov, G. Neumann, G. Maeda, and J. Peters, "Learning movement primitive libraries through probabilistic segmentation," *The International Journal of Robotics Research*, vol. 36, no. 8, pp. 879–894, 2017.

[6] G. Maeda, M. Ewerton, G. Neumann, R. Lioutikov, and J. Peters, "Phase estimation for fast action recognition and trajectory generation in human–robot collaboration," *The International Journal of Robotics Research*, p. 0278364917693927, 2017.

[7] G. J. Maeda, G. Neumann, M. Ewerton, R. Lioutikov, O. Kroemer, and J. Peters, "Probabilistic movement primitives for coordination of multiple human–robot collaborative tasks," *Autonomous Robots*, vol. 41, no. 3, pp. 593–612, 2017.

[8] J. R. Medina, F. Duvallet, M. Karnam, and A. Billard, "A human-inspired controller for fluid human-robot handovers," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 2016, pp. 324–331.

[9] A. Paraschos, G. Neumann, and J. Peters, "A probabilistic approach to robot trajectory generation," in *2013 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2013, pp. 477–483.

[10] A. Paraschos, C. Daniel, J. Peters, and G. Neumann, "Using probabilistic movement primitives in robotics," *Autonomous Robots*, pp. 1–23, 2017.

[11] M. I. Rabinovich, R. Huerta, P. Varona, and V. S. Afraimovich, "Transient Cognitive Dynamics, Metastability, and Decision Making," *PLoS Comput Biol*, vol. 4, no. 5, p. e1000072, 2008.

[12] J. Rumbaugh, I. Jacobson, and G. Booch, *Unified Modeling Language Reference Manual, The (2Nd Edition)*. Pearson Higher Education, 2004.

[13] S. Schaal, "Dynamic Movement Primitives -A framework for Motor Control in Humans and Humanoid Robotics," in *Adaptive Motion of Animals and Machines*. Springer, Tokyo, 2006, pp. 261–280.

[14] K. W. Strabala, M. K. Lee, A. D. Dragan, J. L. Forlizzi, S. Srinivasa, M. Cakmak, and V. Micelli, "Towards Seamless Human-Robot Handovers," *Journal of Human-Robot Interaction*, vol. 2, no. 1, pp. 112–132, 2013.

[15] A. van der Schaft, "Modeling of hybrid systems," in *An introduction to hybrid dynamical systems*, ser. Lecture Notes in Control and Information Sciences. Springer, London, 2000, pp. 1–34.

Multilingual Speech Control for ROS-driven Robots

Dominik P. Hofer[1], Simon Brunauer[2] and Hannes Waclawek[3]

***Abstract* — To improve the collaboration of humans and robots, a multilingual speech (MLS) control was created, which allows to manage multiple ROS-based robots at any time.**

Keywords—Speech Control, Multilinguality, Speech to Text, Text to Speech, Intent and Variable Detection

I. Introduction

How can various types of Robots (e.g. robot arm, robot car, etc.) be controlled with the same speech control? In this paper, a universal speech control for various robots, which use ROS as a middleware, is presented. The speech control only uses open source tools and libraries which work offline. This allows the user to secure privacy and independence of companies. Furthermore, to allow people with different mother tongues to use the same speech control, multilingualism was implemented.

II. Method

The workflow for the whole audio dialog was split into five steps: First, language identification (LID), next speech-to-text transformation (STT), afterwards the intent and the variables of the command are detected (I&VD), which is followed by command publishing (CP) and finally the creation of the audio response by text-to-speech (TTS). Figure 1 is a graphical representation of the speech control. For the implementation of this workflow, a set of specialised open-source tools was used, which are described in the following subsections.

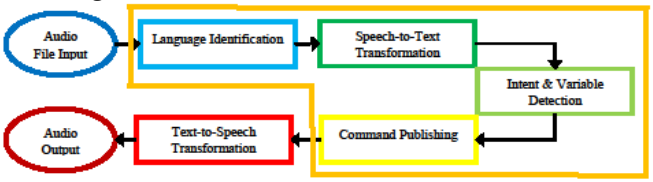


Figure 1: The graphical representation of the processes done by the speech control

A. Language Identification (LID)

For the speech control to transform the audio signal into the correct written language, the LID is necessary. In the case of this project the library iLID, created by Tom Herold and Thomas Werkmeister, is used [1]. The library allows to train a convolutional neural network (CNN) with Mel-Filters on various languages [1].

B. Speech to Text (STT)

The STT-Transformation is done by using the CMU Sphinx library [2]. This open-source offline tool allows to transform any language by letting users create their own dictionaries, language models and acoustic models.

C. Intent and Variable Detection (I&VD)

To let the system know, what the user wants a specific robot to do, the intent, as well as variables, need to be extracted out of the command. ID detects the intent the user has e.g. drive, bring, stop, etc. SF, on the other hand, is the process of finding variables in the command (e.g.: "*fahre 15 Meter*" – 15 meter is the variable). This is achieved by using the Snips NLU tool, which analyses the text via regular expressions or using a logistic regression in combination with Conditional Random Fields (CRFs) [3].

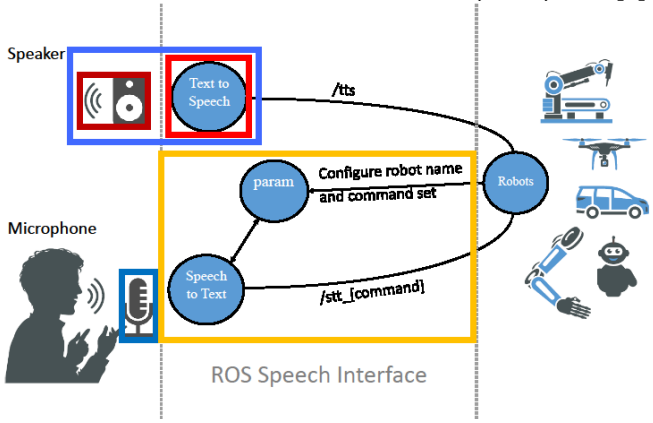


Figure 2: Architecture of the ROS Speech Interface

D. Publishing Command (CP)

For the robot(s), referenced in the audio file, the intent and the stated variables are published on a topic. Two examples of ROS topics can be seen in Table 1. Column one shows the name of the topic. The message format stays the same for all the topics and consists out of three strings. The first string representing the name of the robot. The second referencing the language which was identified by the ID. This was implemented to allow the Text to Speech-Transformation to

ROS Topic	ROS Msg format	Intended Function	Usage Example
/stt_save_pos	stt_std	Save current robot position under a position name	*"Sam, speichere diese Position als Startposition."*
/stt_goto	stt_std	Go to position previously saved via /stt_save_pos	*"Sam, fahre zu Startposition."*

Table 1: Some ROS Topics and the corresponding functionalities

[1] Dominik P. Hofer is with Salzburg Research Forschungs GmbH, 5020 Salzburg, dominik.hofer@salzburgresearch.at and Information Technology & System-Management, Fachhochschule Salzburg GmbH, 5412 Puch bei Hallein, AUSTRIA, dhofer.its-m2018@fh-salzburg.ac.at

[2] Simon Brunauer is with Information Technology & System-Management, Fachhochschule Salzburg GmbH, 5412 Puch bei Hallein, AUSTRIA sbrunauer.its-m2018@fh-salzburg.ac.at

[3] Hannes Waclawek is with Information Technology & System-Management, Fachhochschule Salzburg GmbH, 5412 Puch bei Hallein, AUSTRIA sbrunauer.its-m2018@fh-salzburg.ac.at

respond in the correct language. The final string consists out of the function the robot should execute. The third column of Table 1 shows what function the topic sets in motion. The final column represents sample sentences.

E. Text to Speech (TTS)

The final step, the TTS-Transformation uses the response (further explained in chapter III. Results) of the robot and transforms it into an audio file. This is done by the eSpeak library [4]. This library allows to transform text written in various languages.

III. Results

The architecture of the speech control as shown in Figure 2 consists out of four different ROS nodes: *stt*, *param*, *client* and *tts.*

The *stt* node focuses on the entire process of speech and language analysis, as well as checking if the intended command is supported by the demanded robot. The *param* node, strictly speaking is already integrated in the ROS core node, is a parameter server which holds a list of all the robots currently active in the ROS environment and their supported commands. The parameter server does this process automatically. The *client* node represents the various robots. The final node, *tts*, only focuses on audio output and listens on the tts-topic. The node saves any kind of message in a first in first out (FIFO) concept.

For the implementation of the speech control, a demo scenario was created. This scenario is represented as a dialog flow chart in Figure 3. It shows how a user asks the robot for a certain task. After that the speech control evaluates the command, whether the robot can even execute the task, or if everything is fine. The error analysis is done multiple times during the execution. If the user wants the execution to stop, the stop command can be executed at any given moment. This allows handling of dangerous situations, but it is still recommended to have further security tools at hand.

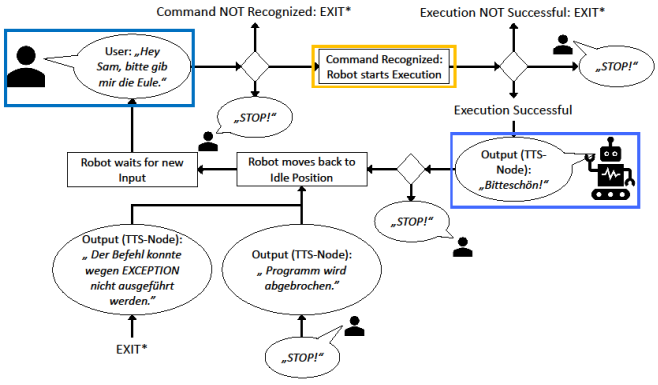


Figure 4: Dialog Flow representation of the Proof-of-Concept

If everything went according to program, the speech control outputs an audio message and the corresponding robot moves back into the idle position. Figure 4 shows an implemented proof-of-concept. Here, the Panda Robot by Franka Emika [5] transports a product (3D-printed owl) from the storage (not in the picture) to the product output facility (conveyor belt).

IV. Further Work

There are some possibilities of improving the speech control. The first and probably most important one would be to allow different accents and dialects. This would provide users an even better experience. This can be achieved by focusing on two points regarding the Speech to Text transformation. The first one would be to allow the use of certain English commands in other languages. This would diminish the need of creating an e.g. German counterpart for e.g. "gripper". The second improvement would be to advance the resources available for CMU Sphinx. This would allow to better use a bigger variety of languages. Another improvement would be understandability. This should mean that it can take several tries till the speech control got the correct command. This problem can also be solved by improving the stt transformation. The last improvement would be speed. The process of analysing the audio input until the robot finally moves currently can take up more than ten seconds. This should be reduced in further research.

V. Conclusio

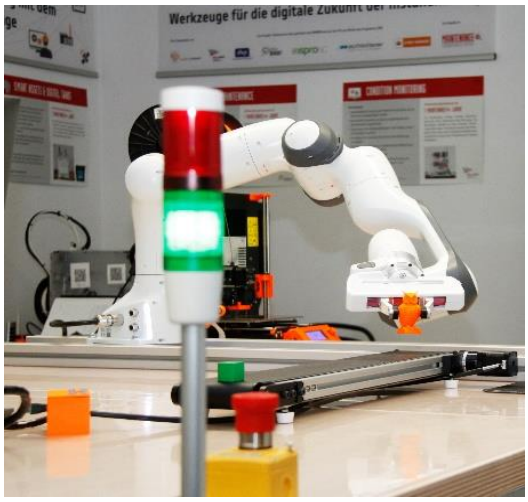


Figure 3: Implementation of the Proof-of-Concept

All in all this short paper presents a speech control that allows to command multiple robots, which have a ROS interface, in multiple languages. The libraries and tools used are on the one hand open-source, which makes the user company independent, and on the other hand do not need an internet connection. This aspect is especially important regarding security issues. However, with the presented modular approach, parts of the workflow can be easily replaced with more sophisticated online speech recognition services. Due to the possibility of intent and variable detection the speech control allows the execution of more programs and a variation of these (e.g. different lengths of driving straight ahead, etc.). It still needs to be pointed out that the clients/robots need to be independently programmed; therefore, the user needs to be able to program the desired robot. In sum, the proof-of-concept works and there are still options left to improve it.

References

[1] T. Herold and T. Werkmeister, "Practical Applications of Multimedia Retrieval," 7 April 2016. [Online]. Available: https://github.com/twerkmeister/iLID/blob/master/Deep%20Audio%20Paper%20Thomas%20Werkmeister%2C%20Tom%20Herold.pdf.. [Accessed 2019 March 9].

[2] CMU Sphinx, "Open Source Speech Recognition Toolkit," 7 Juni 2017. [Online]. Available: https://cmusphinx.github.io. [Accessed 2019 March 9].

[3] A. Coucke, A. Saade, A. Ball, T. Bluche, A. Caulier, D. Leroy, C. Doumouro, T. Gisselbrecht, F. Caltagirone, T. Lavril, M. Primet and J. Dureau, "Snips Voice Platform: an embedded Spoken Language Understanding system for private-by-design voice interfaces," 6 Dezember 2018. [Online]. Available: https://arxiv.org/abs/1805.10190. [Accessed 2019 March 9].

[4] eSpeak, "eSpeak text to speech," 1995. [Online]. Available: http://espeak.sourceforge.net. [Accessed 2019 March 9].

[5] Franka Emika, "Panda," 2018. [Online]. Available: https://www.franka.de/panda. [Accessed 2019 March 9].

Object Grasping in Non-metric Space Using Decoupled Direct Visual Servoing

Bernhard Neuberger[1], Geraldo Silveira[2], Marko Postolov[3] and Markus Vincze[4]

***Abstract*— In this paper we present a robotic system for grasping novel objects. Using a low-cost camera mounted on the end-effector, our system utilizes visual servoing control to command the gripper to a grasp position that is prescribed during a teach-in phase when the object is presented to the system. By using decoupled direct visual servoing, an intensity-based approach, object grasping is done without any 3D input and requires no metric information about the object. Although the robot moves in the 3D Euclidean space and is controlled in the joint space, the command signal is derived completely from pixel information from the input image in the 2D projective space. Furthermore, the control strategy is extended for trajectory following in the control error space to generate smoother and more stable trajectories. This enables more direct and accurate positioning of the end-effector. A set of experiments is performed with a 7 DoF KUKA LWR IV robotic arm and shows the capability of precisely grasping objects from cluttered scenes. The system also shows robustness to object movement during the grasping process as well as robustness against errors in the camera calibration.**

I. INTRODUCTION

Robotic arms are widespread in industrial environments for production tasks and are capable of precise sub millimeter positioning. In order to exploit such high precision the system is required to perceive the environment with similar accuracy.

One common task in robotics is grasping, which works well when considering controlled conditions but becomes increasingly more challenging when its necessary to adapted to an changing environment.

The focus in this work is on the task of grasping objects with a robotic arm. The robotic system is equipped with a low cost camera that provides the perception. The described approach is capable of grasping objects that are newly presented to the robot and only require a teach-in phase to generate a single reference image.

State-of-the-art methods [5], [13], [9] have shown that visual servoing control can stabilize a robotic system around an equilibrium space. Here the work from Silveira et. al. [15] is extended such that the visual servoing approach is used for the task of grasping. The considered objects have a planar and textured surface that will be visible to the robot during the task of grasping.

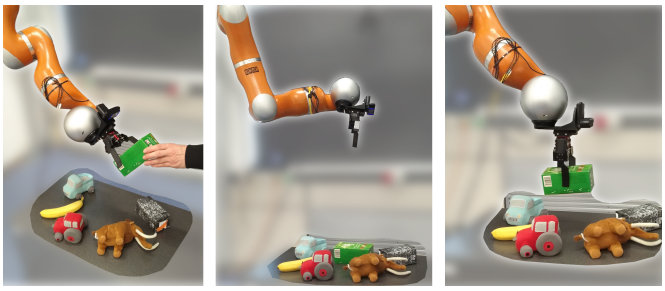


Fig. 1: Teaching the robot to grasp a newly presented object.

In order to use visual servoing for grasping the task was broken down to an end-effector positioning. Therefore first experiments were conducted were it is shown that the system is able to position itself in regards to a reference image. This approach is extended in such a way that the robot is guided along a desired trajectory to the final pose. For this guided movement a desired control error is predetermined so that the initial control error is gradually reduced. This enables a direct smooth movement and is used to grasp objects in a cluttered environment without colliding with the surroundings of the target object. For first results on grasping the clutter was placed such that it was guaranteed that the target object was good visible and the desired grasp was reachable.

In the teach-in phase a new object is learned by showing it to the robot. Therefore it is needed to present the object to the camera of the system as it would be seen when grasped by the gripper. We do not require any form of object model or depth information and purely rely on the input from the mounted camera. Figure 1 shows the teach-in phase, the system state when starting the grasping process and the state of the robot after grasping the newly learned object.

Additionally, a number of experiments was conducted which test the system performance in regard to errors in the camera parameters. For this case wrong camera intrinsic parameters, particularly the focal lengths f_x and f_y in the algorithm were disturbed when a positioning task was performed.

The presented system builds on the work from [15] and contributes with the following additional features:

- visual servoing is used as a tool for grasping newly presented objects

*This work was not supported by any organization

[1]Bernhard Neuberger is with Faculty of Electrical Engineering, ACIN, V4R, Technische Universität Wien, 1040 Wien, Austria neuberger@acin.tuwien.ac.at

[2]Geraldo Silveira is with DRVC Division, CTI Renato Archer, CEP 13069-901, Campinas/SP, Brazil Geraldo.Silveira@cti.gov.br

[3]Marko Postolov is with Chair of Automatic Control Engineering, Technische Universität München, 80333, München, Germany marko.postolov@tum.de

[4]Markus Vincze is with Faculty of Electrical Engineering, ACIN, V4R, Technische Universität Wien, 1040 Wien, Austria vincze@acin.tuwien.ac.at

- an easy to use teach-in phase is used that requires a human supervisor to take a single reference image of the object in the desired grasp pose
- a desired error trajectory is presented for smoother movement that enables grasping in clutter
- we show that the approach has high robustness against errors in the camera intrinsic parameters
- object grasping remains successful even when the target is repositioned during the process

II. RELATED WORK

Saxena et. al. [11] present a robotic system that is able to grasp newly seen objects. Their approach calculates corresponding points for grasping the object and then calculates a 3D grasp point from a set of sparse points on the object. In contrary to this work we are not able to grasp completely unseen objects but we need to teach the object to the robot which doesn't require any 3D object information.

Fischinger et. al. [6] present a grasping approach that uses depth data to extract features for grasping objects in piles and cluttered scenes. They present a method that is able to grasp newly presented objects without any object knowledge.

In [7] Levine et. al. show a deep learning approach that learns to move the robot in the task space such that it results in a high probability for a successful grasp. They show that continuous servoing corrects the mistakes from the network and improve the grasp quality. Their method requires a large number of training data compared to a single reference image in our method.

Another deep learning based method for grasping is presented by Mahler et. al. in [8]. They show that their network is able to predict grasp points with a high success rate when trained on a large synthetic dataset. They also present a grasp planner that is needed to position a robot within workspace constraints. In contrary our method directly positions in regards to the target and does not use grasp points at all.

Chaumette and Hutchinson present in [3] and [4] an overview of various state-of-the-art visual servoing approaches. They present control strategies for image-based visual servo control (IBVS) and position-based visual servo control (PBVS). The difference between IBVS and PBVS are examined and stability of the strategies are investigated. PBVS methods such as [17] and [16] have the advantage of having full control over the trajectories in the Cartesian space but have the disadvantage of being sensitive to camera parameters. Our approach is similar to IBVS methods as presented in [2] but compared to them we don't use image features. Instead pixel intensities are used directly as presented in [14] and use it for the specific robotic task of grasping.

In [9] Mariottini et. al. show how to use IBVS to control a nonholonomic mobile robot to a desired pose. Similar to our approach they don't use any metric information and control the robot in the epipolar geometry. Contrary to our method they use extracted features from the images for the IBVS. In our system the visual tracker from [10] is used.

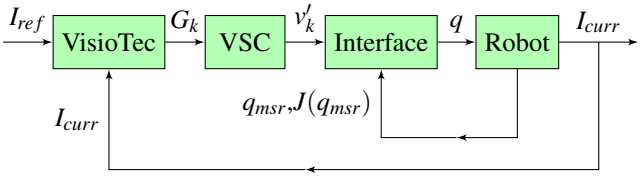


Fig. 2: Conceptional Overview of the Visual Control

Nogueira presents in [10] an intensity-based homography estimation which is implemented in the robot operating system(ROS). In this paper we will refere to this software as the "VisioTec" that is publicly available[1]. It is used to calculate the homography between the reference image and the currently captured camera image.

III. VISUALLY GUIDED OBJECT GRASPING

For our grasping approach, the system perceives the environment with a camera mounted on the end-effector of the robotic arm. In the teach-in phase, the object is presented to the robot in the final grasp pose. This provides the system with an reference image in the precise configuration for the final grasping pose. After this phase, the robot arm automatically moves from any initial pose to a grasp pose such that the live image from the camera coincides with the recorded reference image. This enables the system to grasp the object (by closing the fingers of the end-effector) even if the object is moved at any time during the control procedure.

Our approach enables a set of objects to be grasped without the need for object models. As such, novel objects can be simply presented to the system for immediate grasping. Human involvement is reduced to the task of positioning an object in front of the wrist mounted camera and recording of the reference image.

This section gives an overview of the visual control and details for the steps of the robot control strategy. An extended control strategy for trajectory following in the control error space is then presented. This extension guides the end-effector motion along a coarse trajectory for smooth positioning.

A. Concept Overview

Figure 2 shows a block diagram of the conceptional overview of the robot control loop that is used for the grasping task. The first part in the control loop is the VisioTec visual tracker [10] that takes a selected bounding box from the reference image I_{ref} and performs intensity-based image registration for the tracked area in the current image I_{curr}. The output of the VisioTec tracker is the homography G between the reference target area and the corresponding area in the current image.

This homography is the input to the visual servo controller (VSC), which is described in III-B. This component takes the homography and outputs the desired end-effector velocity commands v'_k.

[1] http://wiki.ros.org/vtec_ros

Our implementation includes an interface between the VSC and the robot control due to the lack of a velocity control mode. The interface transforms the commanded velocity v_k' directly into desired joint states q.

The computed joint states are received by the robot control unit to move the robotic arm to a desired configuration. Additionally, the robot control unit returns the measured joint states q_{msr} and also the current Jacobian matrix $J(q_{msr})$. Within this block $J(q_{msr})$ is used to calculate the pseudo inverse Jacobian matrix J^+ to calculate the joint velocities according to

$$\dot{q} = J^+ v_k'. \tag{1}$$

The measured joint states q_{msr} and joint velocities $\dot{q}$ are used to determine the commanded joints q using the sampling time t_Δ

$$q = q_{msr} + \dot{q} t_\Delta. \tag{2}$$

The robot arm moves according to the commanded joint states and captures a different view of the environment. The latest captured image is returned to the VisioTec tracker and closes the control loop.

B. Robot Control

The purpose of the robot control is to transform the homography G_k from the VisioTec to the end-effector velocity v_k' to move the robot arm closer to the reference pose.

The implementation of the robot control requires the template location in the location matrix G_l to be set in terms of pixel coordinates (l_x, l_y). So long as the target is located in the selected area the control error ε_k will be zero. The location matrix G_l, the camera intrinsic parameters $K \in \mathbb{R}^{3\times3}$ and the hand-eye calibration $T' \in \mathbb{SE}(3)$ are set according to

$$G_l = \begin{bmatrix} 1 & 0 & l_x \\ 0 & 1 & l_y \\ 0 & 0 & 1 \end{bmatrix}, K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, T' = \begin{bmatrix} R' & t' \\ 0 & 1 \end{bmatrix} \tag{3}$$

where f_x and f_y are the focal lengths of the camera and (c_x, c_y) is the principal point coordinate. The homography G_k from the VisioTec tracker is than transformed to the image frame with

$$G_k' = G_k G_l^{-1}. \tag{4}$$

The control point $p^* \in \mathbb{P}^2$ and the control vector $c^{*\prime} \in \mathbb{R}^3$ are used to calculate the control error

$$\varepsilon_k = \begin{bmatrix} 2I & [m^{*\prime}]_\times \\ -[c^{*\prime}]_\times & I \end{bmatrix} \begin{bmatrix} (H-I)m^{*\prime} \\ \vartheta\mu \end{bmatrix} \tag{5}$$

where

$$H = K^{-1} G_k' K, \quad m^{*\prime} = K^{-1} p^* \tag{6}$$

and

$$r = \frac{1}{2}\mathrm{vex}(H - H^\top) \tag{7}$$

$$\vartheta = \begin{cases} \arcsin(||r||) & \text{if} \quad tr(H) \geq 1, \\ \pi - \arcsin(||r||) & \text{otherwise}, \end{cases} \tag{8}$$

$$\mu = \frac{r}{||r||}. \tag{9}$$

The error from the control law is used to calculate the required camera velocity in order to reduce the control error

$$v_k = \lambda(\varepsilon_k)\varepsilon_k. \tag{10}$$

Here $\lambda(\varepsilon_k)$ is a variable gain that was used in our setting. This ensures that the gain declines with an increasing control error and will result in small end-effector velocities with high control error. This reduces velocities exponentially for very high control errors and keeps them within boundaries.

$$\lambda(\varepsilon_k) = \alpha \mathrm{e}^{-\gamma||\varepsilon_k||} \tag{11}$$

The control parameters $\alpha > 0$ and $\gamma > 0$ can be tuned such that a higher α increases the velocity and a higher γ increases the damping of the gain.

Finally, the velocity from the camera frame is transformed to the tool center point (TCP) frame with

$$v_k' = \begin{bmatrix} R' & [t']_\times R' \\ 0 & R' \end{bmatrix} v_k \tag{12}$$

where $[t']_\times$ is the skew symmetric matrix of the translation vector between the camera and TCP frame and R' is the rotation matrix for the transform.

C. Trajectory Following

The control strategy as described above moves the robot to a reference pose. For a more reliable grasping, the system can be adapted for smoother and more stable end-effector movement. This first requires the introduction of a desired control error trajectory $\varepsilon^*(t)$. The actual control error is derived from the error trajectory $\varepsilon^*(t)$ according to

$$\varepsilon'(t) = \varepsilon(t) - \varepsilon^*(t) \tag{10}$$

Stable trajectories are then achieved by adapting the control law from Equation (13) to

$$v_k = \lambda(\varepsilon')\varepsilon'(t) + \frac{\partial \varepsilon^*(t)}{\partial t} \tag{14}$$

Specific details of the desired control error is presented in Section V.

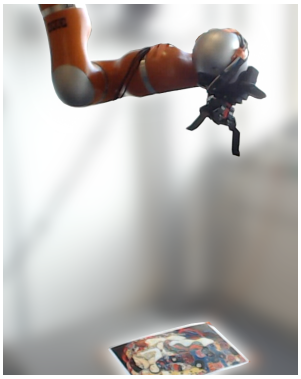
IV. EXPERIMENTAL SETUP

In our experiments, we use a KUKA LWR IV [1] robotic arm with the provided control unit. The robot arm has 7 degrees of freedom (DoF) and is controlled with position commands for the joints. A Logitech HD C920 webcam and a dynamixel AX-12A Dual Gripper are mounted on the end-effector of the arm with a 3D printed support structure.

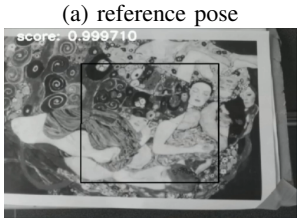
The VisioTec, the VSC and the interface depicted in the block diagram of Figure 2 run on a remote PC. The remote PC and the KUKA control unit communicate via Ethernet. Communication between the remote PC and the robot control unit is enabled with the kuka-lwr-ros package[2], which uses the fast research interface[3] (FRI) [12].

[2]https://github.com/epfl-lasa/kuka-lwr-ros
[3]https://cs.stanford.edu/people/tkr/fri/html/

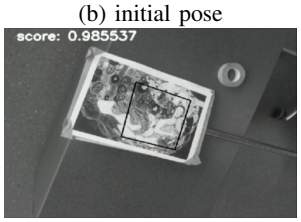
(a) reference pose



(b) initial pose



(c) reference image



(d) initial image

Fig. 3: Positioning Experiment

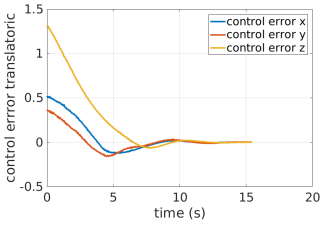
V. RESULTS

A number of different experiments were executed to evaluate the performance of our approach. First, we perform positioning experiments to establish a base line to evaluate the benefit of the trajectory following adaptation. In these experiments, the robot arm is tasked to position itself with respect to a known planar textured surface. Second, we perform grasping performance to showcase the capability of the system to grasp novel objects in clutter. The grasping task is restricted to rigid textured objects with a planar surface, where for each experiments the object is presented to the system in its final grasping pose. Finally, we conduct experiments to test the robustness under camera parameter errors.

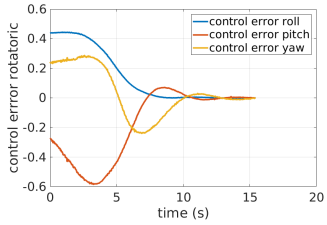
A. Positioning Experiment

For this experiments, a plain textured image is used as target. After a reference image is recorded from a reference pose the robot arm is moved to an initial pose. The goal is to control the robot arm in such a way that it moves back to the reference pose while reducing the control error. Figure 3 shows the setup of the positioning experiment with the robot arm in the reference pose and initial pose. The reference image with the selected target area and the initial image with the tracked target area are also shown.

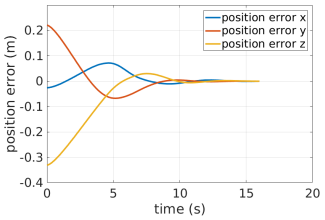
For the evaluation, the robot state is recorded along the whole trajectory from the initial pose to the final pose. Figure 4 plots the pose and control errors over the duration of the experiment. We can see that the errors reduce and eventually go to zero after 15 seconds. Although the final pose is reached, the movement shows unintended behaviour as seen by the significant overshoot in Figure 4c and Figure 4d.



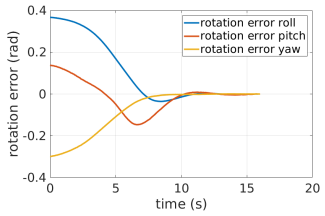
(a) control error translatoric



(b) control error rotatoric

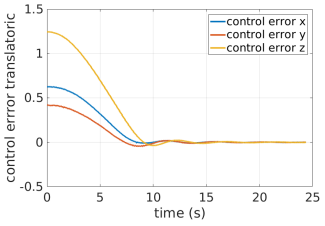


(c) position error translatoric

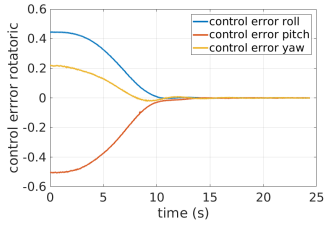


(d) position error rotatoric

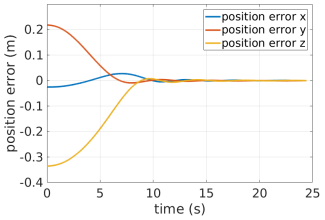
Fig. 4: Results from the positioning experiment



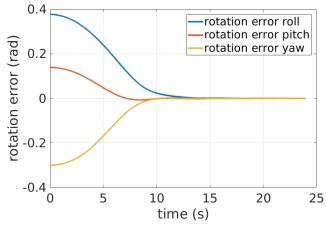
(a) control error translatoric



(b) control error rotatoric



(c) position error translatoric



(d) position error rotatoric

Fig. 5: Results from the trajectory following experiment

B. Trajectory Following

The desired control error is set in such a way that the initial error is gradually reduced along a smooth function with the property of being continuously differentiable. A Lipschitz continuous desired control error is made possible by setting

$$\varepsilon^* = \begin{cases} \varepsilon_0(1+2(\frac{t}{T})^3 - 3(\frac{t}{T})^2) & t \leq T \\ 0 & t > T \end{cases} \quad (15)$$

where ε_0 is the initial control error from the initial end-effector pose. This choice of control error guarantees a Lipschitz continous commanded velocity.

Figure 5 shows the pose and control errors with trajectory following activated. Compared to Figure 4, we see that the control error is now reduced in a more controlled way, which results in a more direct and smooth movement. It can be seen in Figure 5c that the pose error is gradually reduced with less overshoot along the x-axis compared to Figure 4c.

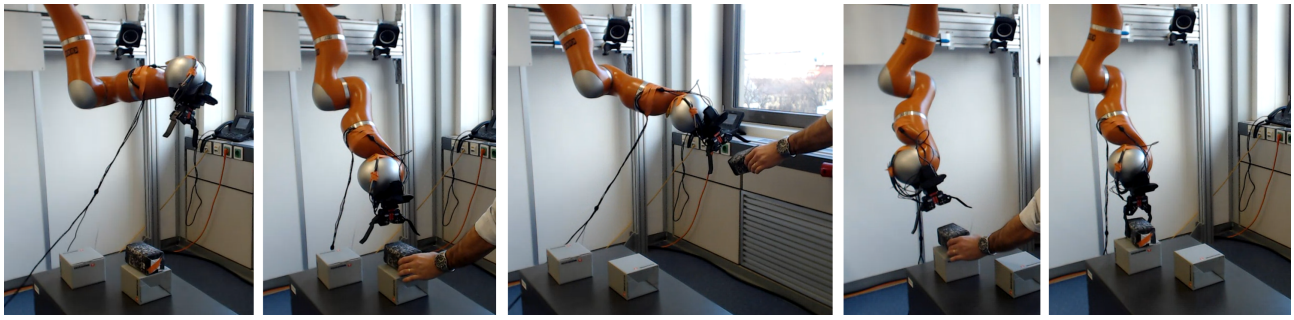
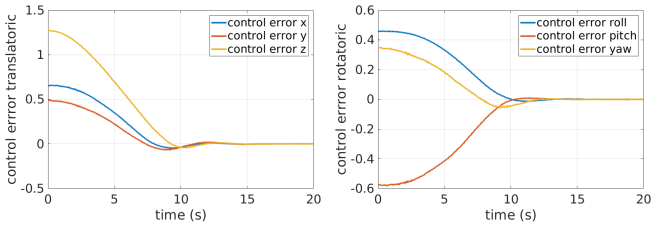
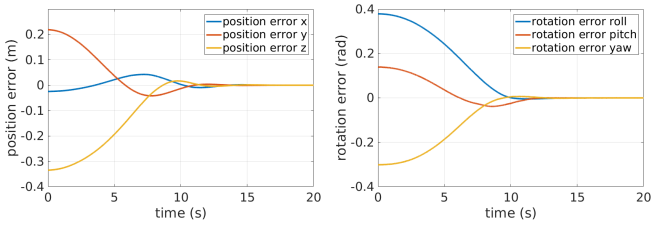


Fig. 6: Grasping an object while repositioning the target object

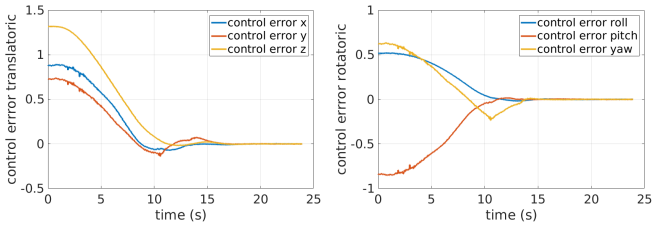


(a) control error translatoric (b) control error rotatoric

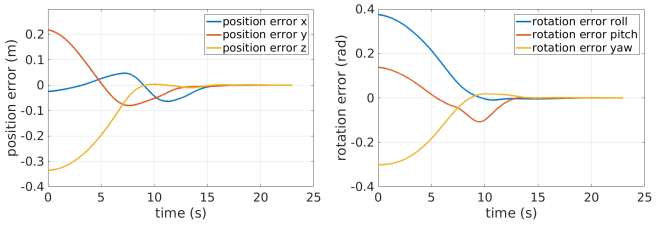


(c) position error translatoric (d) position error rotatoric

Fig. 7: Results from the trajectory following with 37.40% errors in camera parameters



(a) control error translatoric (b) control error rotatoric



(c) position error translatoric (d) position error rotatoric

Fig. 8: Results from the trajectory following with 113.74% errors in camera parameters

C. Object Grasping

The setting from the trajectory following experiments were used to grasp objects. Qualitative results are presented in Figure 6. This shows not only a successful grasp but also the fact that the system successfully compensates for the movement of the target during the process of positioning to the final pose. So long as the target object remains in the camera's field of view the robot is able to successfully follow the target. Grasping is achieved by closing the fingers of the gripper when the pose error is below a predifined threshold.

These results show that our visual servoing approach can be used to grasp objects in unpredictable conditions. Further experiments show that the inclusion of trajectory following enables grasping in cluttered scenes, where collisions are possible near the target object, i.e. obstacles sitting on the same plane as the target object. As shown in Figure 1, grasping is successful as the gripper moves directly towards the object without colliding with surrounding objects.

D. Robustness against errors in the Camera Parameters

The positioning experiment is repeated with an added error in the focal lengths f_x and f_y if the camera intrinsic matrix K.

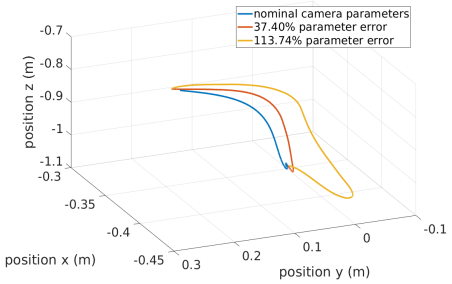


Fig. 9: End-effector trajectory for different camera parameters

In order to obtain the nominal parameters K, a checkerboard and the ROS camera calibration package[4] is used.

Figure 7 shows the pose and control error over time with an error of 37.40% in the focal length of the camera parameters and Figure 8 shows the same plots with 113.74% of error. The results show that even with 113.74% error the robot still manages to reach the reference pose. Further increase in the error of the camera parameters results in a

[4]http://wiki.ros.org/camera_calibration

failed experiments because the target shifts out of the field of view. In this case the reference image can longer be tracked.

Although the reference pose is still reached with a very high error the resulting end-effector trajectory is not feasible for grasping experiments. Figure 9 shows the trajectories of the end-effector from the initial pose to the the reference pose with different settings of the camera parameters. We can see that the trajectories with the nominal parameters and 37.40% error are similar and follow a very direct path. But it is visible that the trajectory with 113.74% error of camera parameters follows a more complicated path that would result in a collision with surrounding objects.

VI. CONCLUSION

The results show that a reliable grasp is possible with our method even if we add errors to the camera intrinsic parameters. We show that newly taught objects can be tracked and grasped with the system.

Future work will exploit the redundancy of the KUKA LWR IV. The 7 DoF of the robot arm has one additional degree of freedom compared to the workspace of the robot. This can be used to avoid singularities, joint limits or keeping distance between the joints and obstacles.

Further plans will improve the system in such a way that the robot can detect the target in a newly presented image even if the target is completely lost in between. This can be beneficial for a mobile platform which can exploit the room and than plan the object manipulation accordingly. This would allow a robot to grasp or manipulate previously learned objects in an novel environment.

REFERENCES

[1] R. Bischoff, J. Kurth, G. Schreiber, R. Koeppe, A. Albu-Schäffer, A. Beyer, O. Eiberger, S. Haddadin, A. Stemmer, G. Grunwald, *et al.*, "The kuka-dlr lightweight robot arm-a new reference platform for robotics research and manufacturing," in *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)*. VDE, 2010, pp. 1–8.

[2] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," in *The confluence of vision and control*. Springer, 1998, pp. 66–78.

[3] F. Chaumette and S. Hutchinson, "Visual servo control. i. basic approaches," *IEEE Robotics & Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.

[4] ——, "Visual servo control. ii. advanced approaches [tutorial]," *IEEE Robotics & Automation Magazine*, vol. 14, no. 1, pp. 109–118, 2007.

[5] P. I. Corke and S. A. Hutchinson, "A new partitioned approach to image-based visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 507–515, 2001.

[6] D. Fischinger, M. Vincze, and Y. Jiang, "Learning grasps for unknown objects in cluttered scenes," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 609–616.

[7] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421–436, 2018.

[8] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," *arXiv preprint arXiv:1703.09312*, 2017.

[9] G. L. Mariottini, G. Oriolo, and D. Prattichizzo, "Image-based visual servoing for nonholonomic mobile robots using epipolar geometry," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 87–100, 2007.

[10] L. Nogueira, E. de Paiva, and G. Silveira, "VISIOTEC intensity-based homography optimization software: Basic theory and use cases," CTI, Brazil, Tech. Rep. CTI-VTEC-TR-01-2017, 2017.

[11] A. Saxena, J. Driemeyer, and A. Y. Ng, "Robotic grasping of novel objects using vision," *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 157–173, 2008.

[12] G. Schreiber, A. Stemmer, and R. Bischoff, "The fast research interface for the kuka lightweight robot," in *IEEE Workshop on Innovative Robot Control Architectures for Demanding (Research) Applications How to Modify and Enhance Commercial Controllers (ICRA 2010)*. Citeseer, 2010, pp. 15–21.

[13] G. Silveira, "On intensity-based nonmetric visual servoing," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 1019–1026, 2014.

[14] G. Silveira and E. Malis, "Direct visual servoing: Vision-based estimation and control using only nonmetric information," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 974–980, 2012.

[15] G. Silveira, L. Mirisola, and P. Morin, "Decoupled intensity-based nonmetric visual servo control," *IEEE Transactions on Control Systems Technology*, 2018.

[16] B. Thuilot, P. Martinet, L. Cordesses, and J. Gallice, "Position based visual servoing: keeping the object in the field of vision," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, vol. 2. IEEE, 2002, pp. 1624–1629.

[17] W. J. Wilson, C. W. Hulls, and G. S. Bell, "Relative end-effector control using cartesian position based visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 684–696, 1996.

Carl Friedrich TUK; a Social Companion Robot

Fabian Garber and Nima TaheriNejad

***Abstract*— With improvements in electronics and mechanics, robots have become more compact as well as more space and energy efficient. Hence, they are now a more integral part of our everyday lives. Thanks to Artificial Intelligence (AI) they are on the verge of entering our social lives too. Following this trend, Technische Universität Kiwi (TUK) is a family of social robots developed and under further development at the Institute of Computer Technology at TU Wien. The project deals with the design and creation of a companion robot. The main purpose of this work is to realize a relatable robot which can eventually serve in therapeutic applications, in particular for the children on the autism spectrum. To this end, the companion robot should be able to interact with the user and express emotions. The goal of the companion robot is to create a safe environment by serving as a safety blanket, in particular where other aids such as therapeutic pets cannot be used. Ultimately, we hope that by collecting helpful data, the companion robot can contribute to the therapy procedures as well as improvement of daily life interactions with family and friends. In this paper, we present Carl Friedrich, the first of TUK family.**

I. Introduction

Autism Spectrum Disorder (ASD) describes a variety of disorders including a wide range of symptoms, characteristics and levels of disability. The characteristics include, but are not limited to: social difficulties such as in communicating and interacting with others, repetitive behaviors and also a lasting, intense focus and interest in certain topics [8]. People with ASD may also show extreme sensitivity to the environment, including light, noise and temperature sensitivity. About 1 in 59 children is identified with ASD according to estimates from Centers for Disease Control and Prevention (CDC)'s Autism and Developmental Disabilities Monitoring (ADDM) Network [6] [2]. Thus, enabling them and better integrating them in the society would be advantageous not only for them, but also for the society.

Treatments and therapies can help in overcoming the individual's challenges while also helping them to learn new skills and build up on their strengths. Due to the wide spectrum of the disease there is no single best treatment. Therefore, working closely with a specialist is an important part of finding the right treatment. Using a companion, such as pets, has shown positive effects on the autistic individuals [3]. However, other than hygiene and allergic concerns, which prevents them from a constant companionship, a major issue with pets is their limited life time. Their death can have a significant negative effect on the emotional state of their companions, in particular for the people on the ASD who are often emotionally more sensitive. Moreover, the predictability of the robot provides an emotionally more secure environment for children with ASD. It can also help the children to understand and "learn" empathy [4]. Furthermore, a well designed robot can collect proper information, such as emotional trigger, improvements in behavior, and more detailed information of in-home therapy works, and provide it to the therapist to improve the therapy procedure. In addition, other than serving as a safety blanket and calming the individual which helps in their social interactions, the companion robot could learn about particular behaviors and habits of the autistic individual and communicate these to their family, friends, and colleagues to enable them to understand and interact better with the autistic individual.

Although, the final goals of project TUK is to create social robots for autistic children, its scope includes others who could benefit from a social companion robot as well. The first step of this project, which is presented here, is to create a robot which can show emotional expressions and interact with a user. To this end, different from other companion robots such as Milo [1], we have tried to avoid "Uncanny Valley" [7], while creating emotional expressions that are understandable for humans. We contend that this helps the user to accept the robot as a new being and as-is, and relate to it better, as opposed to consciously or unconsciously comparing it with real humans (as is the case for many humanoids). Thus, the unsettling negative feelings due to this comparison which impede the social bonding can be avoided. In the rest of this paper, we introduce Carl Friedrich who is the first member of TUK family.

II. Design and Expressed Emotions

The first prototype of TUK, as seen in Figure 1, is named Carl Friedrich. This prototype and the Technische Universität Kiwi (TUK) family, is inspired by the flightless birds called "Kiwis" which are native to New Zealand. Carl Friedrich consists of 6 sensors around its body, 8 servo motors and two VGA cameras in the eyes. The servo motors are responsible for the movement of the eyes and the eyelids. The eyes can move vertically and horizontally. The eyelids are not only able to close and open, but also to rotate inwards and outwards. Thus, acting as what would be eyebrows, and help in expressing emotions. The body and the head are fixed and covered in a green soft fabric. The eyelids are covered in purple fabric and the eyes in white fabric. This helps Carl Friedrich to look more like a stuffed animal rather than a robot. We believe that this facilitates bonding with it. For a fluffier experience there is also stuffing beneath the green fabric.

Authors are with the Institute of Computer Technology, TU Wien, Vienna, Austria, Email: {e1425023, nima.taherinejad}@tuwien.ac.at

Fig. 1. First Prototype of TUK, named Carl Friedrich.

For sensing, capacitive sensors are used and each of the sensors trigger a different emotion. The sensors are located on the front, back, sides, head and nose. The processing and computation power are provided by two ATmega 328p micro-controllers. One is used for processing sensor information and the second one for decision-making and controlling the actuators.

The emotions used for Carl Friedrich are inspired by the six different basic facial expressions as identified by Paul Ekman [5]. All the emotions are displayed in Figure 2 and are described as follows:

1) Disgust: One eye is closing a bit.
2) Surprise: The eyelids are opening fast and remain high.
3) Sadness: The eyelids rotate outwards and open slightly.
4) Happy: The eyes are closing slowly to indicate pleasure.
5) Fear: In the current development stage, the same movement as surprise are performed.
6) Anger: Eyelids turn inwards and close a bit.

In addition to the facial expressions, Carl Friedrich also starts to blink after having no interactions for 15 seconds. This is an attempt to make it seem more realistic and alive. Furthermore, following the examples of real-life owls it is alternating the blinking eye. That is, when blinking, only one eye closes at a time and for the next blink the other eye will be closed. This has the advantage of not losing the visuals for the future image processing and facial recognition software.

III. Future Work

One of the future plans is to implement a facial recognition software, in order to have visual interactions as well. That is also the reason for installing the cameras. However, for this purpose additional hardware for processing is needed. Moreover, the robot can be used as a hub for collecting other physiological signals such as Electrocardiography (ECG), heart rate or skin temperature. Using those signals for better identification of the emotion, Carl Friedrich should react to the registered feelings by expressing emotions itself. The aim of adding additional features is to make Carl Friedrich more expressive. Moreover, using such additional devices TUK can be used for therapeutic games as well. In its current position, Carl Friedrich can only express itself with the help of its eyelids. We plan to enable future members of the TUK family to move their beak for more expressiveness. The beak should have the ability to move upwards and downwards, to amplify feelings like sadness or surprise, or shiver to express fear. Also Carl Friedrich should be able to differentiate between getting petted and getting hit, and react in different ways. For a more realistic behaviour the robot should be able to rotate its head. The possibility to feed the robot can both serve as a reinforcement mechanism for learning in the robot and also for improving the relationship between the user and the companion robot. Last but not least, a case study needs to be conducted to evaluate how children react to Carl Friedrich.

References

[1] "https://robots4autism.com/milo/," accessed: 2019-04-23.
[2] J. Baio *et al.*, "Prevalence of autism spectrum disorder among children aged 8 years autism and developmental disabilities monitoring network," in *Morbidity and Mortality Weekly Report: Surveillance Summaries*, vol. 63, 2014, pp. 1–21.
[3] A. Curtis *et al.*, "Dance Dance Pleo: developing a low-cost learning robotic dance therapy aid," 2011, pp. 149–152.
[4] K. Dautenhahn *et al.*, "Kaspar a minimally expressive humanoid robot for humanrobot interaction research," vol. 6, 2009, pp. 369–397.
[5] T. Hashimoto *et al.*, "Development of the face robot saya for rich facial expressions," in *2006 SICE-ICASE International Joint Conference*, 2006, pp. 5423–5428.
[6] J. R. Lucker and A. Do, "Auditory hypersensitivity and autism spectrum disorders: An emotional response," in *Autism Science Digest: The Journal of Autismone*, 2014, pp. 103–108.
[7] M. Mori, "The uncanny valley," in *Energy, 7(4)*, 1970, pp. 33–35.
[8] F. H. Uta Frith, "Autism spectrum disorder," in *Current Biology*, vol. 15, no. 19, 2005, pp. 786–790.

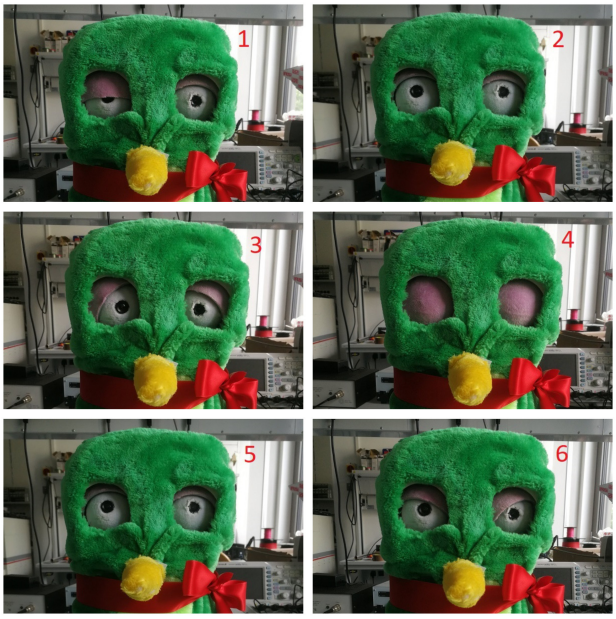


Fig. 2. The six emotions of TUK: 1) Disgust, 2) Surprise, 3) Sadness, 4) Happiness, 5) Fear, 6) Anger.

A case study on working with a commercial mobile robotics platform

Clemens Koza[1] and Wilfried Lepuschitz[2]

***Abstract*— During a period of roughly one and a half years, the authors had the opportunity to work with a commercial mobile robotics platform, namely the "Apollo" platform by Slamtec, to implement real-world use cases in the domain of interactive entertainment installations. During the course of this development, the authors have gained insight into the strengths and limitations of the platform in realistic scenarios. This work will present the use cases, and discuss the experiences with and insights into working with the Apollo platform.**

I. INTRODUCTION

In the domain of interactive multimedia entertainment systems, there are many established forms of installations, from ordinary touch screens, to presence and gesture sensitive spaces, to multisensory "5D" cinemas. Mobile robots present an additional form of visitor interaction. However, entertainment installations bring their own set of challenges for a mobile robot, including an environment with a highly variable number of untrustworthy actors such as kids, and generally a lack of specialized personnel for supervision and upkeep.

II. APOLLO PLATFORM

The authors used the "Apollo" platform by Slamtec [1] to implement two real-world entertainment use cases. The cylindric robot weighs 40kg and has a diameter of 500mm. It has two driven wheels and four smaller casters for support. Payloads up to 35kg can be mounted on the circular top.

Among the robot's sensor systems is a LIDAR with a 270° field of view and 15m maximum scan radius, a depth sensor and six front-facing ultrasonic distance sensors. Hardware interfaces include WiFi and Ethernet, and a 20-25.2V 5A DC power interface to power additional devices on the robot.

Apollo has built-in simultaneous localization and mapping (SLAM), path planning with obstacle avoidance, and can dock with its charging base autonomously [1]. The "RoboStudio" software allows to view and edit the robot's map, edit the position of the home dock, inspect the robot's sensors, and to move the robot to specific positions [2].

The robot is controlled using the "Slamware SDK" (software development kit), which is provided as precompiled binaries for Windows, Linux, iOS and Android. The authors have only worked with the Linux SDK, although working with the Windows version was attempted as well.

The authors are with the Practical Robotics Institute Austria, Wexstraße 19-23, 1200 Vienna, Austria; [1]koza@pria.at; [2]lepuschitz@pria.at

III. DESCRIPTION OF USE CASES

A. Robot Swarm

In a first scenario, four robots decorated as animals were programmed to roam a small area of 3.5m x 5m. Three sides of the site were walled and one side open, providing clear landmarks for SLAM. Charging docks were placed against the three walls. Figure 1 shows a map of the space created by Apollo's SLAM system. The robots showed three basic behaviors:

- When a visitor is present in the area of operation, the robots head towards them. After a randomized timeout, they "lose interest" in visitors and continue roaming.
- When no visitors are present, or for a minimum amount of time after a robot "lost interest", the robots roam around the area in a random fashion.
- When their battery runs low, robots return to their charging stations.

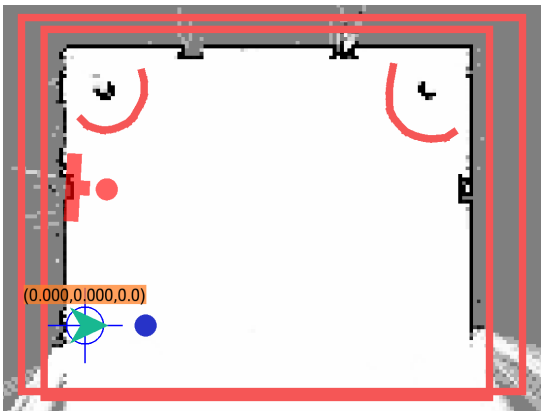


Fig. 1. A SLAM map of the robot swarm operation area. Displayed in red are virtual walls around the area and two hard to recognize chairs, and the first robot's charging dock on the left wall.

B. Greeting Robot

In the second scenario, two robots were used as mobile terminals, with touch screens mounted on top of the platform. These robots operated outside in a roofed area, on a rough concrete floor with some pebbles left on the ground.

Apart from some columns around the area, only the glass walls of the neighboring building were available as landmarks. Each robot had its own area of operation where it executed one of three behaviors:

- When a visitor is present in the area of operation, the robot heads towards that person and greets them.

- While a visitor is interacting with the touch screen, the robot does not move at all.
- Otherwise, the robot drives along a fixed path.

Automatic charging was of no concern, as the chargers were kept indoors and the duration of operation was short.

IV. FINDINGS

A. General findings

Some robots were found to turn in circles or sporadically change direction for no apparent reason. Testing showed that this happened because of faulty ultrasonic distance sensors. Some sensors did not recognize obstacles, while others reported false positives. False negatives did not generally impact the robot's behavior, as the LIDAR readings were accurate and reliable enough for obstacle avoidance.

While Apollo incorporates a depth camera, the camera's data can not be accessed by client applications. It is used solely for recognizing obstacles such as desks that are not visible to the LIDAR, which is mounted at a height of roughly 240mm. The depth camera would be a great tool for obstacle identification or human pose recognition [3], which makes it regrettable that it is not usable for these tasks.

Some features in the library turned out not to be implemented. For example, movement commands support specifying the desired speed. The speed option would be ignored however; support suggested to set the system-wide speed prior to any movement command instead.

B. Findings from the "Robot Swarm" use case

The defining requirement of this scenario was the need for coordination between robots: with detecting visitors limited to Apollo's LIDAR sensor, robots had to be aware of each other to not mistake each other for humans. An architecture with a centralized controller was used. That controller had an ethernet connection to a WiFi router, which would then connect to the robots. As Apollo has two strong WiFi antennas, this allowed for a reliable connection.

Although the Android Slamware SDK includes functions for Domain Name System Service Discovery (DNS-SD), the authors were not able to find robots by a service identifier or host name on the local network. This necessitated a more complicated design than was originally anticipated.

Apollo does not support coordinated path planning, e.g. as proposed in [4]. There is no shared coordinate system among robots and each robot is only aware of itself. It was observed that two robots driving towards each other would not recognize each other in time to prevent a collision. This was largely addressed by avoiding "risky" goals on the controller. Still, as Apollo's path planner may choose a different path from what the controller anticipated, this risk assessment could not be perfect and additional measures had to be taken.

While obstacle avoidance is a built-in feature of Apollo, target tracking had to be implemented manually. The authors first tried to let Apollo plan a path to the target. This did not work, as each update to the target position would stop the robot before potentially trying a vastly different path. Instead, using direct motion commands was necessary to track targets. For this, straight motion and stationary rotations are available; driving circular arcs is not supported. Targeting using direct motion commands worked very well, but these bypass obstacle avoidance, requiring greater care in the client code.

Finally, the presence of four robots, plus potentially multiple visitors, meant that a significant amount of stationary SLAM landmarks were obscured at any point in time. This impacted robot localization, which led to robots occasionally mistaking other robots for visitors. Fortunately, the correct location would be recovered eventually, and the swarming of robots around another did in fact look very fitting.

C. Findings from the "Greeting Robot" use case

The greatest challenge of the second scenario was the outdoor environment. As rough ground, pebbles and weather are hard to predict, commercial mobile robots are generally not advertised as suitable for outdoor operation. Apollo is no exception, but it operated fairly well, as long as the operation area was cleaned from pebbles every day before starting the robots. Apollo was able to recover from getting stuck on the rough ground fairly reliably, although getting stuck frequently led to jagged motions. Larger casters would be necessary to improve Apollo's outdoor driving performance.

The adjacent buildings had glass fronts with wall decals; the transparency made the fronts unsuitable as SLAM landmarks. An opaque tape was added to the fronts, which immediately remedied the problems.

It was observed that the low sun disturbed the operation of the LIDAR. This was no problem during operation, as the desired operation hours were after sundown, but limited the realism of testing conditions during the rest of the day.

V. CONCLUSIONS

Although Apollo showed some shortcomings, especially regarding multiple robots operating in the same space, it is a capable platform. The authors found that the actual tasks were relatively easy to handle, but details around the setup, such as discovery on the local network and map management, could be presented to application developers in an easier fashion.

The fact that Apollo is closed-source and is not using Robot Operation System (ROS) must be considered when evaluating the platform, especially if Apollo is to be added to an environment that is already based on ROS.

REFERENCES

[1] Apollo medium robot development platform parameters. [Online]. Available: https://slamtec.com/en/Apollo/Spec

[2] Robostudio extendable robot management and development software. [Online]. Available: https://slamtec.com/en/RoboStudio

[3] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2011, pp. 1297–1304.

[4] T. Siméon, S. Leroy, and J.-P. Lauumond, "Path coordination for multiple mobile robots: A resolution-complete algorithm," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 1, pp. 42–49, 2002.

Traffic cone based self-localization on a 1:10 race car

Axel Brunnbauer[1] and Markus Bader[1]

***Abstract*— This document describes a feature-based self-localization running on-board on an automated 1:10 race car. Traffic cones are detected using an Intel RealSense depth camera and integrated into the self-localization as landmarks. The work presents a novel approach for how to detect traffic cones by fusing depth and RGB data. Motion commands and sensor readings are combined with an Extended Kalman Filter (EKF). The performance of the overall system was evaluated using an external motion capture system.**

I. INTRODUCTION

Learning the technical principles of autonomous driving with a real car is not only expensive but also dangerous. Therefore our work group uses multiple car models scaled at 1:10 with self-printed components for teaching, research and developing new approaches.

This work presents a localization system that is able to detect traffic cones in various contexts and, as a consequence, localize the vehicle using a prior known feature map. The car shown in Fig. 1 is an enhanced version of the model presented in [2]. The car is equipped with a depth camera and an Single Board Computer (SBC) running a high-level processing unit with Robot Operating System (ROS). The Brushless Direct Current (BLDC) motor for driving and the servomotor for the front Ackermann steering are controlled by a self-designed board connected by a serial cable to the SBC. A more detailed explanation of the model's components is provided later.

One challenging aspect of designing such a system is the detection of traffic cones. Difficulties in object detection are presented by various factors. Camera-based detection methods, in particular, are greatly influenced by changing lighting conditions. Currently, the car provides simple odometry by dead reckoning using Hall effect sensors on the motor and no wheel encoders. Therefore the number of revolutions of the motor and the state of the servomotors are used to calculate odometry. However, these measurements are subject to noise introduced by external and internal factors. External factors, on the one hand, constitute calibration errors regarding wheel diameter, wheelbase distance, slightly different wheel sizes, etc. Internal factors, on the other hand, are inaccuracies in the mathematical model used to describe the motion of the vehicle. Since the model is an abstraction of the real world, simplifying assumptions are necessary to keep the model feasible. The uncertainty presented by the errors in odometry and the sensor readings must be considered when building a reliable localization system. To cope with these issues, this work applies a variant of the Extended Kalman Filter (EKF) algorithm, which can deal with these types of uncertainties. Some other approaches dealing with localization and mapping of traffic cones are presented below, followed by a discussion of the methods applied in this project. Finally, the results of the experimental evaluation are compared with the current performance of the system.

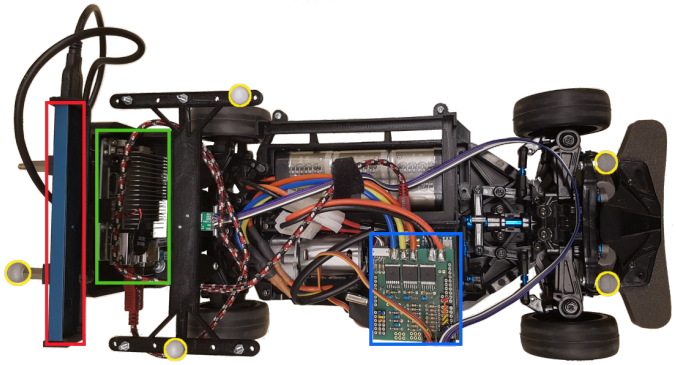


Fig. 1: 1:10 race with an on-board PC (green), a depth camera (red), a motor control board (blue) and reflectors (yellow) to record ground truth pose data with a motion capture system.

II. RELATED WORK

In [1], the development of the *flüela driverless*, the first car to win the Formula Student Driverless (FSD), was introduced. The *flüela driverless* is an electrically-powered race car equipped with a variety of sensors for traffic cone detection and state estimation, including a 3D LiDAR sensor, a stereo camera system, an inertial navigation system and an optical ground sensor. The vehicle is able to map the track defined by pairs of traffic cones using a particle-filter-based SLAM approach. Further, for localizing the car, a landmark-based EKF was implemented. The previously-mapped traffic cones serve as landmarks used for updating the current estimation of the vehicle's pose. To detect the traffic cones, the team used a 3D LiDAR scanner. Traffic cones are recognized by first removing the ground plane from the scan. Then, the filter clusters the traffic cones using the Euclidean clustering algorithm and further classifies these clusters according to size. In a second filtering step, traffic cones get rejected based on their distance from the sensor. The approach presented in this work also filters traffic cones based on their size. In addition, it rejects objects which do

*The research leading to these results has received funding from the Austrian Research Promotion Agency (FFG) according to grant agreement 855409 (AutonomousFleet) and 854865 (TransportBuddy)

[1]Axel Brunnbauer and Markus Bader are with the Faculty of Informatics, Computer Engineering, Vienna University of Technology, Vienna 1040, Austria, axel.brunnbauer@tuwien.ac.at

not fit the visual appearance of the traffic cones.
Zeilinger et al. present their approach to designing an autonomous vehicle for the FSD in [5]. Their system uses a planar laser scanner and color cameras for the detection of traffic cones. Furthermore, to estimate the vehicle's movements, a rich set of sensors, including IMU's, GPS sensors, several wheel spin sensors and rotary encoders are mounted on the car. The traffic cone detection includes segmentation by color, exploiting the distinct color of the traffic cones. Also, techniques to incorporate the data from the stereo cameras and the laser scanner are presented. To map the traffic cones and localize the vehicle, an EKF-SLAM implementation was used. The work proposed in this paper also proposes an EKF implementation for localization, but does not take SLAM into consideration. The trajectory planning in [5] also exploits the fact that the pairs of cones already describe a path which have to be followed. This path is then smoothed using a model-predictive motion controller.

III. APPROACH

In the approach presented here, traffic cones are first detected and then located relative to the pose of the vehicle. Next, mapping is determined based on the perceived location of the landmarks and a known map.

A. Traffic Cone Detection

The algorithm developed for this task uses the guess and check method. First, objects in the vehicle's environment are located by extracting objects from the 3D point cloud. This is done by simulating a laser scan parallel to the ground. After potential candidates are detected, the algorithm filters out objects which do not correspond to distinct features of the landmarks. The first step of the landmark detection process is the extraction of objects from the laser scan. To extract objects from the set of points $P = \{p_1, p_2, \ldots, p_n\}$, the Euclidean clustering method proposed in [3] is used.
Filtering of the objects detected is realized as a filter pipeline where each filter takes the clusters extracted as an argument. In addition, each filter has individual input arguments necessary for detecting certain features of the objects. Currently, the pipeline consists of two filters. The first filter applied sorts out objects which do not correspond to the expected width of traffic cones, as estimated by the laser-based detection. Fig. 2 shows the laser scan obtained

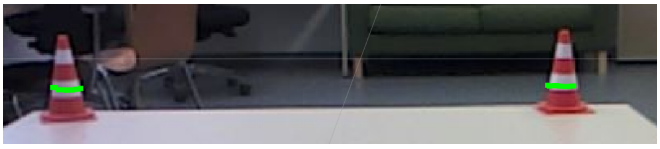
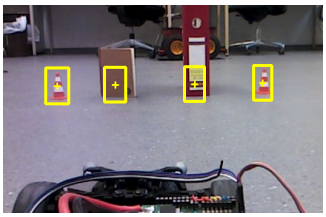
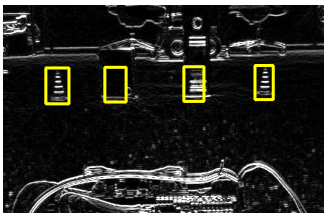


Fig. 2: Visualized laser scan readings in green on detected traffic cones

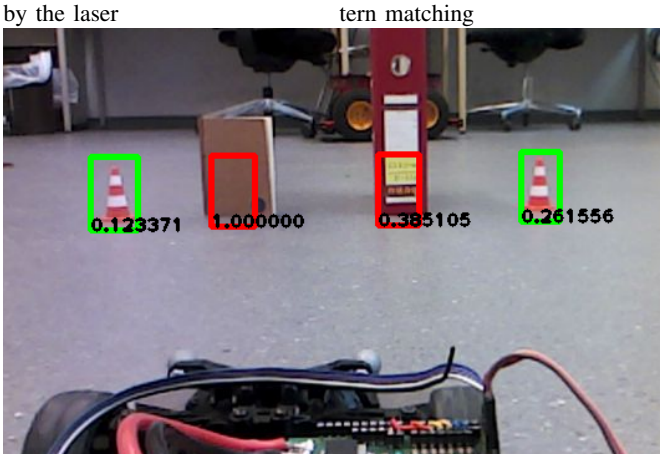
from traffic cones. If a segment does not correspond to a certain degree to the expected width, the segment is removed from the traffic cone candidate list.
The image-based filter exploits the visual properties of a traffic cone, especially its striped appearance. The classification of traffic cones is done using a template-matching algorithm. The goal of the algorithm is to match objects which correspond to the typical striped pattern of traffic cones. By applying the *Sobel* operator to find horizontal edges, unnecessary information is removed and the pattern of the traffic cones can be emphasized. Fig. 3b shows the extracted edges of the image. To speed up the template-matching process, the search space is reduced to a narrow area around the projected point. This area is defined by the traffic cone's height and width. First, the coordinates of the objects detected in the previous step are projected onto the image plane using extrinsic and intrinsic camera parameters. Now that the location of the candidate objects in pixel space is known, the search space is limited by the surrounding bounding boxes. In Fig. 3a, search spaces are marked with yellow bounding boxes. The marker inside a bounding box depicts the center of the detected object which is projected onto the image. The next step involves matching the sub-



(a) Cone candidates, estimated by the laser



(b) Horizontal edges used for pattern matching



(c) Final estimatation of traffic cones

Fig. 3: Image processing steps for traffic cone detection

regions of the image, defined by the candidate objects, with templates. For each bounding box in Fig. 3b, the template matching algorithm determines which template matches best. The method used for calculation is the *normalized sum of squared differences*, resulting in a value between 0 and 1. Values close to zero indicate strong similarities between a template and a candidate object. Then, a simple threshold operation is applied in order to reject regions with a matching score of higher than a certain value. Fig. 3c shows such matching using a threshold value of 0.3, which yielded good results throughout several experiments under various lighting

conditions. The green bounding boxes indicate objects which have been matched, whereas the red ones indicate rejected candidate objects. A matching score is displayed at the bottom of each bounding box.

Fig. 4: Templates in different resolutions

Fig. 4 shows the templates used for template-matching in different resolutions and pre-processing steps. From left to right, each template is converted to a gray-scale image and then the horizontal edges are extracted. To get robust results, about forty templates were recorded from various angles and distances of up to 1.5 metres.

B. Localization

For the localization of the vehicle, a landmark-based EKF with unknown correspondences was implemented. Limited computational power and simplicity of implementation informed the selection of this approach. Implementation is based on the localization approach for unknown landmark correspondences, as proposed in [4]. Detected traffic cones are associated with the map using an Maximum Likelihood (ML) estimator which maximizes the probability of a detection being the landmark actually observed. To address the problem of false data association, a distance metric is used to reject landmark associations exceeding a predefined threshold. The metric of choice is the *Malahanobis* distance, as it is scale-invariant and unit-less. For a detected landmark $\boldsymbol{z}^i$ and landmark $\hat{\boldsymbol{z}}^j$ and its covariance $\boldsymbol{S}^j$, chosen by the ML estimator, the constraint shown in (1) must be satisfied.

$$\sqrt{(\boldsymbol{z}^i - \hat{\boldsymbol{z}}^j)^T [\boldsymbol{S}^j]^{-1} (\boldsymbol{z}^i - \hat{\boldsymbol{z}}^j)} < d_{max} \tag{1}$$

The EKF algorithm was implemented as closely as possible to the original. One challenge was embedding the algorithm into the ROS environment, since the original algorithm leaves out details on how to handle the complexity introduced by the dimension of time. In the real world, sensor readings and control updates are not received simultaneously or even at fixed intervals. The algorithm presented simply processes control and measurement updates in the order in which they arrive. That means that as soon as a measurement z arrives at time t, it is treated as it is perceived at the time it arrives. This causes the pose estimation to be biased in terms of it being always a few moments behind the actual pose. The reason for this is that when a measurement z is obtained at time t, the processing delay ε induced by the image detection is neglected. To separate the control updates from the measurement update, a global pose estimate is obtained via a series of relative transformations. Fig. 5 shows an example of such a transformation hierarchy.

The *map* frame depicts the initial pose at which the localization began. Control updates are successively incorporated into a transformation between a coordinate frame *odom* and

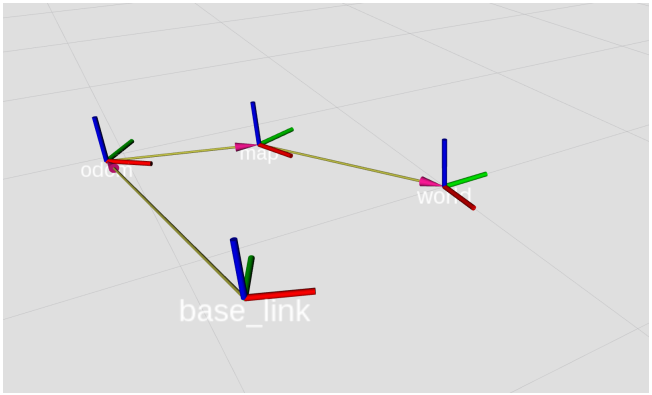


Fig. 5: Transformation hierarchy

a *base_link* frame. The transformation between these two frames corresponds to the position estimated by the odometry. Furthermore, a control update history is maintained. An example can be found in Fig. 6 in the *Odometry* queue. For each control update u_t, the most recent pose x_{t-1} relative to the *odom* frame's origin is selected from the queue. Then the control u_t is applied to x_{t-1} and Σ_{t-1}, respectively, and the result is appended to the queue. To determine the correction in this approach, another transformation between the *map* frame and the *odom* frame is applied. When a measurement z_t arrives at time $t+\varepsilon$, the last known pose before time t is queried from the queue. This pose is then transformed into the *map* coordinate frame and the EKF algorithm computes the correction. The transformation from the *map* frame to the *odom* frame can now be calculated by simply subtracting the transformation between *odom* and *base_link* from the correction computed. Since the approach

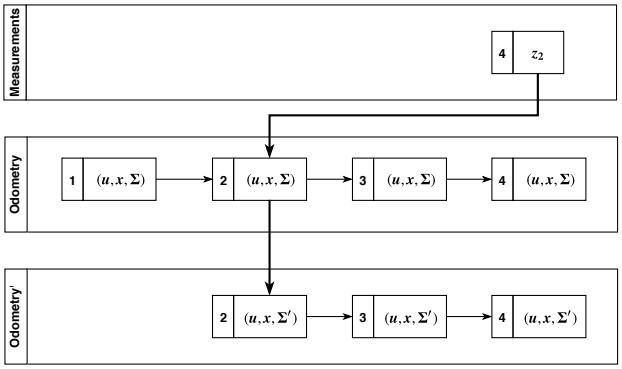


Fig. 6: Covariance update

so far only considers the vehicle's pose, another step is necessary to maintaining the covariance. The covariance of the queue entry at time t is replaced by the covariance of the corrected pose. Then, the motion model gets reapplied iteratively to all entries afterwards, while all entries prior to the one selected can be dismissed. Fig. 6 shows such a case. At time 4, a measurement obtained at time 2 arrives. The algorithm searches for an appropriate entry and makes the correction. Afterwards, the covariances are updated and all entries with a timestamp of under 2 are dismissed.

IV. EXPERIMENTAL RESULTS

To evaluate the approach presented so far, the proposed models are tested with a 1:10 scaled race car. The experiments were carried out in the case studies laboratory at the Institute. The laboratory is equipped with an OptiTrack[1] motion capture system allowing for the localization of visual markers with sub-millimeter accuracy. Fig. 7 shows the laboratory with the motion capture system indicated by red boxes. As shown in Fig. 1, the car is equipped with visual markers which can be localized by the cameras.

Fig. 7: Case studies laboratory, IR-cameras indicated in red.

A. Traffic cone detection

The evaluation of landmark detection accuracy was done by comparing the measurements with the true position of a traffic cone determined by the motion capture system. The dataset used contains about 4,000 measurements obtained from various distances and angles while driving the vehicle towards a single traffic cone with varying speeds. Fig. 8 shows the error of ρ in meters and the error of ϕ in radians. The data retrieved from the measurements suggests that uncertainty originates mainly from the distance measurement. The mean error of ρ, however, is approximately 4cm and therefore of systematic nature and can be compensated for easily. Below, Fig. 8 visualizes the distribution of the measurement errors. While the error of ϕ is almost distributed evenly around zero, the error of ρ is skewed towards the positive. The co-variance matrix used in the EKF algorithm was derived from this data, represented by the light blue covariance ellipse. Fig. 9a suggests that the error of ρ correlates with the distance to the traffic cone, while it is not significantly influenced by the angular offset to the vehicle. By applying polynomial fitting, a function describing the error of ρ regarding the distance to the traffic cone could be obtained. The function derived could be used to improve the localization algorithm by replacing the static sensor model by a dynamic approach.

B. Localization

To evaluate the accuracy of the localization algorithm, the vehicle was navigated through a number of test tracks. For each run, both the estimated pose, using only odometry, and the corrected pose, using traffic cone detection, were recorded simultaneously and compared.

[1]OptiTrack - Motion Capture System: https://optitrack.com/

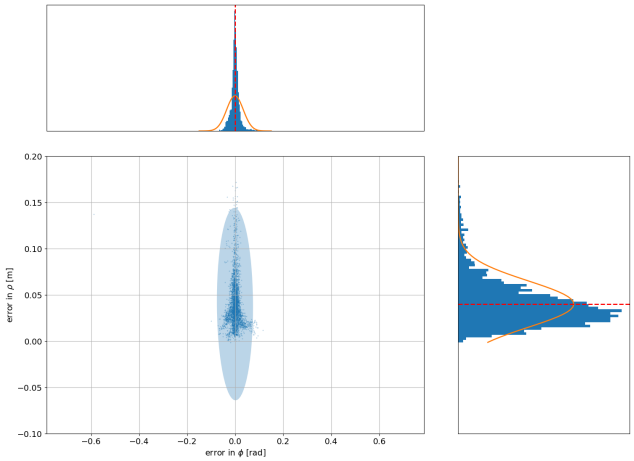
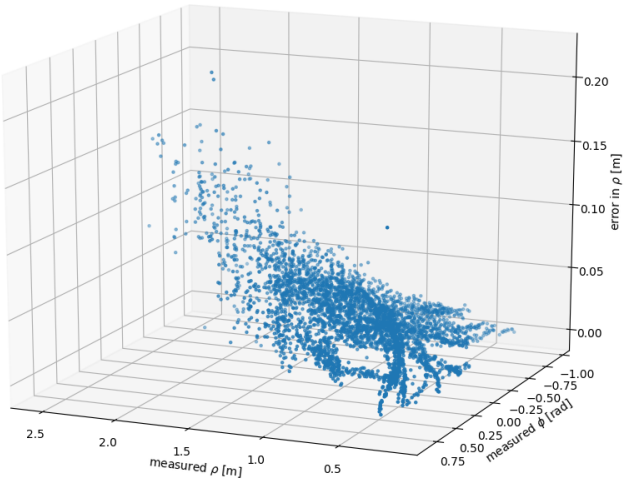


Fig. 8: Measurement error

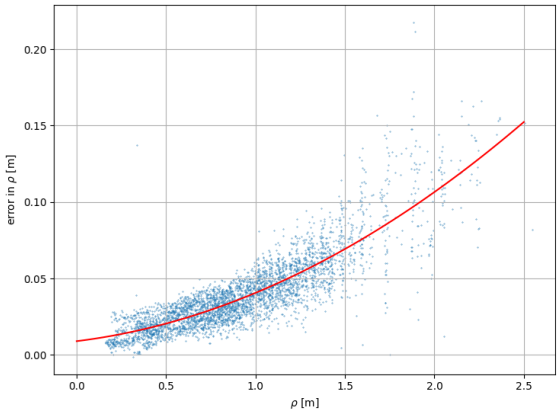
Fig. 10 shows the performance of the self localization on a simple straight course. The blue points depict the ground truth position of the car obtained by the motion capture system. The orange points show the estimated position of the vehicle based on either odometry only (10a) or odometry combined with measurement updates (10b). The ellipses around each orange point show the covariance of the pose estimation and the blue crosses represent the traffic cones defining the course. A relatively strong deviation from the actual pose in Fig. 10a shows the inaccuracies of the mechanical configuration of the vehicle. One can see that the odometer always underestimates the actual speed of the vehicle. Furthermore, the vehicle tends to drift to the left. Therefore, some counter-steering was necessary to keep the vehicle on track when driving through the straight course.

This is the cause for the odometry-based estimation in Fig. 10a drifting towards the negative y-quadrant. However, correction using traffic cone detection yields a significantly better estimation of the true pose, as can be seen in Fig. 10b. The deviation induced by drift could be compensated for almost completely. The evaluation reveals as well that the deviation from the true pose, interpreted as Euclidean distance, was able to be maintained at a relatively stable level when using the measurement update, while the approach using solely odometry shows that the vehicle tends to fall further behind the longer the run is.

Fig. 11 shows another test run through a slalom-like course. Again, the odometry base pose estimation shows great inaccuracies after a short period of time. Correction led to a more accurate estimation of the pose in this run, as well. In Fig. 11b, it can be seen how the vehicle's drift affects the estimation. After the second pair of traffic cones in a left turn, the deviation between actual position and estimation grows, due to the tendency of the vehicle to pull to the left. Fig. 12 shows the evaluation of the localization on a circular course. The position estimation in Fig. 12a reveals, again, the sideways drift and the underestimation of the vehicle's speed.



(a) Error w.r.t location



(b) Fitted error function of ρ

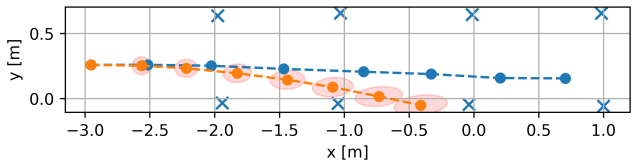
Fig. 9: Error with respect to different distances

Whereas the car completed almost a whole lap on the course, the odometry-based estimation supposed that the vehicle completed only half of a lap. Although the correction does not work as well as on the previous courses, the estimation yields an improvement over the odometry estimation.

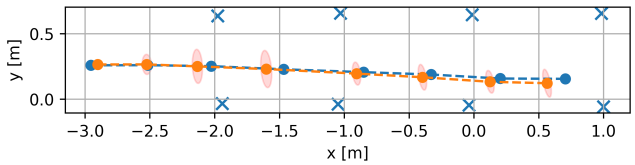
V. CONCLUSIONS

As the evaluation results suggest, the localization of the car can be improved by using the measurements obtained from the on-board camera. Although the odometry yields great uncertainty, pose correction can compensate for these inaccuracies in many instances.

An important step towards a more reliable self localization would be the improvement of the on-board odometry. Currently, only the speed of the wheels is estimated based on the revolutions of the motor and the wheel's diameter. For instance, a wheel encoder could be used to accurately determine the velocity of the vehicle. The localization system, as it is, requires that location of the traffic cones be known. The next step would be the implementation of a SLAM approach, to enable the vehicle to build up a map while driving a course. Potentially, in order to enable a vehicle to drive such a course autonomously, a navigation module could be developed that is able to navigate a path through a sequence of traffic cone pairs without colliding with any obstacles.

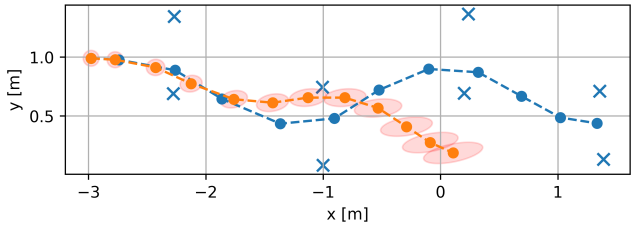


(a) Odometry only

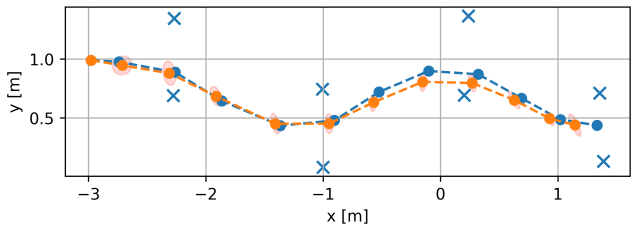


(b) Correction enabled

Fig. 10: *Straight course:* The estimated trajectory is shown in orange, the actual trajectory in blue. The figures show the position estimation of the vehicle based on its odometry, with EKF correction enabled.



(a) Odometry only



(b) Correction enabled

Fig. 11: Slalom course

VI. APPENDIX

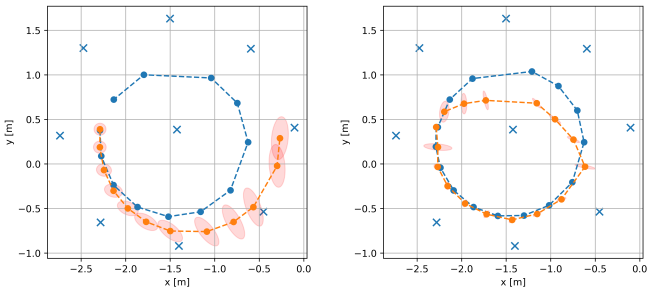
The data sets used to evaluate the components developed throughout this paper can be downloaded from the public repository[2].

REFERENCES

[1] M. de la Iglesia Valls, H. F. C. Hendrikx, V. Reijgwart, F. V. Meier, I. Sa, R. Dubé, A. R. Gawel, M. Bürki, and R. Siegwart, "Design of an autonomous racecar: Perception, state estimation and system integration," *CoRR*, vol. abs/1804.03252, 2018. [Online]. Available: http://arxiv.org/abs/1804.03252

[2] B. B. Eugen Kaltenegger and M. Bader, "Controlling and Tracking an Unmanned Ground Vehicle with Ackermanndrive," in *Proceedings of the Austrian Robotics Workshop (ARW-16)*, Wels, Austria, May 2016.

[2]https://github.com/axelbr/rccar-results



(a) Odometry only (b) Correction enabled

Fig. 12: Circular course

[3] R. B. Rusu, "Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments," Ph.D. dissertation, Technische Universität München, 2009.

[4] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[5] M. Zeilinger, R. Hauk, M. Bader, and A. Hofmann, "Design of an Autonomous Race Car for the Formula Student Driverless (FSD)," in *Proceedings of the OAGM ARW Joint Workshop (OAGM ARW-17)*, W. K. A. M. B. B. Peter M. Roth, Markus Vincze and S. Stolc, Eds., Vienna, Austria, May 2017, pp. 57–62.

Computational Performance of the Forward and Inverse Kinematics of an Anthropomorphic Robot Arm

Christian HARTL-NESIC[1], Martin MEIRINGER[1]

I. INTRODUCTION AND MOTIVATION

In robotics, two fundamental coordinate spaces are utilized, i. e. the so called configuration space and the joint space. Most of the robot tasks are naturally defined in the configuration space (also known as task space) whilst actors and sensors of a robot operate in the joint space (e. g. encoders, electric motors, torque sensors ...). Therefore a bidirectional transformation is desired for describing a vector of one space in the other one.

Modern applications require increasing accuracy during task execution, which demands an increasing bandwidth of the control algorithms. To meet these increasing requirements, fast and efficient algorithms are necessitated for calculating the robot kinematics, i. e. the forward and inverse kinematics. While the forward kinematics describes the end effector pose (position and orientation) in the configuration space as a function of the joint space coordinates and the inverse kinematics describes the joint space coordinates as a function of the end effector pose. Two concepts are investigated in the course of this work: i) homogeneous coordinates, see, e. g., [3] and ii) dual quaternions, see, e. g., [1]. The calculations are performed for the 7-degree-of-freedom (DOF) anthropomorphic arm *KUKA LWR IV+*.

II. COMPUTATIONAL EFFORT FOR KINEMATICS

This work aims to compare the calculation costs of the robots kinematics between two well known methods of describing the kinematics. First, elemental kinematic operations are separately investigated, followed by the forward and inverse kinematics for the 7-DOF robot.

A. Method of Performance Calculation

The computational performance is quantified by the calculation costs, i. e. the number of additions, multiplications and function calls required to execute an algorithm and the mean time to numerically perform a calculation. All presented results, including the costs, where evaluated using the algebra system MAPLE 2018. The numerical results where calculated using MATLAB 2017b, using an office PC equipped with an Intel Core i7-6700K 4 GHz CPU and 16 GB RAM.

*This work was not supported by any organization.

[1]Faculty of Electrical Engineering, Institute for Automation and Control, TU Wien, Vienna, Austria {hartl, meiringer}@acin.tuwien.ac.at

B. Fundamental Transformations

The discrete transformation (translation and rotation) between two coordinate frames using homogeneous coordinates is given by the 4×4 matrix $\mathbf{T}$ given by

$$\mathbf{T} = \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} \\ \mathbf{d} & \mathbf{R} \end{bmatrix} , \tag{1}$$

where $\mathbf{d}$ represents the translation between the two frames and $\mathbf{R}$ denotes the rotation matrix representing the change in orientation. The same transformation, formulated in dual quaternions is given by

$$\underline{\mathbf{u}} = \mathbf{q}_R + \frac{1}{2}\varepsilon \mathbf{q}_R \otimes \mathbf{q}_d , \tag{2}$$

where $\mathbf{q}_R$ denotes the unit quaternion representing the change in orientation, $\mathbf{q}_d$ denotes the quaternion representing the translation between the two frames, ε is the dual unit and $\otimes$ denotes the quaternion multiplication (see [1]). The transformation matrix $\mathbf{T}$ in (1) and the transformation dual quaternion $\underline{\mathbf{u}}$ in (2) are used to perform discrete transformations, e. g. for changing the reference frame of a point in task space. By utilizing the well known Denavit-Hartenberg convention, see, e. g., [3], the relative transformations between two coordinate frames, attached to two consecutive links, can be directly parameterized. Comparing the computational costs of (1) and (2) gives a first idea concerning the computational performance of the two used methods. The costs are summarized in Table II. For each calculation, the costs without code optimization and the costs using the code optimization provided by MAPLE 2018 are given, where the non-optimized results are denoted with the ˜ symbol. Note, the transformations for this comparison where constructed using general, nonzero Denavit-Hartenberg parameters.

C. Forward Kinematics

The forward kinematics is given by the composition of N transformations, where N is the number of degrees-of-freedom. The forward kinematics for the end effector pose is then calculated by

$$\mathbf{f}_H(\theta) = \mathbf{T}_0^N(\theta) = \mathbf{T}_0^1(\theta_1)\mathbf{T}_1^2(\theta_2)\cdots\mathbf{T}_{N-1}^N(\theta_N) , \tag{3}$$

$$\mathbf{f}_{dQ}(\theta) = \underline{\mathbf{u}}_0^N(\theta) = \underline{\mathbf{u}}_0^1(\theta_1)\otimes\underline{\mathbf{u}}_1^2(\theta_2)\cdots\otimes\underline{\mathbf{u}}_{N-1}^N(\theta_N) , \tag{4}$$

where $\otimes$ denotes the dual quaternion multiplication (see [1]). The vector θ in (3) and (4) summarizes the relative joint angles of the robot[1]. The relative transformations $\mathbf{T}_{i-1}^i$ and $\underline{\mathbf{u}}_{i-1}^i$ are parameterized using the Denavit-Hartenberg

[1]The dependencies $\mathbf{T}(\theta)$ and $\underline{\mathbf{u}}(\theta)$ are omitted in the following for brevity.

i	a_i	α_i	d_i	θ_i
1	0	$\pi/2$	0	q_1
2	0	$\pi/2$	0	q_2
3	0	$\pi/2$	l_1	q_3
4	0	$\pi/2$	0	q_4
5	0	$\pi/2$	l_2	q_5
6	0	$\pi/2$	0	q_6
7	0	0	0	q_7

TABLE I

DENAVIT-HARTENBERG PARAMETERS FOR THE *KUKA LWR IV+*.

calculation	add	mult	fcn. calls	mean CPU time
$\tilde{\mathbf{T}}$ in (1)	2	6	14	-
$\mathbf{T}$ in (1)	2	6	4	-
$\tilde{\underline{\mathbf{u}}}$ in (2)	4	52	24	-
$\underline{\mathbf{u}}$ in (2)	4	20	4	-
$\tilde{\mathbf{f}}_H(\theta)$	88	238	327	4.109 µs
$\mathbf{f}_H(\theta)$	40	87	14	3.642 µs
$\tilde{\mathbf{f}}_{dQ}(\theta)$	312	1576	880	19.510 µs
$\mathbf{f}_{dQ}(\theta)$	38	88	14	4.398 µs
$\mathbf{f}_H^{-1}(\mathbf{T}^\star)$	1461	2570	52	86.138 µs
$\mathbf{f}_{dQ}^{-1}(\mathbf{T}^\star)$	1070	2258	184	173.104 µs

TABLE II

COST COMPARISON FOR FUNDAMENTAL TRANSFORMATIONS, FORWARD AND INVERSE KINEMATICS.

parameters given in Table I. The resulting calculation costs for the forward kinematics are summarized in Table II.

D. Inverse Kinematics

The inverse kinematics is a more challenging task and often not solvable analytically. Also the number of possible solutions can vary from zero to infinite solutions. For the inverse kinematics, the nonlinear equations (3) and (4) have to be solved for θ. This problem is formulated as

$$\theta = \mathbf{f}_H^{-1}(\mathbf{T}^\star) \,, \tag{5}$$

$$\theta = \mathbf{f}_{dQ}^{-1}(\underline{\mathbf{u}}^\star) \,, \tag{6}$$

where the superscript $\star$ denotes a given end-effector pose. The *KUKA LWR IV+* comprises a spherical sholder joint, an elbow joint and a spherical wrist, see, e. g., [3], this allows an analytical solution of the problems (5) and (6). Thanks to this construction, the kinematics can be separated in a position and orientation task, since the last three joints, the spherical wrist, only change the end effector orientation while leaving its position unchanged. Due to the redundancy of the considered robot, one joint angle remains an independent DOF. In this work, θ_3 is chosen as DOF. For the considered robot, the inverse kinematics problem results in eight sets of one-parametric solutions.

The inverse kinematics implementation, formulated using homogeneous coordinates, follows the work of Pfurner [2]. The method for calculating the inverse kinematics, formulated in dual quaternions is based on the arguments given by Pfurner. The different formulation of the kinematics changes the analytic structure of the obtained equations,

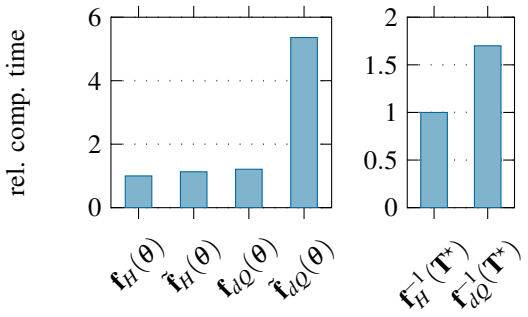


Fig. 1. Mean computation times for the forward and inverse kinematics.

but the dependencies of the equations remain unchanged. The resulting system of equations is solved using MAPLE 2018 following the procedure presented in [2]. The resulting calculation costs for the inverse kinematics are summarized in Table II. For practical applications, in addition to the calculation costs also the mean calculation time to execute an algorithm is of interest. These are also summarized in Table II and visualized in Figure 1. Solving the position part of the inverse kinematics problem, the dual quaternion formulation utilizes twice the analytical solution of a 4th degree polynomial. However, this yields multiple invalid solutions, which have to be eliminated using an additional equation. Thus, the homogeneous coordinate formulation is significantly faster.

III. CONCLUSION

In this work, the forward and inverse kinematics of a 7-DOF anthropomorphic arm, the *KUKA LWR IV+* were formulated using i) homogeneous coordinates and ii) dual quaternions and evaluated with respect to the computational costs and mean computation time.

By utilizing the code optimization functions provided by modern computer algebra systems, e. g. MAPLE 2018, the forward kinematics implementations perform almost equally, while the homogeneous coordinate formulation is slightly faster. The inverse kinematics implementations show a more significant difference in computation time, whereas the dual quaternion implementation is about 70 % slower. Hence, if the robot task does not prescribe the coordinate space, the presented results suggest to use homogeneous coordinates for calculating the robot kinematics.

ACKNOWLEDGMENT

This project was carried out within the framework of the course "Topics in Higher Geometry" held by Dr. techn. Georg Nawratil of the Institute of Discrete Mathematics and Geometry, TU Wien.

REFERENCES

[1] K. M. Lynch and F. C. Park, *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, July 2017.

[2] M. Pfurner, "Closed form inverse kinematics solution for a redundant anthropomorphic robot arm," *Computer Aided Geometric Design*, vol. 47, pp. 163–171, Oct. 2016.

[3] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*. Wiley New York, 2006, vol. 3.

Robbie – A tele-operated robot with autonomous capabilities for EnRicH-2019 robotics trial*

S. Emsenhuber, P. Klammer, C. Pöschko, F. Voglsinger, G. Novotny and W. Kubinger[1]

***Abstract*— In public emergencies such as nuclear accidents or natural disasters, an immediate and accurate overview as well as an assessment of the area is the basis of all coordinated plans and actions for the rescue team. The persistent lack of such information leads to high risks and casualties for rescue workers. Mobile robots help to minimize risks and support the rescue teams with urgent information, as well as with debris clearing and search and rescue operations. This work discusses the necessities and requirements of mobile robots in search and rescue (S&R) applications such as an nuclear disaster. Further it describes the current hardware setup as well as the software architecture of the mobile robot *Robbie* of UAS Technikum Wien.**

I. INTRODUCTION

One of the main reasons why deaths follow disasters is that the rescue team takes too long to find the victims because they need to grant their own safety [3]. Rescue robots have been designed for situations like these which are too dangerous for humans, e.g. hostage situations or nuclear disasters [6]. To support developing rescue robots, the European Robotics Hackathon - the EnRicH 2019 - takes place at the nuclear power plant (NPP) Zwentendorf. A nuclear disaster is simulated at the competition. The UAS Technikum Vienna participates in cooperation with the Austrian Armed Forces at the EnRicH 2019.
The competition includes three different tasks:

- **Exploration:** Generate a 3D map of the area and detect and mark radiation sources inside of it.
- **Manipulation:** Identify a specific pipe containing radioactive coolant and close the corresponding valve.
- **Search and rescue:** Locate human dummies inside the building and mark them on the digital map.

In this work the mechanical construction, sensor configuration as well as the implemented software of Robbie are described for the participation at the EnRicH 2019. The following section II outlines the mechanical and software requirements, followed by the implementation results in section III. Finally section IV summarizes this work and gives an overview concerning feature work.

II. SYSTEM CONCEPT

The following section characterizes the requirements regarding the hardware and software architecture of the mobile robot to successfully participate in this search and rescue hackathon.

*This project has been partly funded by MA23 - City of Vienna within the Project Call 16-02 "Photonics: Foundations and industrial applications".

[1]The authors are with UAS Technikum Wien, Hoechstaedtplatz 6, 1200 Vienna; wilfried.kubinger@technikum-wien.at

A. Hardware

Disaster sites usually are covered with rubble, therefore the base platform of the mobile robot needs to be capable of manoeuvring in rough terrain. Further to enable the rescue part an actuator needs to be placed on top of that base platform. The mobile robot platform "*Tracker*" of the company Taurob GmbH was used as the fundamental building stone. As figure 1 visualises the base platform is steered using tank like tracks and a four degree of freedom robotic arm as an actuator. This base platform was on the one hand chosen because of its tank like track steering, since they add stability, can move on rough terrain and allow the robot to climb steps up down [8], and because of its ROS API implementation.

Beside this base platform a number of sensors need to be included to fulfill the tasks of the hackathon. Since the hackathon takes place inside an NPP and therefore in a GPS restricted area depth and ranging sensors, such as RGB-D (RGB-Depth) cameras or LIDAR (Light Detection and Ranging) systems, need to be used for localisation and mapping. Further Inertial Measurement Units (IMUs) can be used to further improve the localisation, using sensor fusion such as Extended Kalman Filters, and to provide input for a tip-over control [1], [5], [7]. Further to measure the radiation and to be able to detect human dummies radiation sensors and thermal imaging cameras need to be equipped on the robot.

To teleoperate the robot a operator station needs to be developed and connected to a processing unit on the mobile robot.

B. Software architecture

To evaluate the previously introduced sensors and to steer the actuators the Robot Operating System (ROS) is used as high-level API. Besides the basic communication with the actuators, sensors and the operator station ROS is used because of it's scaleability and open source infrastructure. Thus with ROS in combination with the Taurob Tracker a basic teleoperated mobile robot is provided to build on top. Therefore making it a perfect fit for R&D applications. To further improve the teleoperation process a graphical user interface plugin for rviz [9], which displays all sensor data and allows for easier teleoperation of the robot arm, needs to be developed. Additionally a radiation measurement and simultaneous localisation and mapping package needs to be developed to map the radiation sources of the environment and mark them inside the generated digital map. The last package that needs to be created, in order to successfully achieve the tasks of the EnRicH, consists of the inverse

kinematics of the robot arm to enable precise calculation of the arm movements.

III. RESULTS

The following section introduces the hardware setup as well as the generated/ used ROS software packages.

The current setup of the mobile robot Robbie is displayed in figure 1. The fundamental building block is, as mentioned above, the unmanned ground vehicle (UGV) *Tracker* [2]. The Velodyne PUCK VLP-16 LIDAR system

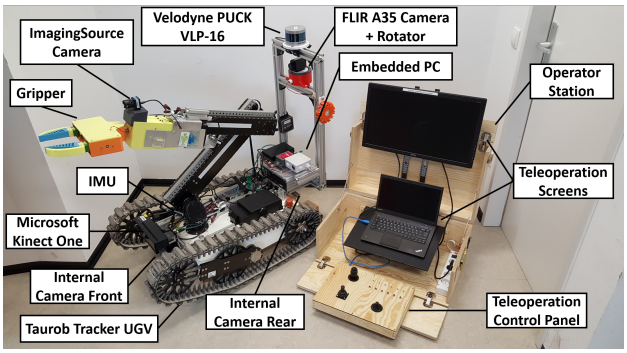


Fig. 1. Robbie hardware setup for EnRich 2019

and a PhidgetSpatial IMU were used in combination with "Google's Cartographer SLAM" [4] to generate a digital map of the environment. The internal cameras as well as the Kinect are used to enable seeing while teleoperating. The ImagingSource camera attached to the gripper is utilized to operate the end effector.

Figure 2 visualises the generated 3D map using the sensor configuration as described above. Further figure 2 visualises

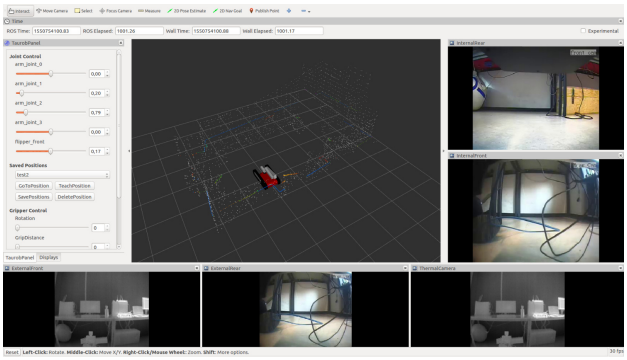


Fig. 2. Generated point cloud example using Cartographer, LIDAR and IMU and the developed rviz GUI

the GUI plugin for rviz with which enables precise movement of the robot arm and flipper joints, via the slide bars, and visualises the video stream of the attached cameras for easier teleoperation.

Finally figure 3 visualises a 2D digital map of the same environment as in the previous picture with marked radiation sources inside of it.

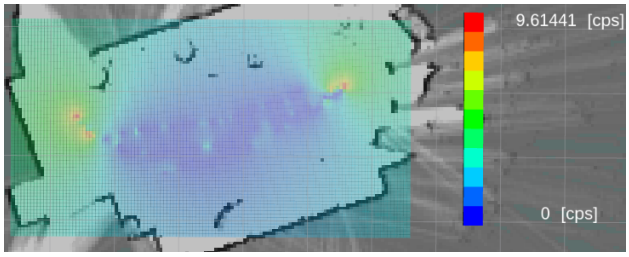


Fig. 3. Generated 2D map with marked radiation sources

IV. SUMMARY AND OUTLOOK

In this paper we described the necessary hardware setup and software architecture of a mobile robot for S&R applications inside an NPP. We utilized "Google's Cartographer SLAM" [4] to build a 3D pointcloud map of the environment using RGB-D, LIDAR and IMU sensors and build a application on top of it to visualise radiation sources inside the projected 2D map. Further a portable operator station was built to enable easier teleoperation and a rviz plugin was developed for visualisation of all sensor data of Robbie. Furthermore, an inverse kinematics application was developed for accurate calculation of the movement of the robot arm's axis.

Further projects will tackle the tasks of Human-detection using the equipped infrared camera (FLIR) as well as calculation of the perfect gripper positioning to drag humans out of hazardous environments. Moreover "*No-Go*" areas need to be defined for the robot arm so that it will not self-collide. Additionally, a tip-over control, using IMU and the robot arm, needs to be implemented so that Robbie will not tip over if the centre of mass and the slope require it.

REFERENCES

[1] B. Choi, G. Park, and Y. Lee, "Practical control of a rescue robot while maneuvering on uneven terrain," *Journal of Mechanical Science and Technology*, vol. 32, no. 5, pp. 2021–2028, May 2018. [Online]. Available: https://doi.org/10.1007/s12206-018-0410-7

[2] T. GmbH. Ugv-taurob-tracker. [Online]. Available: http://taurob.com/de/produkte-2/ugv-taurob-tracker/

[3] S. Grayson, "Search & rescue using multi-robot systems," *School of Computer Science and Informatics, University College Dublin*, 2014.

[4] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d lidar slam," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1271–1278.

[5] K. Khoshelham and S. Zlatanova, "Sensors for indoor mapping and navigation," *Sensors*, vol. 16, no. 5, 2016. [Online]. Available: http://www.mdpi.com/1424-8220/16/5/655

[6] M. N. Kiyani and M. U. M. Khan, "A prototype of search and rescue robot," in *2016 2nd International Conference on Robotics and Artificial Intelligence (ICRAI)*, Nov 2016, pp. 208–213.

[7] G. A. Kumar, A. K. Patil, R. Patil, S. S. Park, and Y. H. Chai, "A lidar and imu integrated indoor navigation system for uavs and its application in real-time pipeline classification," *Sensors*, vol. 17, no. 6, 2017. [Online]. Available: http://www.mdpi.com/1424-8220/17/6/1268

[8] J. Oliveira, L. Farçoni, A. Pinto, R. Lang, I. Silva, and R. Romero, "A review on locomotion systems for robocup rescue league robots," in *RoboCup 2017: Robot World Cup XXI*, H. Akiyama, O. Obst, C. Sammut, and F. Tonidandel, Eds. Cham: Springer International Publishing, 2018, pp. 265–276.

[9] Open Source Robotics Foundation. (2018) rviz. [Online]. Available: http://wiki.ros.org/rviz

General Robot-Camera Synchronization Based on Reprojection Error Minimization

Kenji Koide[1] and Emanuele Menegatti[1]

Abstract— **This paper describes a synchronization method to estimate the time offset between a robot arm and a camera mounted on the robot (i.e., robot-camera synchronization) based on reprojection error minimization. In this method, we detect a calibration pattern (e.g., checkerboard) from camera images while projecting the pattern onto the image space with robot hand poses and forward kinematics. Then, we estimate the delay of the camera data by finding the robot-camera time offset which minimizes the reprojection error between the visually detected and the projected patterns. Since the proposed method does not rely on any camera-specific algorithms, it can be easily applied to any new camera models, such as RGB, infrared, and X-ray cameras, by changing only the projection model. Through experiments on a real system, we confirmed that the proposed method shows a good synchronization accuracy and contributes to the accuracy of a continuous scan data mapping task.**

I. Introduction

Vision based inspection systems are widely considered for industrial applications [1]. To respond to the increasing demand for rapid and efficient production, a lot of automated vision-based inspection systems have been exploited for real systems. However, to our knowledge, most of the existing inspection systems are highly dependent on specific production systems, and their reusability is very limited.

SPIRIT [1], an industrial robot project, aims to develop a "general" inspection robot framework. In this framework, the product to be inspected, the robot, and the camera for inspection can be easily replaced with ones for new inspection tasks. For this purpose, all the components which compose the framework have to be independent of the specific product, robot, and camera model. In particular, the generality with respect to the camera model is important. The framework has to be able to handle various imaging sensors in a unified system, to name a few: RGB-D, thermographic, and X-ray cameras.

Robot-Camera synchronization is one of the essential tasks for visual inspection systems. Usually, there is a delay on images acquired with a camera due to encoding, decoding, and buffered communication [2]. In scenarios where a robot performs a continuous scan motion while mapping images in the robot space, the synchronization accuracy has a significant impact on the final mapping accuracy. If the camera is not synchronized with the robot, we cannot refer the correct robot pose at the moment an image was acquired, and cannot map the image in the correct position in the robot space.

Hardware-based synchronization is the most reliable and accurate way to synchronize two or more devices. Typically, to synchronize data from multiple devices, the data acquisition of each device is triggered by using an external hardware signal [3]. With such a hardware mechanism, data from different devices can be synchronized in the order of 100 nanoseconds [4]. However, hardware synchronization mechanisms require the devices to have a hardware interface and a special data acquisition mode for triggering the data acquisition using the external signal. In case devices (in our case, camera and robot) do not have such synchronization interfaces, we need a software-based method, which uses only data acquired by the devices, to synchronize them.

Several methods for software-based robot-camera synchronization have been proposed [5], [6], [7]. They first detect a calibration pattern using the camera while moving the robot hand where the camera is mounted, and then estimate the camera poses (i.e., camera motion) with respect to the calibration pattern. The time delay between the camera and robot motion sequences is estimated using, for instance, cross-correlation techniques [6]. There are also several techniques to perform spatial and temporal calibration simultaneously [8]. The problem here is that we need to explicitly estimate the camera pose with respect to the calibration pattern for each input image. For pinhole cameras, typically a PnP algorithm [9] is used to estimate the camera pose from an image. However, in the case with a non-pinhole camera, usual PnP algorithms cannot be used to estimate the camera pose. For instance, with some special camera models (e.g., X-ray source-detector camera model), it is not possible to estimate the camera pose from a single image.

The idea to use the reprojection error term for robot-camera calibration is introduced by [10]. They estimate the robot-camera transformation by minimizing the reprojection error. Since this method does not require to explicitly estimate the camera pose with respect to the pattern for each image, it does not rely on any camera model-specific algorithms, and it can be applied to different camera models by changing the projection model. It has been shown that the reprojection error minimization-based method works well on both pinhole and source-detector camera models on real systems with RGB and X-ray cameras. Following [10], in this work, we introduce a reprojection error minimization-based method to achieve a general robot-camera synchronization method.

In this paper, we propose a general robot-camera synchro-

*This work was supported by EU Horizon 2020 grant No. 779431, the SPIRIT project.

[1]Kenji Koide and Emanuele Menegatti are with the Department of Information Engineering, University of Padova, via Gradenigo 6/B, 35131, Padova, Italy, koide@dei.unipd.it

[1]http://spirit-h2020.eu/

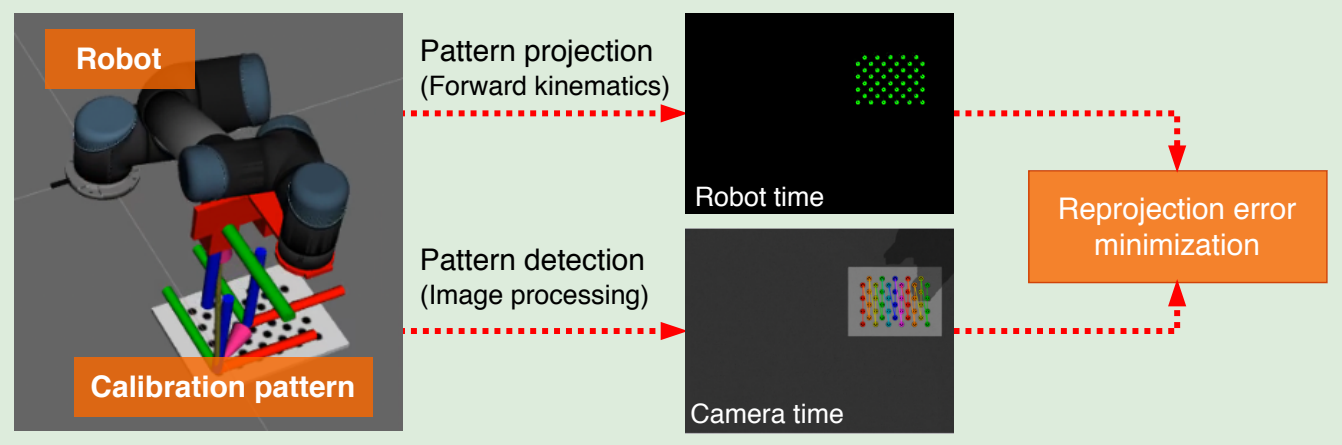


Fig. 1: The proposed robot-camera synchronization method.

nization method based on reprojection error minimization (see Fig. 1). We first collect a sequence of robot hand poses and a sequence of images of a calibration pattern (e.g., checkerboard). Then, we synchronize the robot and the camera by finding the time offset which minimizes the reprojection error between the visually observed calibration patterns and the patterns reprojected with the robot hand poses and forward kinematics. Since the proposed method does not rely on any camera-specific algorithms, like PnP algorithm, it can be applied to any camera models by changing only the projection model.

II. Methodology

Fig. 1 shows an overview of the proposed method. We estimate the time offset Δt by comparing sequences of non-synchronized robot hand poses and camera images. To compare them, we first move the robot hand along a certain path (e.g., sine curve above the calibration pattern), and record a sequence of robot hand poses and a sequence of images of the calibration pattern. Let $\mathcal{R}$ be the robot hand pose sequence, and $\mathcal{I}$ be the calibration pattern image sequence. Let us assume that we collected N images and K hand poses, and the calibration pattern consists of M points.

Let $\hat{\boldsymbol{p}}_t^i$ be the i-th point of the calibration pattern detected from the j-th image with timestamp $\hat{t}$ (in the camera time). Given a robot-camera time offset Δt, we calculate the robot hand pose $\boldsymbol{R}_t$ at the corresponding robot time $t = \hat{t} + \Delta t$ by interpolating the discrete robot hand poses $\mathcal{R}$. We use Slerp (Spherical linear interpolation) [11] to interpolate the robot hand poses:

$$\boldsymbol{R}_t = Slerp(\mathcal{R}, t). \tag{1}$$

With the interpolated robot hand pose $\boldsymbol{R}_t$, we project each point of the calibration pattern $\boldsymbol{p}^i$ into the camera space.

$$\boldsymbol{p}_t^i = Proj(\boldsymbol{R}_t, \boldsymbol{p}^i), \tag{2}$$

where, $\boldsymbol{p}_t^i$ is the projected i-th point of the pattern at robot time t. We can use any projection function suitable to describe the camera. For instance, we use a pinhole camera model for RGB cameras [12], and a source-detector camera model for X-ray cameras [13].

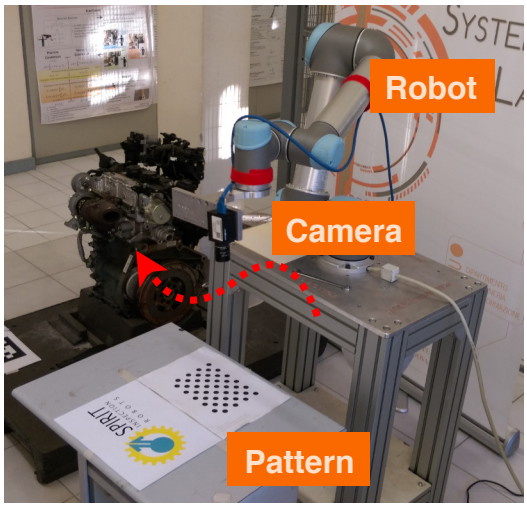


Fig. 2: A snapshot of the synchronization experiment.

The reprojection error is defined by:

$$E(\Delta t) = \sum_{\hat{t}}^{N} \sum_{i}^{M} \|\hat{p}_t^i - Proj(Slerp(\mathcal{R}, \hat{t} + \Delta t), \boldsymbol{p}^i)\|. \tag{3}$$

We estimate the robot-camera time offset $\Delta\tilde{t}$ which minimizes the reprojection error:

$$\Delta\tilde{t} = \arg\min_{\Delta t} E(\Delta t). \tag{4}$$

In this work, we find the optimal time offset $\Delta\tilde{t}$ using exhaustive search in the range $[-0.2s, 0.2s]$. Since the reprojection error shows a good convexity as shown in Fig. 3, this search process can be improved with a line search method [14].

III. Experiments

A. Robot camera synchronization

To validate the proposed method, we conducted an experiment on a real system. Fig. 2 shows a snapshot of the

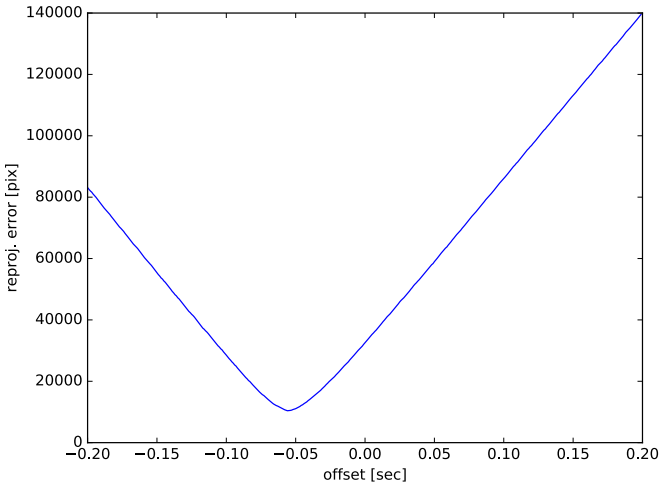


Fig. 3: The reprojection error with different time offset Δt. The reprojection error is minimized at $\Delta t = -0.056$ [sec].

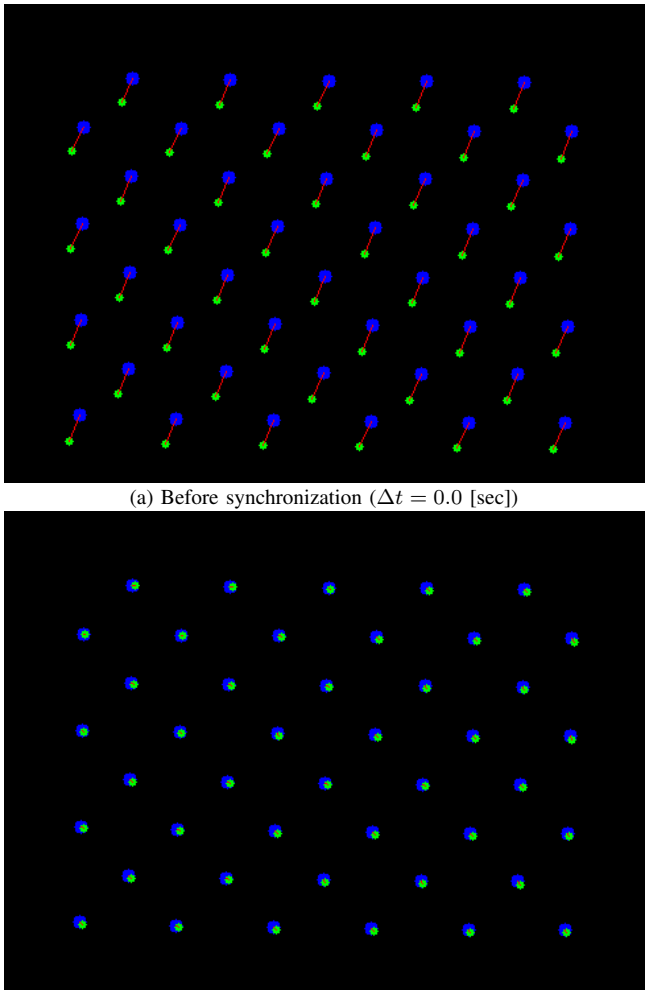
experimental environment. We mounted a camera (Pointgrey Flea3) on a robot arm (Universal robot UR5), and placed a calibration pattern so that the camera can see it. The robot-camera transformation and the calibration pattern pose are estimated with an automatic calibration technique [15]. We moved the robot along a sine curve above the calibration pattern (width=0.4[m], height=0.3[m], altitude=0.5[m]), and recorded images and robot hand poses while the robot was moving. We fed these sequences of images and robot hand poses to the proposed method and estimated the time offset between the camera time and the robot time.

Fig. 3 shows the plot of the reprojection errors E versus the time offset Δt. We can see that the reprojection error is minimized at the point $\Delta t = -0.056$ [sec]. This means that the images (camera time) are delayed from the robot hand poses (robot time) 0.056 [sec], and by adding this offset to the camera timestamp, we can obtain the corresponding timestamp in the robot time.

Fig. 4 shows the reprojected and the visually observed patterns at the frame when the robot was moving at the maximum speed ($\sin(\text{phase} = \pi)$, about 0.25 [m/sec]). The green points are the reprojected pattern with the robot hand pose and forward kinematics, and the blue points are the pattern visually detected from the image. Before synchronization, the pattern reprojected based on the robot hand are obviously precede the visually detected pattern. With the estimated time offset, the reprojected pattern is well synchronized with the visually detected pattern, and we observe very small reprojection errors between them.

B. Continuous scan data mapping

To demonstrate that the camera and the robot are well synchronized with the proposed method, we conducted a continuous scan data mapping experiment. With the same setting as Sec. III-A, we recorded images and robot hand poses. We estimate the plane of the calibration pattern from the very first frame of the images, and then project all the images acquired with the continuous robot motion onto the



(a) Before synchronization ($\Delta t = 0.0$ [sec])

(b) After synchronization ($\Delta t = -0.056$ [sec])

Fig. 4: The visually observed (blue) and the projected (green) points before and after the synchronization. The robot was moving at about 0.25 [m/sec]

calibration pattern plane with the robot hand poses (without any image processing, such as image stitching and alignment). Fig. 6 shows the images projected and accumulated on the calibration pattern plane. 47 images are accumulated in total. Without synchronization, the positions of the projected images deviate due to the delay of the camera images, and as a result, the accumulation image is blurred. With the estimated time offset, the images are synchronized with the robot hand poses and precisely projected on the plane, and we observe a clear accumulation image with the continuous scan.

IV. Conclusions

This paper proposed a robot-camera synchronization method which estimates the robot-camera time offset by minimizing the reprojection error. Since the proposed method does not rely on any camera-specific algorithms, it can be applied to any imaging sensors by changing only the projection model. The experimental results show that the proposed method can be applied to real systems, and it

(a) (b) (c) (d) (e)

Fig. 5: The image sequence recorded while the robot was drawing a sine curve above the calibration pattern.

(a) Before synchronization ($\Delta t = 0.0$ [sec]) (b) After synchronization ($\Delta t = -0.056$ [sec])

Fig. 6: The continuous scan data mapping results. 47 images are accumulated in total.

contributes to the accuracy of continuous scan data mapping tasks.

We are planning to apply the proposed method to real X-ray imaging systems to show that the method can be applied to new imaging sensors with a small modification of the projection model. We are also planning to conduct further assessment of the synchronization accuracy, and how it affects the continuous scan data mapping accuracy.

ACKNOWLEDGMENT

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 779431.

REFERENCES

[1] J. Ghosh, *Vision based inspection*. Springer Netherlands, 1994.

[2] "Latency in live network video surveillance," Axis communications, Tech. Rep., 2015. [Online]. Available: https://www.axis.com/files/whitepaper/wp_latency_live_netvid_63380_external_en_1504_lo.pdf

[3] A. Vibeck, "Synchronization of a multi camera system," 2015.

[4] P. Loschmidt, R. Exel, A. Nagy, and G. Gaderer, "Limits of synchronization accuracy using hardware support in IEEE 1588," in *IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication*. IEEE, sep 2008.

[5] D. Rosen, L. Carlone, A. Bandeira, and J. Leonard, "A certifiably correct algorithm for synchronization over the special Euclidean group," in *Intl. Workshop on the Algorithmic Foundations of Robotics*, San Francisco, CA, Dec. 2016.

[6] K. Pachtrachai, F. Vasconcelos, G. Dwyer, V. Pawar, S. Hailes, and D. Stoyanov, "CHESS—calibrating the hand-eye matrix with screw constraints and synchronization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2000–2007, jul 2018.

[7] F. Furrer, M. Fehr, T. Novkovic, H. Sommer, I. Gilitschenski, and R. Siegwart, "Evaluation of combined time-offset estimation and hand-eye calibration on robotic datasets," in *Field and Service Robotics*. Springer, 2018, pp. 145–159.

[8] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, nov 2013.

[9] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Epnp: An accurate o(n) solution to the pnp problem," *International Journal of Computer Vision*, vol. 81, no. 2, pp. 155–166, 2009.

[10] K. Koide and E. Menegatti, "General hand–eye calibration based on reprojection error minimization," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1021–1028, apr 2019.

[11] K. Shoemake, "Animating rotation with quaternion curves," 1985.

[12] D. A. Forsyth and J. Ponce, "A modern approach," *Computer vision: a modern approach*, vol. 17, pp. 21–48, 2003.

[13] M. Mitschke and N. Navab, "Recovering the x-ray projection geometry for three-dimensional tomographic reconstruction with additional sensors: Attached camera versus external navigation system," *Medical Image Analysis*, vol. 7, no. 1, pp. 65–78, 2003.

[14] M. J. B. D. D. W. H. Swann, *"Non-linear optimization techniques"*. Oliver & Boyd,, 1969.

[15] M. Antonello, A. Gobbi, S. Michieletto, S. Ghidoni, and E. Menegatti, "A fully automatic hand-eye calibration system," in *European Conference on Mobile Robots*. IEEE, 2017, pp. 1–6.

Control of Autonomous Mobile Robot using Voice Command

Trung Quoc Nguyen[1], Peter Nauth[2] and Sudeep Sharan[3]

Abstract— **Controlling the machine by voice or speech has always aroused the curiosity of humans. After many pieces of research and developments, the voice recognition system becomes an important and comfortable system to communicate with machines in day today's life. In this paper, a voice control software system is created and integrated with the mapping algorithm available in Robotic Operating Systems (ROS) and implemented with a mobile robot Volksbot [Figure 1]. This paper also expresses the development of a Graphical User Interface (GUI) with different tabs to control the system in different ways, first by clicking voice command to navigate the robot to its destination in the available map, and second by typing the command in words or numbers. If the commands are mistaken, then the user can abort the commands by clicking stop tabs. In order to test the voice system accuracy, the experiment is performed with different voices as well as with different pitches. This work also shows the results of the accuracy of reaching the destination's room in the map.**

I. INTRODUCTION

Service robots have recently become an important part of human society. The number of robots is increasing rapidly in the market. The more user-friendly robot execution becomes, the more easily people can control it. One easy way to control a robot is by means of voice. Human often finds it more suitable to give order by saying something rather than typing the code. Voice communication between robots and users plays a critical part in any applications [1,2]. The user can command the robot to execute the task by speaking, even if the user has no knowledge of machines and computers.

In this article, a voice recognition system is integrated with a mapping algorithm which is implemented on an intelligent mobile robot. Using ROS firmware [3], the system controls the robot to move to different locations.

Section 2 introduces the system structure of a robot control system. Section 3 describes the voice recognition. Section 4 expresses the system implementation of the voice control system on a mobile robot including the mapping algorithm integrated using ROS. The experimental results and testing are discussed in section 5, followed by the conclusion in section 6.

II. DESIGN OF VOICE CONTROL SYSTEM

The mobile robot system is controlled by two computer systems. One is attached to the mobile system and the other is set in a fixed place and used as a virtual system. Both computers are connected to each other via WLAN. Static virtual system (Computer 1) is connected to a microphone, which receives the voice signals from the user and transfers them to the robot system computer (Computer 2). The voice recognition detects the signal here and then the robot executes the action according to the given voice command.

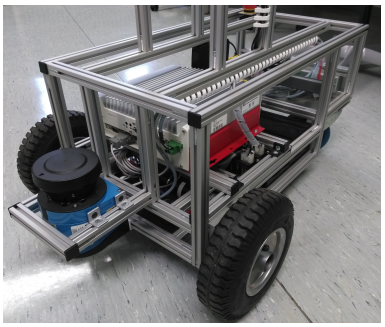


Fig. 1. Mobile Robot "Volksbot"

The mobile system is a Volksbot developed by Fraunhofer Institute. It has two front actuated wheels and two back castor-wheels. The encoders are also connected with the wheel motors to achieve odometry data. A Lenovo Thinkpad laptop is placed on the volksbot which is responsible for all processes of computation. The laser range finder LMS100 is mounted in front of the volksbot, which scans the entire surrounding environment and provides information about the environment, which helps to create a map of an unknown environment. The operating range and angle of view are 0.5m to 20m and 270 degrees respectively. The sensor is used because of its high update rate of 50 Hz. A 3D Kinect-camera is connected to the volksbot in order to generate a 3D cloud map of the environment. Kinect Xbox 360 is used because of its high adaptation and availability. The Kinect camera is connected to the laptop placed on the mobile system and a microphone is connected to a static computer on the desktop (Computer 1), which communicates with the laptop (Computer 2) on the mobile robot via secure shell (SSH). The structure of the mobile voice control system is shown in Figure 2.

The voice control robot system consists of different modules, such as automatic speech recognition (ASR) module and the control module. ASR system analyzes the user's voice and then transfers the command to the robot to execute the user's request. The control module understands the voice command and performs the action accordingly. The process system is expressed in Figure 3.

[1]Trung Quoc Nguyen is with Faculty of Computer Science and Engineering, Frankfurt University of Applied Sciences, Nibelungenplatz 1, Frankfurt am Main, Germany trungquo@stud.fra-uas.de

[2]Peter Nauth is with Faculty of Computer Science and Engineering, Frankfurt University of Applied Sciences, Nibelungenplatz 1, Frankfurt am Main, Germany pnauth@fb2.fra-uas.de

[3]Sudeep Sharan is with Faculty of Computer Science and Engineering, Frankfurt University of Applied Sciences, Nibelungenplatz 1, Frankfurt am Main, Germany s.sharan@fb2.fra-uas.de

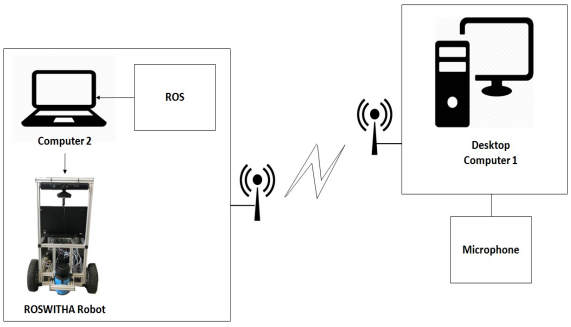


Fig. 2. Hardware connection of the mobile control system

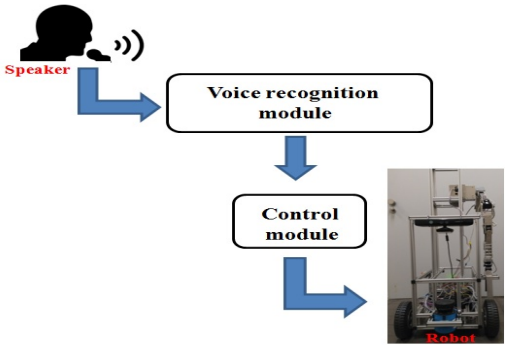


Fig. 3. Command control diagram

The system of the voice command is complicated. When a user commands to the robot, the voice is sent from the microphone to the speech recognition module. Then the module tries to detect the user's voice by extracting features from it. Voice features are analyzed using a trained database, which is given to the system by the user. At the end of the speech recognition process, a text version of the user's command is generated.

The captured commands will be transferred to the control module. Then the control module executes the commands as it gets from the speech recognition module and makes the mobile system move accordingly.

There are few basic trained sentences in the database as follows:
GO TO ROOM TWO ZERO THREE
GO TO ROOM TWO ZERO FOUR
GO TO ROOM TWO ZERO FIVE
GO TO ROOM TWO ZERO SIX
GO TO AUTONOMOUS LAB (Home location)

If the command of the user is any of the above sentences, then the robot executes the command and moves towards that room. The extension of the commands can enable the robot to move to multiple rooms if there are multiple commands such as "GO TO ROOM TWO ZERO THREE then GO TO ROOM TWO ZERO FIVE".

To navigate in an unknown environment, the robot has to create a map and navigate to the proper path to reach the destination. The robot should also have the ability to avoid object. The SLAM algorithm RTAB-MAP [4], which is already available in ROS as a package, is implemented on the mobile system to enable the robot to draw the map. With the help of the navigation stack node, the robot navigates towards the destination by planning the path and avoiding the obstacles in the path. All the names of destination rooms on the map are trained in the voice control module.

III. VOICE RECOGNITION

In order to control the autonomous mobile robot by voice, it is necessary to employ a speech recognition system inside the robot. In this article, the CMU Sphinx recognition system is used. It is a leading toolkit for voice detection which has been developed for a long time at Carnegie Mellon University. CMU Sphinx contains many powerful packages which are used for different situations and environments. In this work, Pocketsphinx, Sphinxbase, and Sphinxtrain are used to develop the voice recognition system.

Pocketsphinx is a lightweight version specialized for mobile devices. Other versions of CMU Sphinx such as Sphinx-II and Sphinx-III are complex, require many memories and often take considerable time to translate the speech, which is favorably used for the static system. On the other hand, Pocketsphinx is aimed for portability, simplicity of implementation and memory efficiency, which is suitable to be installed on the mobile robot. The decoder architecture, which is explained in [5], consists of three consecutive search strategies. Each latter search strategy uses the result of the previous and narrows the search space. Sphinxbase is a support library accompanied by Pocketsphinx. Sphinxtrain is needed during the processing of adapting and training acoustic models, which will be discussed later.

In our system, the process of designing a speech recognition system is divided into four parts:

- Build a dictionary
- Build a language model
- Adapt an existing acoustic model
- Provide decoder models in the previous three steps to start the detection processing

A dictionary contains all words that the robot needs to detect, together with the corresponding sequences of phonemes. An acoustic model describes how likely an acoustic realization is given that the text is known. Finally, a language model illustrates the likelihood of the next word in the sequence when the previous words have been detected [5,6].

IV. SYSTEM IMPLEMENTATION

The implementation of the system is structured in ROS and integrated with the mobile system as shown in figure 4.

The system is structured in ROS by using different packages for controlling hardware including *freenect_launch* driver for the Kinect camera, *lms1xx* for the laser scanner and *volksbot_driver* for driving the volksbot. SLAM algorithm RTAB-Map is implemented in the system, which is available in ROS as *rtabmap*. The RTAB-Map receives the data from odometry, laser scanner and Kinect camera to create a 3D

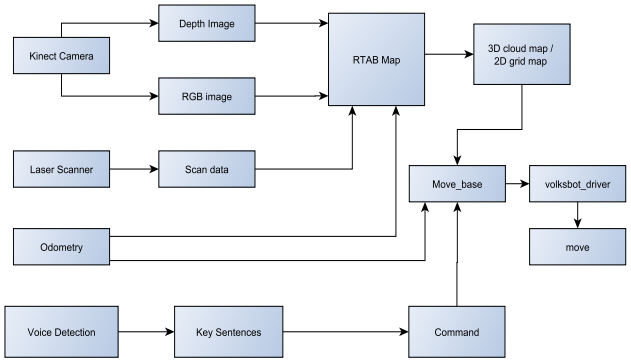


Fig. 4. ROS structure for building autonomous robot system

cloud map/2D grid map of an unknown environment. The 2D grid map is sent to the *move_base* [5] node. The information from the *move_base* node is transferred to the mobile robot so that it navigates according to the user's command.

A. Building voice recognition module

The recognition module is constructed by first building the dictionary, the language model and the acoustic model, then initializing the language decoder with these configurations. The dictionary should include all the words that the system is going to detect, otherwise it cannot detect correctly. First, a corpus file, which is a list of needed sentences, was created. All sentence commands should be included in this file. Next, from sentences in corpus file every word is extracted and stored in another file, together with their corresponding sequence of phonemes. This is the main dictionary file, which should have extension of *.dict*.

During the recognition process, the decoder looks in the dictionary to find the best sequence of phonemes that matches with the acoustic properties of the voice and picks the corresponding words in the dictionary for the next steps. However, providing a dictionary is not enough. A language model must also be generated, which specifies the probability of the next word given existing word. There are three ways to build a language model, which is explained in more details in [6]. The first way is using a keyword list. This list is just like a corpus file, with each sentence linking with a threshold. Long sentences should have a bigger threshold than short sentences. The second way is to use grammar. The third way, also the most complex way, is using a statistical language model. It enables users to use natural language rather than just simple command.

The statistical language model is a file in which probabilities of words and word combination are calculated. These probabilities illustrate how often a phrase or sentence is said by a user. Different words and phrases have different probabilities. The determination of probabilities is based on the input corpus file. One of the largest advantages of statistical language model is that it can detect other combinations than just ones in the corpus. For example, in corpus file, two sentences 'GO TO ROOM TWO ZERO SIX' and 'GO TO ROOM TWO ZERO FIVE' are listed. They have the largest probabilities in the language model. However, probabilities of other sentence combinations still exist, such as GO TO ROOM FIVE ZERO SIX. The user is able to speak more freely. The robot is therefore friendlier to human.

There are three common forms of language model: text ARPA format, a binary BIN format, and a binary DMP format [6]. A text ARPA file has the extension *.lm*. Although it takes more space, it is editable. On the other hand, a binary BIN file is a compressed version of ARPA file, with the extension of *.lm.bin*. It takes less space and loads faster. The last form is obsolete.

With the task of building an autonomous mobile robot, the data set is small. Therefore, it is possible to use CMU Tool directly on the website LMTool page of CMU by providing a corpus file. With corpus file of large size or with language other than English, it can be done by SRI Language Modeling Toolkit (SRILM) or CMUCLMTK [6]

Finally, an acoustic model is needed to complete the configuration of the system. Normally, a standard acoustic model such as CMUS Sphinx model for US English is suitable for dealing with a simple recognition task, since it has been developed and optimized for many years. However, in some special cases where users speak with slightly different accents, such as UK English or Indian English, adapting the standard model to user's voice may improve the performance greatly.

The prerequisite for building an acoustic model is an existing corpus file of all necessary sentences. In this step, each sentence is accompanied by a unique ID. These IDs are then stored in an ID file.

A corresponding dictionary for all words in the corpus file is also used to adapt acoustic model. In order to adapt the standard acoustic model, a recording of user speaking each of those sentences is given to the system. Each sentence is recorded independently at sampling rate of 16 kHz in mono with a single channel and stored in a single audio file *.wav*, which is named according to the ID of each sentence. The sound recorder Audacity is used in this project for recording and editing audio files. The third step is extracting acoustic features of audio files for later adapting process. This is done using the Sphinxbase[6]. Each audio generates a feature file with the extension $.mfc$. These feature files, together with standard acoustic model, dictionary, corpus file and ID files are exploited to collect statistics by using the Sphinxtrain command *bw*[6].

In the last step, Maximum Likelihood Linear Regression (MLLR) transform is deployed. Its suitability with a small dataset and online adaptation makes it a good adaptation method for the autonomous system[6]. The result of adapting process is a file called $mllr\_matrix$, which is passed to the decoder in Algorithm 1.

B. Graphical user interface of the system

The graphical user interface (GUI) is responsible for interacting with the entire system. Using GUI, users neither

Algorithm 1: DETECTING SPEECH

```
Result: A Phrase is detected
begin
    initialize configuration for recognition process by
      choosing dictionary, standard acoustic model and
      language model and mllr_matrix;;
    initialize decoder with existing configuration;
    using Pyaudio to set up portaudio system;
    open a stream on input device to record audio;
    start the stream;
    set the decoder to start decoding utterance;
    while true do
        read frames from stream and store it in buffer;
        if buffer not None then
            detect word from buffer and store it; if last
              feed audio buffer does not contained
              speech then
                stop decoding utterance;
                output the hypothesis phrase that is
                  detected;
```

have to type complex command to control the robot, nor they have to understand the mechanism behind the system. Figure 5 shows the window of a GUI.

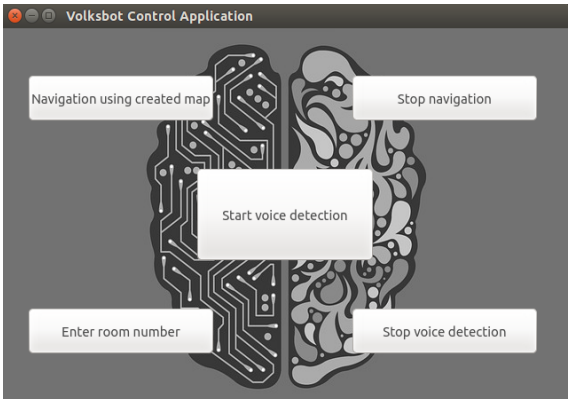


Fig. 5. Main window of a graphical user interface

In the GUI, five tabs are created to control the robot system in different aspects:

- Navigation using a created map: choose a map from the database and make the system ready to navigate in that environment.
- Stop navigation: stop all commands and make the system stop at any point.
- Start voice detection: initialize the voice detection function of the system and make the speaker ready to speak.
- Enter room number: The robot can be controlled by typing numbers, if the robot fails to recognize the voice.
- Stop voice detection: stop the voice command, if the speaker speaks the wrong command.

V. EXPERIMENT & TESTING

In the real experiment and testing, when the user speaks a word or sentence to the microphone, the system checks the voice with the trained database and if the voice matches the robot will execute according to the command.

In this work, the voice commands in the system are trained to go to the autonomous lab, go to room 203, go to room 204, go to room 205, go to room 206. In the real scenario, after the user gives the instruction through the microphone, the signal goes to the voice recognition module and then processes to the robot control system. The entire experiment was performed in an environment of the second floor, Building 8 of Frankfurt University of Applied Sciences, Germany. First, the speech was recorded and saved in a database, then the database is generated as explained in section III. Once the voice command is detected, the speech command system recognizes that voice and provides a control signal to the control system of the robot and makes the robot perform the action according to the command received. The robot is moving in real time during the speech recognition process.

A. Testing

1) Testing voice recognition: The test of voice recognition was performed in two parts, the first test was to check the accuracy of detection of trained sentences, where the different commands were spoken, which told the robot to go to different destinations. Each command was spoken 10 times. The result, which was detected sentences, was shown on the computer screen. The word error rate of each detected sentence is calculated. It states that, if both original text and recognition text have a length of N words, I be the number of inserted words, D be the number of deleted words and S be the number of substituted words. The word error rate (WER) is: $WER = (I+D+S)/N$. The second test was performed by making three different people to command the robot to execute the action and this test was performed 5 times for each person and investigate the error rate of the voice detection. The test results of the trained sentences are shown in Table I and the test results of the voice recognition are shown in Table II

Sentences trained	Error rate
GO TO AUTONOMOUS LAB	1%
GO TO ROOM TWO ZERO SIX	0%
GO TO ROOM TWO ZERO FIVE	0%
GO TO ROOM TWO ZERO FOUR	1%
GO TO ROOM TWO ZERO THREE	0%

TABLE I

ERROR RATES OF TRAINED SENTENCES

B. Experimental Results

The experiment was done on the volksbot mobile platform of the robot ROSWITHA (**RO**bot **S**ystem **WITH** **A**utonomy), a self constructed assistive robot system under the roof of Frankfurt University of Applied Sciences, Germany. The robot navigated to different room numbers

Commands	Trials	Speaker 1	Speaker 2	Speaker 3
GO TO AUTONOMOUS LAB	5	5	5	5
GO TO ROOM TWO ZERO SIX	5	5	4	5
GO TO ROOM TWO ZERO FIVE	5	5	5	5
GO TO ROOM TWO ZERO FOUR	5	5	5	5
GO TO ROOM TWO ZERO THREE	5	5	5	5

TABLE II

ERROR RATES OF RECOGNIZING DIFFERENT VOICES

according to the commands. The experiment results are expressed in Figure 6.

Fig. 6. Experiment Results

Finally it is analysed that the trained sentences and the different voice recognition accuracy are almost 100%, which shows that the voice recognition system is robust.

VI. CONCLUSIONS

In this paper, a voice control system is implemented in a mobile robot. The voice system shows better results and recognizes the speech of a user and performs the action according to the user's commands. The system is proved to be efficient in a real-time scenario. This implemented system with GUI shows the potential for being used in a voice related application in automation like HMI (Human Machine Interface). Furthermore, the system is found to be effective in understanding different people's voice commands to execute the actions and navigate to its destination.

ACKNOWLEDGMENT

This work has been funded by the Frankfurt University of Applied Sciences, Germany. The authors would like to thank Mr. Umansky, Mr. Michalik and other colleagues of the Faculty of Computer Science and Engineering, Frankfurt University of Applied Sciences for their guidance and valuable support throughout the project.

REFERENCES

[1] Prof. Dr. Subhash P. Rasal (2014) Voice Controlled Robotic Vehicle, International Journal of New Trends in Electronics and Communication (IJNTECISSN: 2347 - 7334), vol. 2, no. 1, pp. 28-30, 2014.
[2] Peter X. Liu, A.D.C. Chan, R. Chen, K. Wang, Y. Zhu, Voice Based Robot Control,Proceedings of the 2005 IEEE International Conference on Information Acquisition June 27 - July 3, 2005, Hong Kong and Macau, China.
[3] Robotic operating Systems [Available online] http://wiki.ros.org/
[4] RTAB-Map: http://wiki.ros.org/rtabmap
[5] David Huggins Daines, An Architecture for Scalable, Universal Speech Recognition, PhD dissertation (chapter 3), Carnegie Mellon University, 2011.
[6] CMUSphinx, Basic concepts of speech recognition. [Online Available: https://cmusphinx.github.io/wiki/tutorial/]. [Accessed: January 2019]
[7] Navigation Stack: http://wiki.ros.org/move_base

Kinematics of a Human Steering a Car

Michael Schwegel[1] and Jakob Maderthaner[1]

***Abstract*— This paper presents an analysis of the kinematic manipulability of the human arm while steering a car. The human arm is modeled as 7-axis robot and a specialized measure of manipulability for the problem at hand is introduced. The analysis of different steering scenarios shows that optimal manipulability yields handling scenarios that are intuitive to a human operator. Furthermore the shoulder joint position is optimized to find the optimal seat position.**

I. INTRODUCTION

There are various approaches for steering a car, including different grips and angles at which the hand is in contact with the steering wheel. Since the human arm can be modeled as a 7-axis serial manipulator [1] its kinematics can be easily found using the Denavit-Hartenberg (DH) convention. The analysis of the manipulability of such a kinematic model shows that intuitive approaches of steering a car are similar to those found by optimizing a specialized measure of manipulability.

II. MATHEMATICAL MODELING

The process of steering was modeled using a serial manipulator with 7 rotational joints. The spherical joint at the shoulder was modeled by 3 intersecting rotational joints. Considering only the shoulder position the torso can be neglected. Further, two symmetric arms are assumed. Thus, the model contains 8 coordinate frames attached to the body, see Table I for the DH-parameters. Frame 1 is located in the

n	ϑ_n	d_n	r_n	α_n
1	0	0	0	$\pi/2$
2	$q_1-\pi/2$	0.15m	0	$\pi/2$
3	$q_2+\pi/2$	0	0	$\pi/2$
4	$q_3+\pi/2$	0.31m	0	$\pi/2$
5	q_4	0	0	$-\pi/2$
6	$q_5-\pi/2$	0.19m	0	$-\pi/2$
7	$q_6-\pi/2$	0	0	$-\pi/2$
8	q_7	0	0.05m	0

TABLE I
DH PARAMETERS OF THE MATHEMATICAL MODEL.

shoulder and is assumed to be inertially fixed. The frames 1 to 7 and q_i, $i \in [1;7]$ are the joint angles of the arm, with $q_{1,2,3}$ for the shoulder, q_4 for the elbow and $q_{5,6,7}$ for the wrist. Frame 8 is used to describe the offset of the contact point on the steering wheel relative to the wrist. Furthermore, d_2 is half the distance between the shoulders, d_4 is the length of the upper arm, d_6 the length of the lower arm and r_8 is the distance from the wrist to the contact point of the hand with the steering wheel.

The initial position of the seat relative to the steering wheel was chosen such that the shoulder is rotated $q_1 = 45°$ forwards from the vertical position and the elbow joint is $q_4 = 105°$. Additionally, the steering wheel was tilted by 20°. For the desired pose of the hand on the steering wheel different scenarios of steering are investigated. Each scenario is defined by desired positions and orientations of the contact point between the hand and the steering wheel during a steering maneuver. Figure 1 shows the links of the arm

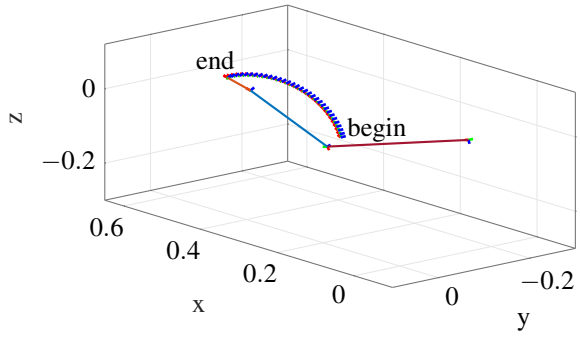


Fig. 1. Kinematic model for steering a car. Modeled arm with shoulder at (0,0,-0.15). end effector coordinate frames are traced along the steering wheel for a 90° turn.

model and the trace of the end effector frame of a steering maneuver.

III. OPTIMIZATION

For simulation, the position and orientation of the end effector frame are given as a reference pose. In order to solve the inverse kinematics problem an optimization was used. Compared to an analytical solution this allows for a relative weighting of the hand orientation and position and yields realistic movements of the hand. The joint angles of the model are then used to analyze the manipulability of the scenario. The test case is a rotation of the steering wheel of 90°.

For scenario 1 (s1) the right hand touches the steering wheel at the right side. The orientation is given by the back of the hand pointing radially outwards and the hand pointing perpendicular to the steering wheel. Scenario 2 (s2) is given by the driver grabbing the steering wheel at the top from below and pulling it downwards. Thus, the wrist points radially outwards and the back of the hand is orthogonal to the steering wheel center.

The end effector position p and the wrist orientation

*This work was not supported by any organization

[1]Michael Schwegel, Jakob Maderthaner are with TU Wien, {schwegel,maderthaner}@tuwien.ac.at

quaternion o are used in the cost function

$$J(q) = \sum_{k=1}^{N} (f(q_k) - p_{k,ref})^{\mathrm{T}} (f(q_k) - p_{k,ref}) + \quad (1)$$
$$(o_{v,k} - o_{v,k,ref})^{\mathrm{T}} W_o (o_{v,k} - o_{v,k,ref}) +$$
$$(q_{k+1} - q_k)^{\mathrm{T}} W_{\Delta q} (q_{k+1} - q_k) +$$
$$q_k^{\mathrm{T}} W_q q_k \,,$$

with k being the discrete time index, $f(q_k)$ the position of the hand, and $o_{v,k}$ the vector part of its orientation quaternion. The constant diagonal weighting matrices W_o, $W_{\Delta q}$, and W_q determine the relative influence of the individual terms. This allows for the orientation to be less restrictive in the solution of the inverse kinematics problem, thus resulting in natural movements of the wrist.

IV. Manipulability

The resulting joint angles of the hand model are used to analyze the manipulability of the different steering scenarios. Given the joint angular velocities $\dot{q}$ the end effector linear velocities $\dot{p}$ can be calculated according to

$$\dot{p} = J\dot{q} \,, \quad (2)$$

with the Jacobian matrix $J \in \mathbb{R}^{3\times 7}$. Note, that for steering the translation of the end effector is the deciding quantity for turning the wheels, thus the orientation is omitted in the following. The manipulability is given by

$$m = \sqrt{\det(JJ^{\mathrm{T}})} \,, \quad (3)$$

see, [2]. Turning the steering wheel is achieved by moving the hand in the direction tangent to the steering wheel at the current contact point. This steering movement is actuated by joint velocities which can be mapped to the steering wheel using the Jacobian matrix. Thus, the *directional manipulability*

$$m_{\mathrm{dir}} = \|t^{\mathrm{T}} J\| \quad (4)$$

is introduced, which is the projection of the position part of the Jacobian matrix (and thus the manipulability ellipsoid) onto the unit tangent vector to the steering wheel t. This measure indicates the possibility to move the end effector in the direction t only.

These measures for a 90° turn using scenario 1 (s1) and 2 (s2) are depicted in Figure 2.

V. Shoulder Joint Position

In order to calculate the optimal position of the driver seat the optimization problem is solved for different shoulder joint positions. To design an indicator for the optimal seat position, the manipulability m is integrated along the steering angle φ, i.e.

$$M = \int_0^{90^\circ} m(\varphi) \mathrm{d}\varphi \,. \quad (5)$$

The classical manipulability has a distinct optimal shoulder position since both straight and lateral movements are considered, see Figure 3. In contrast, the directional manipulability suggests that the driver should be seated far away

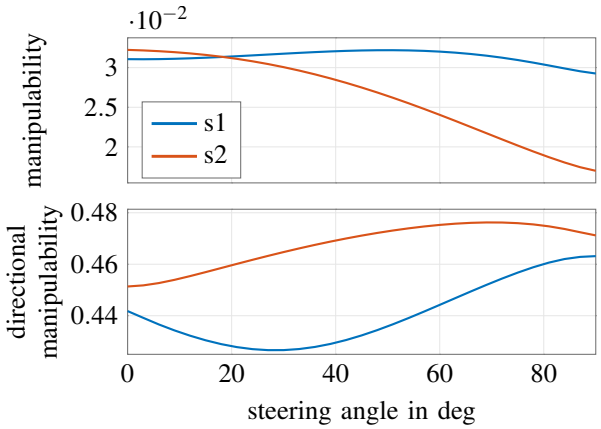


Fig. 2. Manipulability of scenario 1 and 2.

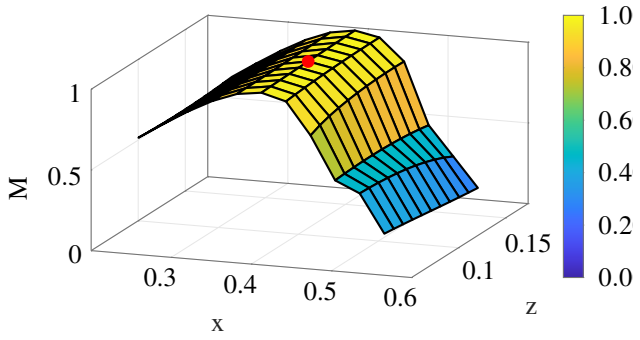


Fig. 3. Manipulability for scenario 1. The red dot marks the manipulability for the nominal values.

from the steering wheel. The z coordinate of the seat has only little influence on the manipulability. This can be reasoned with the longer distance causing higher velocities for a given joint angular velocity.

VI. Conclusions

A kinematic model of a human arm was used to analyze the steering of a car. In this context a specialized manipulability measure was introduced to evaluate the tangent velocity acting on the steering wheel. Two manipulability measures were used to optimize the position of the seat in the car. This spacially discretized optimization showed that the vertical position has only little influence on the manipulability. The presented algorithm proved to be a viable approach for optimizing the driver seat position in a car. Furthermore, it can be extended to satisfy specific requirements by using problem specific optimality criteria.

Acknowledgment

This paper was supervised in the course of "Topics in Higher Geometry" by Georg Nawratil of the Institute of Discrete Mathematics and Geometry, TU Vienna.

References

[1] E. V. Biryukova, A. Roby-Brami, A. A. Frolov, and M. Mokhtari, "Kinematics of human arm recunstrutced from spatial tracking system recordings," *Journal of Biomechanics*, 2000.

[2] T. Yoshikawa, *Foundations of Robotics: Analysis and Control*. Cambridge, MA, USA: MIT Press, 1990.

Evaluation of Human-Robot Collaboration Using Gaze based Situation Awareness in Real-time*

Lucas Paletta[1], Martin Pszeida[1], Harald Ganster[1], Ferdinand Fuhrmann[1], Wolfgang Weiss[1], Stefan Ladstätter[1], Amir Dini[1], Bernhard Reiterer[2], Inka Brijacak[2], Guido Breitenhuber[2], Sandra Murg[1], and Harald Mayer[1]

***Abstract*— Human attention processes play a major role in the optimization of human-robot collaboration (HRC) systems. This work describes a novel framework to assess the human factors state of the operator primarily by gaze and in real-time. The objective is to derive parameters that determine information about situation awareness which represents a central concept in the evaluation of interaction strategies in collaboration. The control of attention provides measures of human executive functions that enable to characterize key features in the collaboration domain. Comprehensive experiments on HRC were conducted with typical tasks including collaborative pick-and-place in a lab based prototypical manufacturing environment. The methodology measures executive functions and situation awareness (SART) in the HRC task in real-time for human factors based performance optimization in HRC applications.**

I. INTRODUCTION

Collaborative robotics has recently progressed to human-robot interaction in real manufacturing. Human factors are crucial as industrial robots are enabling human and robot workers to work side by side as collaborators and to assess the user's experience with a robot, while understanding how humans feel during their interaction with it [1]. Furthermore, human-related variables are essential for the evaluation of human-interaction metrics [2]. To work seamlessly and efficiently with their human counterparts, robots must similarly rely on measurements to predict the human worker's behavior, cognitive and affective state, task specific actions and intent to plan their actions. A typical application is anticipatory control with human-in-the-loop architecture [3] to enable robots to proactively perform task actions based on observed gaze patterns to anticipate actions of their human partners according to its predictions. However, measuring and modeling of the state of human factors as well as the human situation awareness based on gaze triggered information recovery is mandatory for the understanding of immediate and delayed action planning.

This work describes a novel methodology to measure the human factors state of the operator in real-time with the purpose to derive fundamental parameters that determine situation awareness as a central concept in the interaction strategies of collaborative teams. Human situation awareness is determined on the basis of concrete measures of eye movements towards production relevant processes that need to be observed and evaluated by the human. Motivated by the theoretical work of [4] on situation awareness the presented work specifically aims at dynamically estimating (i) distribution of attentional resources with respect to task relevant 'areas of interaction' over time, determined by features of 3D gaze analysis and a precise optical tracking system, and (ii) derive from this human factors in real-time, such as, (a) human concentration on a given task, (b) human mental workload, (c) situation awareness and (d) executive functions related measure, i.e., task switching rate.

Gaze in the context of collaboration is analyzed in terms of - primarily, visual - affordances for collaboration. In this work we stress the relevance of considering eye movement features for a profound characterization of the state of human factors by means of gaze behavior, with the purpose to optimize the overall human-robot collaboration performance.

Figure 1: Human-robot collaboration and intuitive interface (HoloLens, eye tracking, markers for OptiTrack localization) for the assessment of human factors state to characterize key features in the collaboration domain.

[1]DIGITAL – Institute for Information and Communication Technologies, JOANNEUM RESEARCH, Graz, Austria.

[2]ROBOTICS – Institute for Robotics and Mechatronics, JOANNEUM RESEARCH, Klagenfurt, Austria.

*This work has been supported by the Austrian Ministry for Transport, Innovation and Technology (BMVIT) within project CollRob (Collaborative Robotics), FLEXIFF (no. 861264) and MMASSIST (no. 858623). Lucas Paletta is corresponding author with phone: +43-316-876-1769; fax: +43-316-876-91769; e-mail: lucas.paletta@joanneum.at

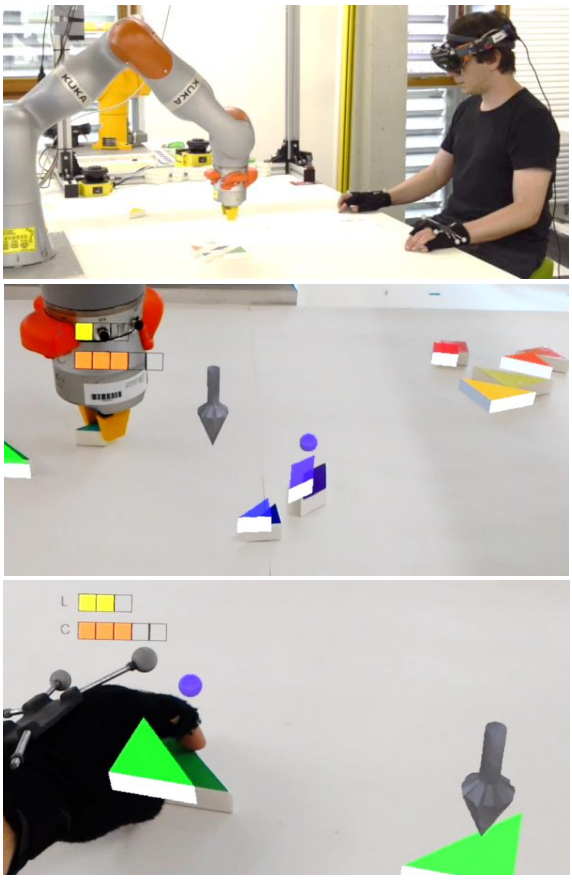


Figure 2: HRC within a tangram puzzle assembly task. (a) The operator collaborates with the robot in the assembly (only robot can treat 'dangerous' pieces). (b) Egocentric operator view with augmented reality based navigation (arrow), piece localization, gaze (blue sphere), current state of mental load (L) and concentration (C) in the HoloLens display. (c) Recommended piece (arrow) and gaze on currently grabbed puzzle piece.

The estimation of situation awareness of the human worker can be crucial for the elaboration of performance analysis through measurement of executive functions, evaluation of interruption impact, as well as for the prediction of accidents.

II. Related Work

A. Human-Robot Collaboration

Human-Robot collaboration has substantially advanced, in the planning domain. [4] presented a planning executive able to handle choices by either team member while respecting causal links and temporal constraints with regard to risk bounds. In a related approach, one of the contributions of [5] defines helpfulness in terms of cost reduction resulting from the utilization of the robot. [6] split the planning process for repetitive collaborative assembly tasks into two phases, 1st is offline and establishes agents' capabilities for hierarchical plans with choices, the second one taking cost-function-based optimization decisions. [7] researched probabilistic planning for collaborative manipulation, aiming at boosting the human's trust in the robot's capabilities for better overall performance. [8] employ a probabilistic approach focused on dynamical switching and on decision-making in hierarchical assembly tasks. The aspect of making the behavior of a planning system better understandable for humans is addressed by Fox et al. [9] who pose a set of questions that the system should be able to answer. They show how to capitalize on features of planning systems that would make similar questions harder or impossible to answer based on other, currently popular AI-based decision making approaches. In this context, the cognitive state of the human gets into the focus of research studies.

B. Evaluation of Human-Robot Collaboration

Kragic et al. [10] presented results of a study featuring a human-robot interaction task using three different feedback modalities: a computer screen, projection into the workspace, and augmented reality. While projection was subjectively higher rated, the study did not yield significant performance differences between the variations. Salem et at. [11] report on an analysis of how various kinds of faulty behavior by a robot affect humans' trust. Measures for Human Robot Collaboration are of three complementary dimensions: team performance measures, measures targeting user satisfaction and experience, and safety and trust related assessments. In this line, the Huang et al. [12] evaluated different handover strategies using both objective measures (Task Completion Time (TCT), Concurrent Activity (CA), Human and Robot Idle Time (HIT and RIT) as well as subjective scales via questionnaires (Fluency, Intelligence, Awareness and Patience)

The human response to robot movements was assessed using objective measurements (TCT, CA, HIT, RIT, as well as average separation distance), and subjective criteria for perceived safety and comfort via questionnaires. Moreover, [3] evaluated the anticipatory control of a robot in user studies. Here, anticipation was derived from gaze analysis, the evaluation measures in the study included the average robot response time and the TCT. Next, [11] used a vast amount of questionnaires to evaluate the influence of robots' mistakes in the quality of the Human Robot Cooperation and Trust. Applied subjective test include the Ten Item Personality Inventory (TIPI), Godspeed Questionnaire, Human Nature Scale, and Uniquely Human Scale. Finally, [13] examines trust, interaction and safety issues of industrial workers on fenceless human robot collaboration. Here, individual questionnaires for specific interaction mechanisms (e.g. voice, gestures), and compared to objective measures such as average robot response time. Safety was regarded the most important factor for successful HRC, followed by usability and efficiency.

III. Human-Robot Collaboration Prototype

In the presented work, we apply a human factors analysis system to a complete human-robot collaboration system which is described as follows in more detail.

System Architecture

The architecture for the here-presented overall system was designed as a 3-tier application. Figure 3 shows the hierarchical system decomposition of the three layers. The hierarchy is based on task abstraction, i.e., a task at one level is achieved by

invoking a set of tasks at lower levels. For connection and communication (i.e. services and messages) between the components we used the Robot Operating System (ROS).

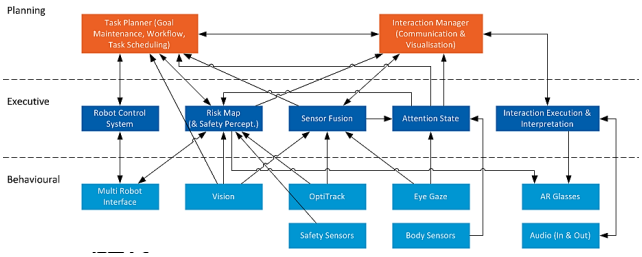


Figure 3: System architecture diagram depicting the 3-tier.

B. *Task Planner & State Description*

In order to deal with the manifold situations that may arise in a collaborative robot application, the orchestration of the robot and interaction components is performed by a task planning subsystem. It has a PDDL 2.1 [14] planner wrapped in ROSPlan [15] at its core, performant enough to be invoked after any significant change in order to always act based on current knowledge. The planning problems in our custom domain contain a mix of abstract knowledge derived from configuration, such as which agents there are and in which zones of the workspace they are allowed to act, and structures to be built at certain locations, from sensor data with respect to configuration, such as visibility and current zone of parts, and from combining the previous two with task-related aspects, such as whether a part counts as added to a structure being built. Such enriched object knowledge is also passed on to the interaction components. Our knowledge provider component keeps the knowledge about the current state and goals up to date according to the various data sources of the system. It takes care that are currently feasible and of top priority, i.e., requests by the participant to pass an object override assembly goals. The action link implementation bridges the gap between high-level plans and the robot control interface. It is able to work around robot joint limit violations and collisions on the table, also considering shape-dependent equivalent rotations of parts.

Based on preliminary experiments leading up to the study, the planning subsystem was configured such that the robot keeps working in its dedicated area as much as possible and only enters the collaborative area when necessary. This principle leads to more opportunities for the human participant to safely contribute to the joint task. Including the human participant's and/or the referee's actions in the planning problem was considered in different variations and leads to promising plans in isolated tests. However, this improvement could not be integrated due to accumulating delays resulting from a temporal lack of flexibility in available planning executive configurations. We aspire to follow up on this in future work incorporating findings from [4].

C. *Computer Vision & Safe Robot Control*

For locating parts to be manipulated on the table, we use a combination of shape and color matching approaches on 2D RGB images from a single camera, the output of which is rectified and projected into the table plane in the robot's 3D workspace based on results of intrinsic camera calibration and hand-eye calibration. As a key element of our multifaceted safety infrastructure, we track the participant's head and hands with a motion capturing system and compute distances to the nearest robot parts, applying worst-case assumptions in case of bad marker visibility. Based on a combination of this evaluation and the participant's stress level, the robot's current movement speed is limited to a configurable degree, down to a full stop. Further safety measures include low overall speed, spatial limits to the robot's workspace, sufficient distribution of emergency stop buttons to the participant, referee and extra personnels.

D. *Study Goals*

Within the described user study we aim at evaluating the quality of the interaction via the here-presented interaction system within a human-robot-collaboration application. In this context, we want to assess the individual interaction components (i.e. speech, gestures, etc.) as well as the overall interaction mechanisms resulting from the interconnection of the different modalities. We particularly examine to which degree the additional interaction mechanisms provided by the interface enhance of the human-robot-interaction. Here we focus on the user perspective on the developed interaction system and emphasize the intuitiveness of the interaction. We thereby assess the targeted qualities in a multi-dimensional way applying objective and subjective measures. To compare the generated results to a baseline system we performed the same experimental procedure within two distinct evaluation settings. On the one hand, a reference system (NUL) is established without interaction assistance. The full interaction functionalities are available in a second evaluation (INT).

IV. Intuitive Multimodal Assistive Interface

Assessing the intuitiveness and performance of an interface that should be regarded as intuitive as possible appears to be an obvious objective for a human factors analysis component. In the following we present a novel intuitive interface for multimodal assistance in human-robot collaboration. This interface will serve as a testbed for the efficiency of elderly.

A. *Multimodal Interaction Design*

Conceptually, we positioned the interaction design towards the user. For intuitive interaction we opted for a human-centered approach and started from inter-human interactions and the collaborative process itself. Following these considerations we implemented an interaction system that will be described in the remainder of this section.

The here-presented interaction system is based on the following principles:

- *Natural interaction*: Mimicking human interaction mechanisms we guarantee fast and intuitive interaction processes.
- *Multi-modal interaction:* We implement speech, gaze, gestural, and Mixed-Reality interaction to offer as much interaction freedom as possible to the user.

- *Tied modalities:* We link the different interaction modalities to emphasize the intuitive interaction mechanisms.
- *Context-aware feedback:* Feedback channels deliver information regarding task, environment to the user. We pay attention at what is delivered when and where.

Figure 4 shows a schematic overview of the presented interaction system. A central component entitled 'Interaction Model' (IM) acts as interaction control and undertakes the communication with the periphery system. The IM also links the four interaction modalities and ensures information exchange between the components. It triggers any form of interaction process, both direct and indirect, and controls the context-sensitivity of the feedback. It is further responsible for dialog management and information dispatching.

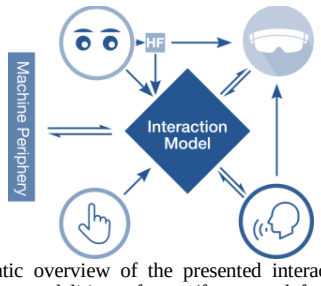


Figure 4: Schematic overview of the presented interaction system. The included interaction modalities refer to (from top left clock-wise): Gaze (including Human Factors (HF)), Mixed-Reality, Speech, and Gestures.

B. Audio Communication

Speech is the most natural communication element and acts as the main connecting element between the modalities in the developed interaction system. It is mostly used to derive the main intention of the interaction, which is strengthened by other modalities. Here, the system applies an acoustic interface to receive speech input. Context awareness is fully guaranteed since the interface is initialized and configured by the IM. It moreover delivers information about the speech interaction process (i.e. voice activity and last recognized command) to the Mixed-Reality display to increase the usability and user experience of the interaction.

Speech interaction is based on state-of-the-art Automatic Speech Recognition (ASR). We use the VoCon[1] library from Nuance, audio input is done via a Bluetooth headset. The system is set up in a constant-listening mode without a wake-up-word, i.e. the user has direct access to the voice commands without having to press a push-to-talk button. Hence, depending on the context (controlled by the IM), a various amount of speech commands is always available at a given moment. Moreover, we use a state-of-the-art speech synthesis engine to sonify the robot speech. We used the system offered by Acapella[2].

C. HoloLens Display (Augmented Reality)

Microsoft HoloLens[3] offers a state-of-the-art mixed reality (MR) development environment. By using an MR based display the system is able to augment the visual environment. We use annotations to visually mark real objects and give real-time feedback regarding the gaze and speech interaction process (e.g. gaze pointer and voice activity). We further provide indicators using visual icons to inform the user about changes in the task and the environment. Moreover, we use the interactive functionalities of the display to enable dialog interaction with virtual object selection via hand gestures. Hence, dialogs triggered by the IM can be resolved either using the Mixed-Reality display or the speech interface.

D. Human State Description and Human-in the-loop

In order to describe the current state of the human operator, a human factors measurement system (Sec. V) is integrated into the interaction system (Figure 4). It is fundamentally based on eye tracking for pervasive measurements of human cognitive and mental state.

V. Human Factors Measurement System

In human factors and ergonomics research, the analysis of eye movements enables to develop methods for investigating human operators' cognitive strategies and for reasoning about individual cognitive states [16]. Situation awareness (SA) is a measure of an individual's knowledge and understanding of the current and expected future states of a situation. Eye tracking provides an unobtrusive measure to measure SA in environments where multiple tasks need to be controlled. [17] provided first evidence that fixation duration on relevant objects and balanced allocation of attention increases SA. However, for the assessment of executive functions, the extension of situation analysis towards concrete measures of distribution of attention is necessary and described as follows.

A. Recovery of 3D Gaze in Human-Robot Interaction

Localization of human gaze is essential for the localization of situation awareness with reference to relevant processes in the working cell. [18] Firstly proposed 3D information recovery of human gaze with monocular eye tracking and triangulation of 2D gaze positions of subsequent key frames within the scene video of the eye tracking system. Santner et al. [19] proposed gaze estimation in 3D space and achieved accuracies ≈ 1 cm with RGB-D based position tracking within a predefined 3D model of the environment. In order to achieve the highest level of gaze estimation accuracy in a research study, it is crucial to track user's frustum / gaze behavior with respect to the worker's relevant environment. Solutions that realize this include vision-based motion capturing systems: OptiTrack[4] can achieve high tracking and gaze estimation accuracy (≈ 0.06 mm).

[1] https://www.nuance.com/mobile/speech-recognition-solutions/vocon-hybrid.html
[2] http://www.acapela-group.com
[3] https://www.microsoft.com/de-at/hololens
[4] http://www.naturalpoint.com/optitrack

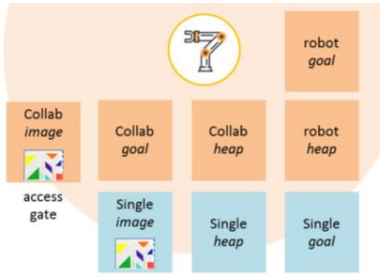


Figure 5: Schematic representation (top view) of the areas of interaction (AOIs) that are used to analyze gaze behavior and fro this derive human factors analysis. Collaborative task related AOIs are in orange, Single task related in blue color.

B. *Situation Awareness*

Based on the cognitive ability, flexibility and knowledge of human beings on the one hand and the power, efficiency and persistence of industrial robots on the other hand, collaboration between both elements is absolutely essential for flexible and dynamic systems like manufacturing [20]. Efficient human-robot collaboration requires a comprehensive perception of essential parts of the working environment of both sides. Human decision making is a substantial component of collaborative robotics under dynamic environment conditions, such as, within a working cell. Situation awareness and human factors are crucial, in particular, to identify decisive parts of task execution.

In human factors, situation awareness is principally evaluated through questionnaires, such as, the Situational Awareness Rating Technique (SART, [21]). Psychological studies on situation awareness are drawn in several application areas, such as, in air traffic control, driver attention analysis, or military operations. Due to the disadvantages of the questionnaire technologies of SART and SAGAT, more reliable and less invasive technologies were required, however, eye tracking as a psycho-physiologically based, quantifiable and objective measurement technology has been proven to be effective [17][22] In several studies in the frame of situation awareness, eye movement features, such as dwell and fixation time, were found to be correlated with various measures of performance. [23] have developed measurement / prediction of Situation Awareness in Human-Robot Interaction based on a Framework of Probabilistic Attention, and real-time eye tracking parameters.

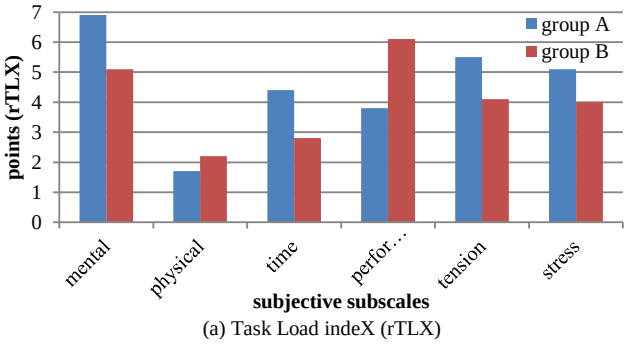
C. *Stress and Concentration Estimation*

For stress quantification we used cognitive arousal estimation based on biosensor data. In the context of eye movement analysis, arousal is defined by a specific parametrization of fixations and saccadic events within a time window of five seconds so that there is good correlation (r=.493) between the mean level of electrodermal activity (EDA) and the outcome of the stress level estimator [25].

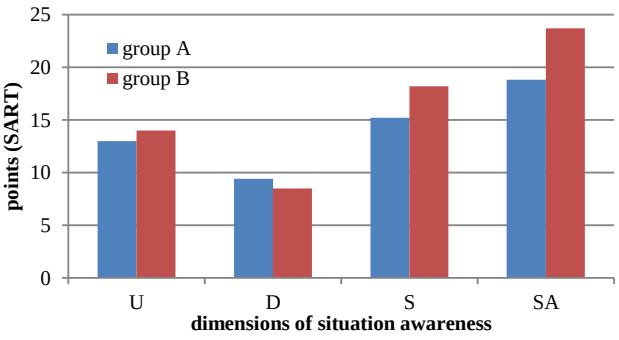
For the estimation of concentration or sustained attention, we refer to the areas of interaction (AOI) in the environment as representing the spatial reference for the task under investigation. Maintaining the attention on task related AOI is interpreted as the concentration on a specific task [26], or on session related tasks in general. Various densities of the fixation rate enable the definition of a classification of levels of actual concentration within a specific period of time, i.e., within a time window of five seconds.

D. *Estimation of Task Switching Rate*

Task switching, or set-shifting, is an executive function that involves the ability to unconsciously shift attention between one task and another. In contrast, cognitive shifting is a very similar executive function, but it involves conscious (not unconscious) change in attention. Together, these two functions are subcategories of the broader cognitive flexibility concept. Task switching allows a person to rapidly and efficiently adapt to different situations [27].



(a) Task Load indeX (rTLX)



(b) SART

Figure 6: Results of the questionnaires from after the session without (group A) & with assistance (group B), respectively. (a) The rTLX subjective subscales show significant reduction of mental workload in group B. (b) The SART results in significant increase in U (understanding) and S (support of attention) and decrease in D (attentional demand) for group B.

In a multi-tasking environment, cognitive resources must be shared or shifted between the multiple tasks. Task switching, or set-shifting, is an executive function that involves the ability to unconsciously shift attention between one task and another. Task switching allows a person to rapidly and efficiently adapt to different situations. The task-switching rate is defined by the frequency by which different tasks are actually operated. The difference between tasks is defined by the differences in the mental model which is necessary to represent an object or a process in the mind of the human operator. Mental models are subjective functional models; a task switch requires the change

of the current mental model in consciousness and from this requires specific cognitive resources and a load.

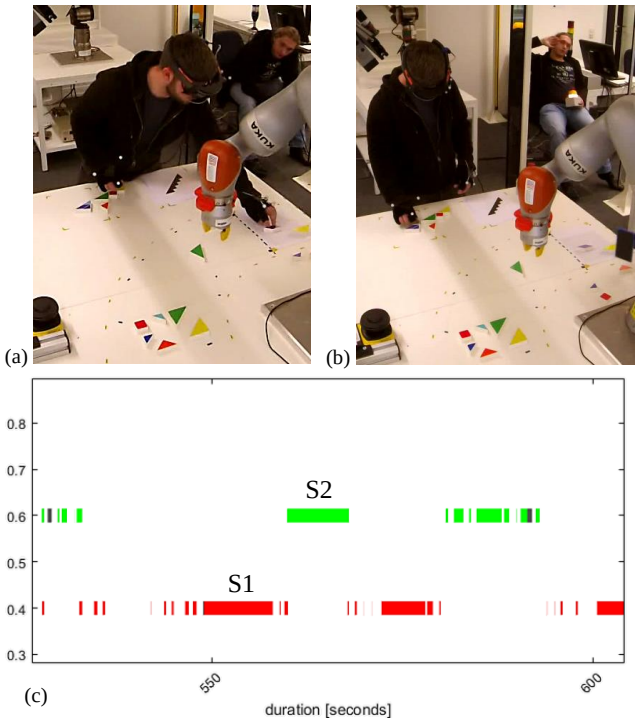


Figure 7: Task switching between collaborative and the single task. (a) Placing a puzzle piece to goal area in the collaborative task. (b) The operator places a puzzle piece to the goal area of single task.(c) Switch between collaborative task (S1) and single task (S2). Task duration is determined by the human gaze being focused within an AOI related to the specific task.

In the presented work, processing of a task is determined by the concentration of the operator on a task related area of interaction (AOI). Interaction is defined by areas in the operating environment where the operator is manipulating the location of puzzle objects, i.e., grabbing puzzle pieces from a heap of pieces, or putting pieces onto a final position in order to form a tangram shape. Whenever the gaze of the operator intersects with an AOI that belongs to a specific task, then it is associated with an on-going task. The task switch rate is then the number of switches between tasks per period of time, typically the time of a whole session (see Figure 7 for a visualization). Task switching has been proposed as a candidate executive function along with inhibition, the maintenance and updating of information in working memory, and the ability to perform two tasks at the same time. There is some evidence not only that the efficiency of executive functions improves with practice and guidance, but also that this improvement can transfer to novel contexts. There are demonstrable practice-related improvements in switching performance [29][30].

VI. Experimental Results

A. Study Setup

The experimental procedure was performed in a robotics laboratory. 20 persons (f=8, m=25) aged 25.4±4.7 years were engaged, mostly with education at university level (BSc). 8 of them were with technical or even scientific background, but all naïve to the task. All participants were introduced into the tasks which consisted of assembling 8-piece tangram puzzles according to target images. The overall task was to finalize three puzzles in collaboration with a robotic arm that performed pick and place actions according to a plan. Tt was possible to assemble in parallel a difficult 'extra' puzzle without any external support. The session was finalized upon the finalization of the 3rd collaborative puzzle, and performance was measured by completion time minus a bonus time for a single finalized 'extra' puzzle. Participants were told that the best 3 performances would earn special gifts (for competitive pace)

Robotic system. The robot arm used was a Kuka iiwa LBR 7 R800, equipped with a Zimmer R800 2-finger gripper prototype and customized 3D-printed fingers with metal reinforcement and soft cover. The RGB camera used for vision was a Basler acA2440-20gc with an f8mm F1.4 lens. It was connected via Gigabit PoE to the computer for vision and safety perception applications (Fujitsu Celsius, Intel Xeon 72-core 2.3GHz, 128GB RAM, Nvidia Quadro P4000 GPU). The motion capturing PC worked with Intel dual core i7 3.6GHz CPU, 16GB RAM and Nvidia Geforce GT 630 2GB, receiving data from 9 OptiTrack Prime 17W cameras.

Wearable system. For the intuitive assistance device we used a Microsoft Hololens AR headset with Pupil Labs binocular eye tracking with 200 Hz eye tracking cameras and USB connector clip that connected to a Microsoft Surface Pro 6 with 8GB/128GB RAM/SSD for the gaze analysis.

B. Outcome Measures

The major outcome parameters of the study were on the one hand related to standardized questionnaires that are commonly used to specify important human factors in the human-machine interaction domain. At the same time, several eye movement features were measured in real-time in order to derive human factors on-site directly from the human-robot collaboration task. The objective of the study was to investigate if on-site measured data would correlate with the results of the questionnaires.

The subjects were asked to complete several questionnaires during the experimental procedure. Specifically, the standardized questionnaires raw Task Load indeX (rTLX) and the Situation Awareness Rating Technique (SART) were used. Furthermore, we designed individual questionnaires to cover general interaction, mixed reality interaction and visualization, speech interaction, and human-robot-collaboration. These include user input via scale ratings as well as open questions relating to good and bad experiences within the respective area. The real-time assessment of the interaction was built upon a mixture of recorded data. The HMD video was captured as well as data from an external video camera for post study analysis. Eye tracking data were collected in real-time and analyzed with respect to the measures presented in Sec. V: the human gaze was positioned in 3D space and intersections with various areas of interaction (AOI) analyzed with respect to level of concentration; all data were recorded in a further study analysis.

C. Descriptive Statistics

The overall performance comparison resulted in a total completion time of M=709.7, SD=193.4 sec. without and M=714.6, SD=159.5 sec. with assistance, respectively, and ANOVA analysis resulted in $F_{crit} > F$ which identifies identical distributions. The bonus addendum was negligible. From the data of twenty participants, only fifteen were used for analysis, for reasons of data loss and loss of eye tracking calibration during the session when persons accidentally changed the orientation of eye tracking cameras when tipping on the helmet. However, the analysis of the questionnaires indicated substantial and interesting differences in the outcomes. Figure 6a depicts subjective subscales of rTLX indicating that users with the intuitive assistance system (group B) subjectively experienced *significantly reduced* levels of *mental workload*, time pressure, tension and stress, respectively. The results of the SART questionnaire (Figure 6b) depict in U (*understanding*) and S (*support of attention*) dimensions with *significant increase* and decrease in D (attentional demand) for group B.

Furthermore, the gaze based outcome measures of the experimental study showed further evidence for interesting differences with respect to without and with intuitive assistance, as documented in Table I: Firstly, the mean *concentration level* ('M(C)') with the use of an intuitive assistance technology was *significantly above the one without assistance*. Figure 8 depicts sample tracks of attentional concentration on tasks during sessions.

In addition, results showed a *significant increase* in the *situation awareness* in terms of attentional attributed to the collaborative AOIs ('Collab'), as well as to the single task ('Single') related AOIs. This clearly shows that the assistance mode successfully *supports the channeling of attentional resources on the task related interaction* areas.

The execution function related feature based on eye movements, i.e., the expected time for a next task switch ('ExpTS'), significantly decreased in case of intuitive assistance. Figure 7a,b shows task switching between the collaborative and the dual, single task, where the operator places a puzzle piece to the goal tangram area as contribution to the collaborative task or to the goal tangram area of the single task. Figure 7c depicts the switches between the collaborative task (S1) and the single task (S2) over time. Substantial switch times are crucial in order to enable meaningful task completion.

The results clearly show an *increased task-switching ability* of the *human operator* while the stress level ('M(S)') factor was not increased (ANOVA shows $F_{crit} > F$ comparing the two distributions of S with and without assistance). Performance on task switching, a paradigm commonly used to measure executive function, has been shown to improve with practice and as a consequence of knowledge [30] .

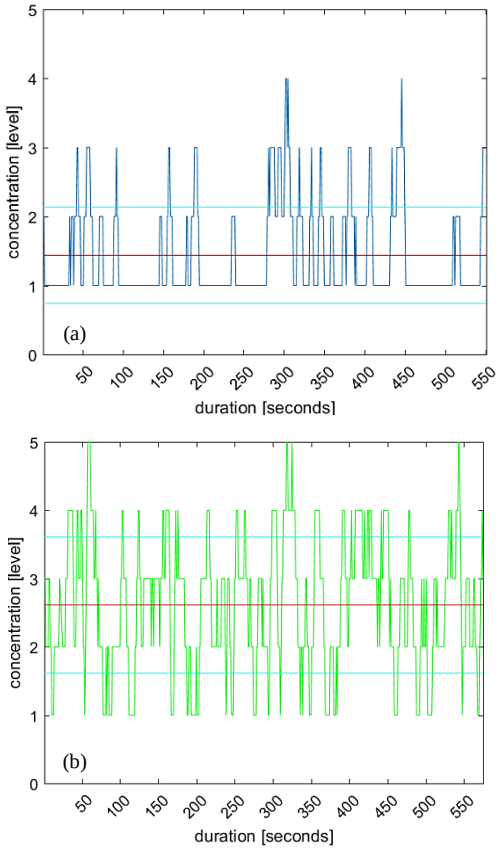


Figure 8: Measure of attentional concentration on tasks. (a) Concentration level during a session without assistance (red line is the mean; red is standard deviation), and (b) concentration level during the session with assistance. On average, the concentration increased when using the intuitive assistance.

D. Inferential Statistics and Discussion

An extensive correlation analysis was performed between real-time gaze based human factors analysis and the questionnaire results. The first important observation is the substantial correlation (Pearson r=-0.404) between the expected task switching time ('ExpTS') and the concentration level ('M(C)') which expresses the fact that an *increased concentration level* at the same time means an *increase in the task switching ability* (shorter expected durations between task switches with the same performance). The second observation is related to SART based situation awareness: M=63.0, SD=10.7% without and M=71.2, SD=16.1% with intuitive assistance, showing significant benefit in applying the assisting technology. *SART correlates positively with attentional resources on collaborative* AOIs (r=0.311 with 'Collab') and negatively on single tasks (r=-0.605 with 'Single').

Discussion. The conclusion with respect to the inferential statistics is that the assessment derived from real-time gaze based human factors analysis is capable to fully quantify the distribution of attentional resources on task relevant space, in real-time, and in this manner not only correlates but represents

the performance of executive functions, i.e., the task switching ability in a human-robot collaborative scenario. This will enable in the future on-site measured attentional resources to represent standard questionnaires on situation awareness which provides a basis to evaluate basic human factors, such as, concentration of attention, complexity of the situation, familiarity of the situation, focusing of attention, information quantity, information quality, instability of the situation, variability of the situation, arousal, and spare mental capacity.

TABLE I. GAZE BASED OUTCOME MEASURES OF THE EXPERIMENTAL STUDY WITH RESPECT TO WITHOUT AND WITH INTUITIVE ASSISTANCE.

user groups	**Outcome measures**					
	M (C)	*M (S)*	*Collab**	*Single*	*ExpTS*	*AcGa*
group A	1.40±0.3	1.35±0.3	23.5	16.1	2.03±1.2	51.15
group B	1.94±0.5	1.52±0.4	125.0	29.8	1.60±0.6	23.7

*$x10^{-3}$. *C=concentration level; S=stress level; Collab=mean attentional time on mixed areas in seconds filtered by concentration, Single= mean time on single (human operator specific) areas in seconds filtered by concentration; ExpTS=expected time for next task switch; AcGa=mean time access gate observation per session.*

VII. CONCLUSION

We presented a novel methodology for the assessment of gaze based human factors which provides a potential to measure executive functions performance, such as, task switching ability, in real-time. Within a typical human-robot collaboration scenario and the study setup including absence/application of state-of-the-art intuitive assistance technology for the performance of collaborative tasks, we illustrated the potential of interpretation from gaze based human factors data in order to evaluate the MRC system.

Future work will include a study with a larger number of participants and evaluate the executive function measures under more competitive pressure for performance. Furthermore we will study the potential of more complex eye movement features to enable a more detailed analysis of the dynamic distribution of attentional resources during the tasks.

REFERENCES

[1] L. Tiberio, A. Cesta, and M. O. Belardinelli, "Psychophysiological methods to evaluate user's response in human robot interaction: a review and feasibility study", *Robotics*, 2(2):92–121, 2013.

[2] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, and M. Goodrich, „Common metrics for human-robot interaction", *Proc. ACM SIGCHI/SIGART Human-robot interaction*, 2006.

[3] C.-M. Huang, and B. Mutlu. "Anticipatory robot control for efficient human-robot collaboration", *Proc. ACM/IEEE HRI 2016*, 2016.

[4] S.J. Levine and B.C. Williams, "Watching and Acting Together: Concurrent Plan Recognition and Adaptation for Human-Robot Teams", *JAIR*, vol. 63, pp. 281–359, 2018.

[5] R. G. Freedman and S. Zilberstein, "Integration of Planning with Recognition for Responsive Interaction Using Classical Planners," in *Proceedings of AAAI* 2017, pp. 4581–4588.

[6] L. Johannsmeier and S. Haddadin, "A Hierarchical Human-Robot Interaction-Planning Framework for Task Allocation in Collaborative Industrial Assembly Processes," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 41–48, Jan. 2017.

[7] M. Chen, S. Nikolaidis, H. Soh, D. Hsu, and S. Srinivasa, "Planning with Trust for Human-Robot Collaboration," in *Proceedings of the 2018 ACM/IEEE HRI'18*, Chicago, IL, USA, 2018, pp. 307–315.

[8] A. Roncone, O. Mangin, and B. Scassellati, "Transparent role assignment and task allocation in human robot collaboration" in *Proc. IEEE ICRA*, Singapore, Singapore, 2017, pp. 1014–1021.

[9] M. Fox, D. Long, and D. Magazzeni, "Explainable Planning", in *Proc. IJCAI-17 Workshop on Explainable AI (XAI)*, 2017, pp. 24–30.

[10] D. Kragic, J. Gustafson, H. Karaoguz, P. Jensfelt, and R. Krug, "Interactive, Collaborative Robots: Challenges and Opportunities," in *Proceedings of IJCAI*, Stockholm, Sweden, 2018, pp. 18–25.

[11] M. Salem, G. Lakatos, F. Amirabdollahian, and K. Dautenhahn, "Would You Trust a (Faulty) Robot?: Effects of Error, Task Type and Personality on Human-Robot Cooperation and Trust," in *Proceedings of the Tenth HRI '15*, 2015, pp. 141–148.

[12] C. Huang, C. Maya, and B. Mutlu, "Adaptive Coordination Strategies for Human-Robot Handovers", *Robotics: science and systems*, 2015.

[13] I. Maurtua, A. Ibarguren, J. Kildal, L. Susperregi and B. Sierra, "Human–robot collaboration in industrial applications: Safety, interaction and trust", *Intl. Journal of Advanced Robotic Systems*, vol. 14, no. 4, 2017.

[14] M. Fox and D. Long, "PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains," *J. Artif. Intell. Res. (JAIR)*, 2003.

[15] M. Cashmore et al., "ROSPlan: Planning in the Robot Operating System," in *Proceedings of the Twenty-Fifth International Conference on Automated Planning and Scheduling*, 2015., 2015, pp. 333–341.

[16] K. Holmqvist, M. Nyström, R. Andersson, R. Dewhusrt, H. Jarodzka, and J. van de Weijler. *Eye Tracking – A Comprehensive Guide to Methods and Measures*, Oxford University Press, 2011, pp. 187

[17] K. Moore, and L. Gugerty, "Development of a novel measure of situation awareness: The case for eye movement analysis," *Human Factors & Ergon. Soc. Ann. Meeting*, vol. 54:19, 2010

[18] S. M. Munn, and J.B. Pelz, "3D point-of-regard, position and head orientation from a portable monocular video-based eye tracker", *Proc. ETRA 2008*, pp. 181-188.

[19] K. Santner, G. Fritz, L. Paletta, and H. Mayer, "Visual recovery of saliency maps from human attention in 3D environments", *Proc. ICRA 2013*, pp. 4297-4303.

[20] C. Heyer, "Human-robot interaction and future industrial robotics applications," *Proc. IEEE/RSJ IROS*, pp. 4749–4754, 2010.

[21] R. M. Taylor, "Situational awareness rating technique (SART): The development of a tool for aircrew systems design", in *Situational Awareness in Aerospace Operations*, 1990, pp. 3/1–3/17.

[22] Stanton, N.A., P.M. Salmon, G.H. Walker, and D.P. Jenkins. Genotype and phenotype schemata and their role in distributed situation awareness in collaborative systems. *Theoret. Issues in Ergon.Sci*, 2009.

[23] Dini, A., Murko, C., Paletta, L., Yahyanejad, S., Augsdörfer, U., and Hofbaur, M. (2017) Measurement and Prediction of Situation Awareness in Human-Robot Interaction based on a Framework of Probabilistic Attention, *Proc. IEEE/RSJ IROS 2017*, 2017.

[24] Paletta, L., Pittino, N., Schwarz, M., Wagner, V., and Kallus, W., Human factors analysis using wearable sensors in the context of cognitive and emotional arousal, *Proc. 4th International Conference on Applied Digital Human Modeling*, AHFE 2015, July 2015.

[25] L. Paletta, M. Pszeida, R. Marton, T. Haspel, and Nauschnegg, B. (2019). Stress Measurement in Multi-Tasking Decision Processes Using Executive Functions Analysis, *Proc. AHFE 2019 International Conference on Human Factors and Ergonomics*, 2019

[26] Bailey, Brian P.; Konstan, Joseph A. (July 2006). On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state. *Computers in Human Behavior*. 22 (4): 685–708.

[27] Stephen Monsell, "Task Switching", *Trends in Cognitive Sciences*, Vol.7 No.3 March 2003.

[28] A. Miyake, N.P. Friedman, M.J. Emerson, A. Witzki, A. Howerter, T.D. Wager. "The unity and diversity of executive functions and their contributions to complex 'frontal lobe' tasks: A latent variable analysis". *Cognitive Psychology*, 41, 49-100, 2000.

[29] A.T. Jersild. "Mental set and shift". *Archives Psychol.*, 89.1927.

[30] Kramer, A. F., Hahn, S., & Gopher, D. (1999). Task coordination and aging: Explorations of executive control processes in the task switching paradigm. *Acta Psychologica*, 101, 339-378.

OAGM Workshop

3D Pose Estimation from Color Images without Manual Annotations

Mahdi Rad[1], Markus Oberweger[1], and Vincent Lepetit[2,1]

I. PROBLEM STATEMENT AND MOTIVATION

3D pose estimation is an important problem with many potential applications. However, 3D acquiring annotations for color images is a difficult task. To create training data, the annotating is usually done with the help of markers or a robotic system, which in both cases is very cumbersome, expensive, or sometimes even impossible, especially from color images. Another option is to use synthetic images for training. However, synthetic images do not resemble real images exactly. To bridge this domain gap, Generative Adversarial Networks or transfer learning techniques can be used but, they require some annotated real images to learn the domain transfer. To overcome these problems, we proposed a novel approach in [3]. Section II gives a short summary of our approach that uses synthetic data only, and Section III shows some results. For more details, we refer to [3].

II. OVERVIEW OF THE APPROACH

We propose a novel method that learns to predict a 3D pose from color images, without requiring labeled color images. Instead, it exploits labeled depth images. Depth maps are easier to generate than realistic color images, as illumination has no interference on depth, and as we will show, the gap between color images and depth maps, and the gap between real and synthetic depth maps can be dealed with easily. Fig. 1 shows an overview.

Training: Our method is split into two steps, each one solving an easier problem than the original one. First, we use an RGB-D camera to capture pairs of color images and corresponding depth maps. Capturing such a set can be done by simply moving the camera around. We apply [4] to this set and learn to map the features from the color images to corresponding depth images. However, this mapping alone is not sufficient: A domain gap between the depth images captured by the RGB-D camera and the available labeled depth images remains, since the labeled depth images could be captured with another RGB-D camera or rendered synthetically. Fortunately, this remaining gap is easier to bridge than the domain gap between real and synthetic color images, since illumination and texture effects are not present in depth images. To handle it, we use Maximum Mean Discrepancy (MMD) [1] to measure and minimize the distance between the means of the features of the real and synthetic depth images mapped into a Reproducing Kernel Hilbert Space (RKHS).

[1]Institute for Computer Graphics and Vision, Graz University of Technology, Graz, Austria {rad, oberweger}@icg.tugraz.at
[2]LaBRI, Université de Bordeaux, Bordeaux, France vincent.lepetit@u-bordeaux.fr

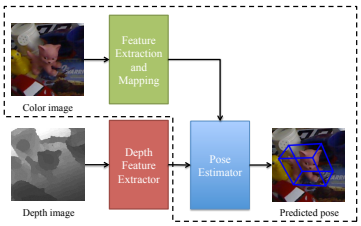


Fig. 1. Method overview. We train a depth feature extractor (red box) together with a pose estimator (blue box). We also train a second network (green box), which extracts image color features and maps them to the depth space, given color images and their corresponding depth images. At run-time, given a color image, we map color features to depth space in order to use the pose estimator to predict the 3D pose of the object (dashed lines). This removes the need for labeled color images.

Inference: At run-time, given a real color image, we extract its features in color space and map them to the depth feature space, and then use a pose estimator trained on depth images to predict the 3D pose of the object.

III. DISCUSSION AND ILLUSTRATIVE RESULTS

Our approach is general, and can be applied to many applications, such as 3D hand pose estimation, human pose estimation, etc. Fig. 2 shows applications to 3D rigid object pose estimation and 3D hand pose estimation from color images, on the LINEMOD [2] and STB [5] datasets, respectively. Our method achieves performances comparable to state-of-the-art methods on popular benchmark datasets, without requiring any annotations for the color images.

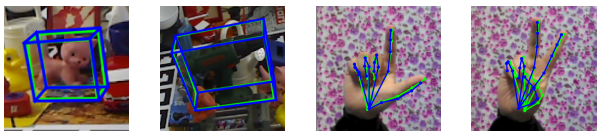


Fig. 2. Our method allows very accurate 3D pose estimation from color images without annotated color images. In case of 3D rigid object pose estimation, we draw the bounding boxes, and for 3D hand pose estimation, we show the 3D joint locations projected to the color image. Green denotes ground truth and blue corresponds to the predicted pose.

ACKNOWLEDGMENT

This work was supported by the Christian Doppler Laboratory for Semantic 3D Computer Vision, funded in part by Qualcomm Inc.

REFERENCES

[1] A. Gretton, K. Borgwardt, M. J. Rasch, B. Schölkopf, and A. J. Smola, "A Kernel Method for the Two-Sample Problem," 2006.
[2] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model Based Training, Detection and Pose Estimation of Texture-Less 3D Objects in Heavily Cluttered Scenes," 2012.
[3] M. Rad, M. Oberweger, and V. Lepetit, "Domain Transfer for 3D Pose Estimation from Color Images without Manual Annotations," 2018.
[4] ——, "Feature Mapping for Learning Fast and Accurate 3D Pose Inference from Synthetic Images," 2018.
[5] J. Zhang, J. Jiao, M. Chen, L. Qu, X. Xu, and Q. Yang, "3D Hand Pose Tracking and Estimation Using Stereo Matching," 2016.

On the Use of Artificially Degraded Manuscripts for Quality Assessment of Readability Enhancement Methods*

Simon Brenner[1] and Robert Sablatnig[1]

***Abstract*— This paper reviews an approach to assess the quality of methods for readability enhancement in multispectral images of degraded manuscripts. The idea of comparing processed images of artificially degraded manuscript pages to images that were taken before their degradation in order to evaluate the quality of digital restoration is fairly recent and little researched. We put the approach into a theoretical framework and conduct experiments on an existing dataset, thereby reproducing and extending the results described in the first publications on the approach.**

I. INTRODUCTION

Written heritage is a valuable resource for historians and linguists. However, the physical medium preserved may be in a condition that prohibits the direct access of the text: fading of the ink, darkening of the substrate or artificial removal due to substrate re-use (palimpsestation) are possible circumstances that render a text unreadable. Several imaging methods have been employed to recover such lost writings in the last fifteen years, with Multispectral and Hyperspectral imaging as well as X-Ray Fluorescence mapping being the most prominent base techniques [2], [7], [10], [11], [13], [20], [23], [24]. While the hardware systems are continually improved, post-processing methods for readability enhancement that were adapted for Multispectral Images of manuscripts over a decade ago [24] are still used by practitioners and prominently appear in recent literature [2], [10], [20]. Developments in the area are impeded by the absence of suitable metrics for automatically evaluating the quality of the results.

In literature describing methods for readability improvements of written heritage, evaluations based on expert ratings or the demonstration on selected examples is common [7], [13], [20], [23], [24]. This practice is unfavorable for the research field, as the evaluation of methods on large datasets is unfeasible if human assessment of results is required. Considering the fact that the development of computer vision methods typically involves multiple iterations of testing and improvement [31], the problem becomes even more apparent. A similar problem is faced by the practitioner, who is forced to manually try and visually evaluate a palette of methods in order to find the optimal result for a given investigated object.

*This work was supported by the Austrian Science Fund (FWF), grant no. P29892

[1]Simon Brenner and Robert Sablatnig are with the Computer Vision Lab, Institute of Visual Computing & Human-Centered Technology, TU Wien, 1040 Vienna, Austria sbrenner@cvl.tuwien.ac.at, sab@cvl.tuwien.ac.at

We propose that an ideal metric for the assessment of readability should have the following properties:

1) *Unsupervised.* The readability assessment does not require user input, such as selection of different pixel classes. Furthermore, a readability score can be calculated for unknown documents, i.e. documents where the contained text is not known a-priori.
2) *Culture agnostic.* The assessment is applicable to writings of any script and language equally.
3) *Consistent with expert ratings.* At the end of the day, domain experts still possess the highest authority for readability assessment, as it is them who then actually read the texts. Therefore, a ranking of a given set of enhancement results based on the calculated readability score should coincide with a ranking created by a domain expert.

Such a metric not only facilitates efficient testing and benchmarking, but also allows for optimization-based parameter tuning for postprocessing algorithms or the pre-selection of the best images from a large number of results from different algorithms.

A. Previous Approaches for Quantitative Evaluation

Several attempts for a quantitative assessment of text restoration quality are found in literature.

Arsene et al. [2] conducted a study on the effectiveness of a number of dimensionality reduction methods on a certain manuscript page. In addition to the obligatory score by expert rating, they employed the Davies-Bouldin Index and the Dunn Index, which are measures for cluster separability, as quality metrics. While all three metrics agreed on the best enhancement method, for the remaining positions of the ranking the computed scores diverged significantly from the human ratings, making their feasibility questionable. The authors acknowledge this and claim that the visual assessment by philologists is still the standard method of evaluating readability enhancement methods.

A natural assumption is that the quality of an image with regard to readability is strongly connected to its contrast. This is problematic however, as high contrast can be found in background noise and non-textual elements of a page (e.g. in the form of stains), especially when dealing with results of dimensionality reduction methods. Furthermore, the nominal contrast of an image can be increased by simple intensity transformations, thus rendering it impractical for the assessment of image quality. Faigenbaum et al. rely on the notion of *potential contrast* [26] to assess the readability

of ostraca [8]. This measure rates the maximum contrast between foreground and background of a grayscale image that can be achieved by any intensity transformation. Although an intriguing idea, its implementation is problematic, as it relies on a binarization of the image by means of manually selected samples of foreground and background pixels, and the resulting score heavily depends on those samplings.

Another approach is to measure the quality of enhancement strategies by the performance of Optical Character Recognition (OCR) [14], [17]. In comparison to the preceding approaches, the evaluation via OCR performance has the advantage of directly related to the property of 'readability'. However, a ground truth is required and the results depend on the OCR algorithm employed and the data on which it was trained. Hollaus et al. [14], for example, evaluate their work on Glagolitic script and use a custom OCR system that has been trained for Glagolitic script only.

B. Image Quality Assessment

A closely related topic is the general Image Quality Assessment (IQA). Relevant approaches are categorized by the amount of information available to the estimator [5], [21].

Full-Reference (FR) methods have knowledge of a reference image that is assumed to be of optimal quality. The quality score is in essence a metric for the similarity between the reference image and a degraded version [1], [30]; a typical use case is the evaluation of lossy image compression, where an original image is naturally available.

No-Reference (NR) methods require no additional information aside from the input image that is to be evaluated. Successful NR IQA approaches, that are not limited to a certain type of distortion, typically employ machine learning one way or the other [4]. While early methods based on natural scene statistics, such as DIIVINE [22] or BRISQUE [21], are largely hand-crafted and just 'calibrated' on a training dataset, recent publications make heavy use of Convolutional Neural Networks (CNNs) [4], [5], [18], [15]. NR-IQA has been used to select optimal parameters for de-noising [21], [33] and artifact removal in image synthesis [3].

The problem of quantitatively evaluating readability enhancements can be considered a special case of IQA. For this application, however, a reference image is typically not available. It is thus natural that, using the taxonomy above, the assessment approaches outlined in Section I-A fall in the category of NR IQA (or Reduced-Reference IQA [30] in the case of evaluations based on OCR-performance). Although a NR approach would be preferable for the application, it is generally an ill-posed problem [18], even more so when focusing on the property of readability [10]. None of the approaches described above satisfies the requirements for an assessment metric we formulate. It is thinkable that CNN based approaches similar to those used for general NR IQA problems can be adapted and trained for readability assessment and used in a processing workflow for parameter optimization or pre-selection from a set of different results. For evaluation and benchmarking applications, however, CNNs are not a feasible option due to their dependence on a specific training process (which even introduces random components in the usual case of stochastic gradient descent optimization) [12] and the general opacity of their decision making [32].

C. Artificial Degradation

Giacometti et al. proposed a way to perform readability assessment in a FR setting [10]. They cut patches from an 18th century document written with iron gall ink on parchment and acquired Multispectral images before and after artificial degradation by various treatments. The resulting dataset [9] consists of 23 manuscript patches, of which 20 were subject to a different treatment each and three were left untreated as control images. Two of the patches were imaged from both sides, giving a total of 25 samples.

The dataset is then used to conduct a study on the performance of Multispectral imaging and postprocessing techniques in recovering information lost in the degradation process. The result images were compared with the untreated originals, which allows to view the approach as an instance of the FR IQA problem. The authors employ mutual information [29] as a similarity metric.

This work is of value and significance because to the best of our knowledge, it resulted in the first dataset systematically documenting the effects of degradation processes on the spectral response of written text and potentially enabling an objective evaluation of attempts to restore the original information. However, it has several restrictions for a broader application: First, the number of samples is small and, as all the samples are taken from the same manuscript, there is no variation in substrate and ink composition. Second, the important case of palimpsestation, i.e. the presence of a new layer of text on top of the degraded one, is omitted. Third, the accompanying paper [10] fails to conclusively show that comparison with the original image is a valid method to assess the quality of text restoration. Although plausible results are shown for selected examples, the generality of the results is not discussed; also it is not made clear which exact image is used for reference to obtain the specified mutual information scores. However, this is a prerequisite to legitimate further studies of this kind with a higher number of samples and greater variation.

In the following, we reproduce and extend the results described in the original paper in order to further investigate this third issue.

II. Continuative Experiments

The dataset described above contains multispectral images acquired with a monochromatic scientific camera as well as color images. In the following, we will only refer to the monochromatic images. For each sample, 21 spectral layers from 400nm to 950nm are available for the untreated and treated variants. The layers are intensity normalized [19] and inter-registered; however, the treated images are not registered to the untreated ones. Also a set of results from

dimensionality reduction methods is provided for each sample; they are registered to the untreated variants, but far from pixel-accurately, prohibiting quantitative comparisons.

A. Preprocessing

For greater flexibility and accuracy we pre-processed the dataset prior to our experiments:

1) From the untreated image, a pan-chromatic image is created by averaging the layers in the visible range ($400nm < \lambda < 700nm$). For the sake of simplicity and uniformity, these panchromatic images will serve as a reference for registration and comparison, and will from here on be referred to as *reference*.
2) One layer of the treated sample is registered to the reference using a deformable registration framework for medical image processing [16], [25]. The 800nm layer was chosen for that purpose, as a visual assessment showed that it shares most of the textual information with the untreated images for the majority of degradation types. A deformable registration approach is necessary due to deformations of the parchment resulting from the treatments.
3) The remaining treated images are registered using the transformation found in the previous step.
4) Panchromatic images and registered treated images are cropped to 900x900 pixels.
5) To produce test images that can be compared with the reference, the cropped registered treated images are processed with five common (but arbitrarily chosen) dimensionality reduction methods: Principal Component Analysis (PCA), Independent Component Analysis (ICA), Factor Analysis (FA), Truncated Singular Value Decomposition (T-SVD) and K-Means Clustering (KM). From each method, five components were extracted, leading to a total of 25 processed variants for each sample, from here on referred to as *processed images*.

The three samples treated with heat, mold and sodium hypochlorite could not be registered satisfactorily due to their condition and were thus omitted, leaving 22 samples for investigation. The resulting modified version of the dataset is available online [6].

B. Comparison metrics

The images retrieved from dimensionality reduction methods visualize statistical dependencies rather than measured intensity values, such that contrast, mean brightness and *polarity* (in our case referring to dark text on light background versus light text on dark background) of these images typically deviate from the original photographs [10]. Therefore, any comparison metrics that rely on absolute intensity differences, such as the Mean Squared Error or Peak Signal To Noise Ratio, are unsuitable for this application. Instead, metrics that provide a measure of structural similarity and are insensitive to contrast and polarity are required.

Viewing the pixel positions as observations and the intensity values of the compared images as observed variables, statistical measures of dependence such as the Pearson Correlation Coefficient (PCC) and Mutual Information (MI) between the variables (i.e. images) are available as relevant comparison metrics. While MI, which Giacometti et al. employed in their work [10], can be used as-is, reversed polarities result in negative PCC vales such that the absolute value is used as a score.

Alternatively, established NR IQA metrics emphasizing structural similarity like the Structural Similarity Index (SSIM) [30] and Visual Information Fidelity (VIF) [27] are available. Although these metrics are not agnostic of contrast, its influence can be adjusted with a parameter for SSIM, while VIF actually rewards images with higher contrast than the reference. To make the methods invariant to polarity, we simply use $max(\varphi(I_{ref}, I_{test}), \varphi(I_{ref}, \neg I_{test}))$ as a comparison score, where φ denotes either SSIM of VIF between to images and $\neg$ is the image complement.

We consciously refrain from employing more advanced FR IQA metrics (e.g. based on learning) for these initial experiments as they would introduce unnecessary complexity.

C. Experiments

In order to reproduce previous results [10] and investigate the feasibility of comparison with an intact original as a measure for readability, we compare each processed image with the reference using MI as well as the adapted variants of PCC, SSIM and VIF described above. The use of additional similarity metrics allows to observe if the choice of metric significantly influences the results. The scores were then used to create rankings of the processed images for each sample, allowing to visually assess their plausibility.

In addition, the influence of contrast enhancement on the respective scores was experimentally evaluated: For each sample, the first five Principal Components (showing varying degrees of initial contrast) were subjected to Contrast Limited Adaptive Histogram Equalization (CLAHE) with varying clip limits, to monitor the influence on the different scores.

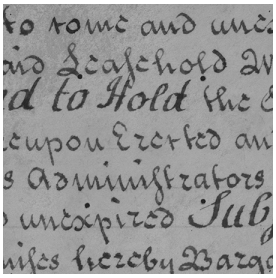
The full results of our experiments as well as relevant source code can be accessed online along with our preprocessed version of the dataset [6].

III. Discussion

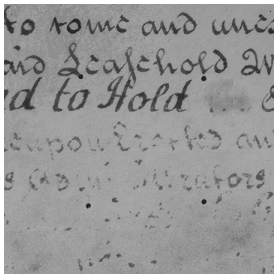
Visually assessing the processed image variants ranked by the employed comparison metrics generally confirms the assumption that similarity to a non-degraded reference image correlates well to the readability of text. The example shown in Figure 1 is representative for the remaining samples, where similar situations are observed.

The rankings derived from different similarity metrics are well correlated, with MI and PCC showing the strongest agreement. This is comprehensible when visually assessing rankings like in Figure 1c, and also manifests in the correlation matrix of the different metrics, which is shown in Table I.

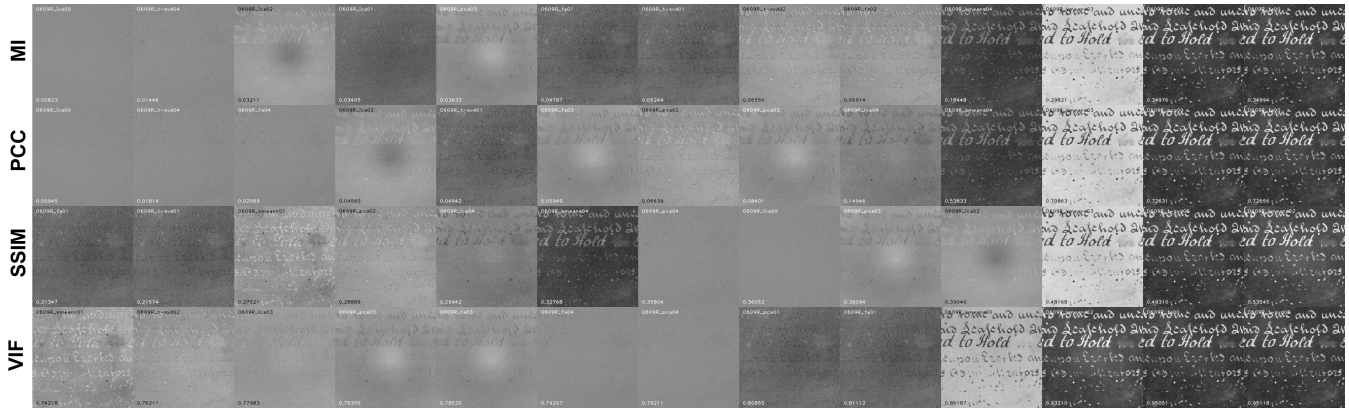
It might seem like the good scores of the highest ranked images are due to their high contrast; this general assumption, however, is readily disproved. Experiments with



(a) Untreated (panchromatic)



(b) Scraped (panchromatic)



(c) Ranked processed images

Fig. 1: An example of quality rankings derived from comparison with a reference image. (a) and (b) show panchromatic images of a sample of the dataset before and after artificial degradation via scraping. The rows of (c) correspond to the different metrics employed; the columns are ordered in ascending quality score. Due to space limitations, we only show every third column of the ranking.

	MI	PCC	SSIM	VIF
MI	1.0	0.9117	0.8189	0.7534
PCC	0.9117	1.0	0.8004	0.7211
SSIM	0.8189	0.8004	1.0	0.7395
VIF	0.7534	0.7211	0.7395	1.0

TABLE I: Correlation matrix of different employed similarity metrics, computed over all compared variants.

different levels of generic contrast enhancement showed that it has no positive effect on the scores. On the contrary, the SSIM and VIF scores decrease with increasing contrast. Figure 2 plots the mean deviations of similarity scores over the clip limit used for CLAHE contrast enhancements, along with the respective standard deviations. Note that the mean MI and PCC scores remain almost constant, whereby MI exhibits lower standard deviations. MI is thus the most stable of the tested metrics with respect to contrast alterations. The finding that generic contrast enhancements do not improve comparison scores is comprehensible, because the contrast of signal and noise is enhanced likewise. It also suggests that high comparison scores result from contrast that is also present in the original image (especially between text and foreground), which in turn supports the feasibility of image comparison as a quality metric for text restoration.

Although the results are visually convincing in general, individual examples for obviously erroneous ratings are

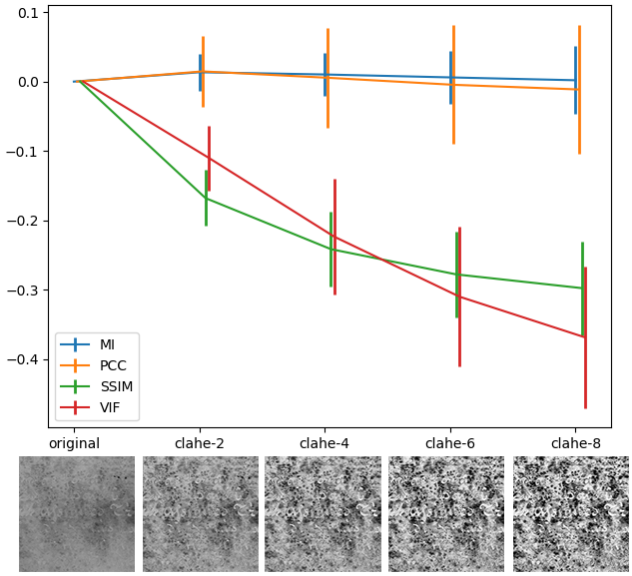
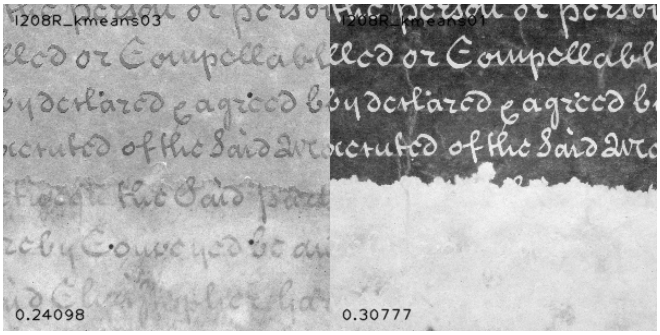
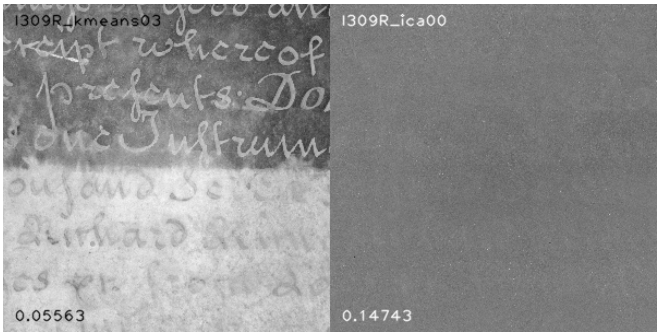


Fig. 2: The effect of applying CLAHE with increasing clip limit to the processed images before comparison with the respective metrics. Standard deviations are shown as vertical bars. The images below the plot give an example of a source image and resulting contrast-enhanced images. Note that the background structure is enhanced as well as the text.



(a) Irregularity in MI score



(b) Irregularity in PCC score

Fig. 3: Examples of wrong ratings. Images on the right were rated higher than images on the left.

found frequently. Figure 3 shows examples. The reason for those errors have not been investigated yet.

To definitely validate the feasibility of the approach, a user study is necessary to obtain a strong ground truth dataset containing subjective quality ratings from multiple individuals. Such datasets are the basis for any quantitative evaluation of image quality metrics, just as it is the case for general IQA problems [4], [28].

IV. Conclusion

In this paper we have surveyed the approach to assess the quality of readability enhancement methods by comparison with intact reference images, both theoretically and experimentally, and formulated it as a special case of Full-Reference Image Quality Assessment. Intuitively the approach is sensible, because the goal of any digital restoration is to produce results as similar to the originals as possible. Using four relatively simple image comparison metrics we produced visually convincing rankings of processed images; however, cases where the method fails were observed as well. In general, the four tested metrics correlate well, with Mutual Information and Pearson Correlation Coefficient showing the strongest agreement. We also showed that generic contrast enhancements have no positive effects on the comparison score and identified Mutual Information as the most stable metric in this regard. However, for a definite confirmation of the validity of this approach, a set of test images with expert-rated readability scores is required. To this end, a systematic user study is necessary. Only with this prerequisite can an improvement of the method be attempted, that is, the development of a of a more specialized and stable metric for image comparison. These attempts can also pave the way for the exploration of No-Reference IQA methods for readability assessment, which would be the optimal solution for this problem.

References

[1] S. A. Amirshahi, M. Pedersen, and S. X. Yu, "Image Quality Assessment by Comparing CNN Features between Images," *Journal of Imaging Science and Technology*, vol. 60, no. 6, pp. 60 410–1–60 410–10, 2016.

[2] C. T. C. Arsene, S. Church, and M. Dickinson, "High performance software in multidimensional reduction methods for image processing with application to ancient manuscripts," *Manuscript Cultures*, vol. 11, pp. 73–96, 2018.

[3] T. O. Aydn, K. I. Kim, K. Myszkowski, and H.-p. Seidel, "NoRM : No-Reference Image Quality Metric for Realistic Image Synthesis," *Computer Graphics Forum*, vol. 31, no. 2, 2012.

[4] S. Bianco, L. Celona, P. Napoletano, and R. Schettini, "On the use of deep learning for blind image quality assessment," *Signal, Image and Video Processing*, vol. 12, no. 2, pp. 355–362, 2018.

[5] S. Bosse, D. Maniry, K.-r. Müller, T. Wiegand, and W. Samek, "Deep Neural Networks for No-Reference and Full-Reference Image Quality Assessment," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 206–219, 2018.

[6] S. Brenner, "On the Use of Artificially Degraded Manuscripts for Quality Assessment of Readability Enhancement Methods - Dataset & Code," DOI: 10.5281/zenodo.2650152FNo, 2019. [Online]. Available: https://doi.org/10.5281/zenodo.2650152

[7] R. L. Easton, W. A. Christens-Barry, and K. T. Knox, "Spectral image processing and analysis of the Archimedes Palimpsest," *European Signal Processing Conference*, no. Eusipco, pp. 1440–1444, 2011.

[8] S. Faigenbaum, B. Sober, A. Shaus, M. Moinester, E. Piasetzky, G. Bearman, M. Cordonsky, and I. Finkelstein, "Multispectral images of ostraca: Acquisition and analysis," *Journal of Archaeological Science*, vol. 39, no. 12, pp. 3581–3590, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.jas.2012.06.013

[9] A. Giacometti, A. Campagnolo, L. MacDonald, S. Mahony, S. Robson, T. Weyrich, and M. Terras, "UCL Multispectral Processed Images of Parchment Damage Dataset," DOI: 10.14324/000.ds.1469099, 2015. [Online]. Available: http://discovery.ucl.ac.uk/id/eprint/1469099

[10] A. Giacometti, A. Campagnolo, L. MacDonald, S. Mahony, S. Robson, T. Weyrich, M. Terras, and A. Gibson, "The value of critical destruction: Evaluating multispectral image processing methods for the analysis of primary historical texts," *Digital Scholarship in the Humanities*, vol. 32, no. 1, pp. 101–122, 2017.

[11] L. Glaser and D. Deckers, "The Basics of Fast-scanning XRF Element Mapping for Iron-gall Ink Palimpsests The Basics of Fast-scanning XRF Element Mapping for Iron-gall Ink Palimpsests," *Manuscript Cultures*, vol. 7, no. December 2013, 2016.

[12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, pp. 275–276, http://www.deeplearningbook.org.

[13] F. Hollaus, M. Diem, S. Fiel, F. Kleber, and R. Sablatnig, "Investigation of Ancient Manuscripts based on Multispectral Imaging," *DocEng 2015 - Proceedings of the 2015 ACM Symposium on Document Engineering*, no. 1, pp. 93–96, 2015.

[14] F. Hollaus, M. Diem, and R. Sablatnig, "Improving OCR accuracy by applying enhancement techniques on multispectral images," *Proceedings - International Conference on Pattern Recognition*, pp. 3080–3085, 2014.

[15] L. Kang, P. Ye, Y. Li, and D. Doermann, "Convolutional neural networks for no-reference image quality assessment," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1733–1740, 2014.

[16] S. Klein, M. Staring, K. Murphy, M. A. Viergever, and J. P. W. Pluim, "elastix: A toolbox for intensity-based medical image registration," *IEEE Transactions on Medical Imaging*, vol. 29, no. 1, pp. 196–205, Jan 2010.

[17] L. Likforman-Sulem, J. Darbon, and E. H. Smith, "Enhancement of historical printed document images by combining Total Variation regularization and Non-local Means filtering," *Image and Vision*

Computing, vol. 29, no. 5, pp. 351–363, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.imavis.2011.01.001

[18] K.-Y. Lin and G. Wang, "Hallucinated-IQA: No-Reference Image Quality Assessment via Adversarial Learning," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 732–741, 2018.

[19] L. MacDonald, A. Giacometti, A. Campagnolo, S. Robson, T. Weyrich, M. Terras, and A. Gibson, "Multispectral imaging of degraded parchment," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7786 LNCS, pp. 143–157, 2013.

[20] S. Mindermann, "Hyperspectral Imaging for Readability Enhancement of Historic Manuscripts Hyperspectral Imaging for Readability Enhancement of Historic Manuscripts," Master's thesis, TU München, 2018.

[21] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-Reference Image Quality Assessment in the Spatial Domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, 2012.

[22] A. K. Moorthy and A. C. Bovik, "Blind Image Quality Assessment: From Natural Scene Statistics to Perceptual Quality," *IEEE Transactions on Image Processing*, vol. 20, no. 12, pp. 3350–3364, 2011.

[23] E. Pouyet, S. Devine, T. Grafakos, R. Kieckhefer, J. Salvant, L. Smieska, A. Woll, A. Katsaggelos, O. Cossairt, and M. Walton, "Revealing the biography of a hidden medieval manuscript using synchrotron and conventional imaging techniques," *Analytica Chimica Acta*, vol. 982, pp. 20–30, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.aca.2017.06.016

[24] E. Salerno, A. Tonazzini, and L. Bedini, "Digital image analysis to enhance underwritten text in the Archimedes palimpsest," *International Journal on Document Analysis and Recognition*, vol. 9, no. 2-4, pp. 79–87, 2007.

[25] D. Shamonin, E. Bron, B. Lelieveldt, M. Smits, S. Klein, and M. Staring, "Fast parallel image registration on cpu and gpu for diagnostic classification of alzheimer's disease," *Frontiers in Neuroinformatics*, vol. 7, p. 50, 2014. [Online]. Available: https://www.frontiersin.org/article/10.3389/fninf.2013.00050

[26] A. Shaus, S. Faigenbaum-Golovin, B. Sober, and E. Turkel, "Potential Contrast – A New Image Quality Measure," *Electronic Imaging*, vol. 2017, no. 12, pp. 52–58, 2017.

[27] H. R. Sheikh and A. C. Bovik, "Image Information and Visual Quality," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, 2006.

[28] H. Sheikh, Z. Wang, L. Cormack, and A. Bovik, "LIVE Image Quality Assessment Database Release 2." [Online]. Available: http://live.ece.utexas.edu/research/quality

[29] P. Viola and W. M. Wells III, "Alignment by maximization of mutual information," *International Journal of Computer Vision*, vol. 24, no. 2, pp. 137–154, Sep 1997. [Online]. Available: https://doi.org/10.1023/A:1007958904918

[30] Z. Wang, A. C. Bovik, H. R. Sheikh, S. Member, E. P. Simoncelli, and S. Member, "Image Quality Assessment: From Error Visibility to Structural Similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[31] D. Xin, L. Ma, S. Song, and A. Parameswaran, "How Developers Iterate on Machine Learning Workflows – A Survey of the Applied Machine Learning Literature," 2018.

[32] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., 2014, pp. 818–833.

[33] X. Zhu, S. Member, and P. Milanfar, "Automatic Parameter Selection for Denoising Algorithms Using a No-Reference Measure of Image Content," *IEEE Transactions on Image Processing*, vol. 19, no. 12, pp. 3116–3132, 2010.

visIvis - Evaluation of Vision based Visibility Measurement

Dr. Harald Ganster[1], DI Jean-Philippe Andreu[1], Dr. Martina Uray[1] and Dipl.-Met. Jürgen Lang[2]

I. Problem Statement and Related Work

In air traffic management (ATM) and monitoring of critical infrastructure, the exact description of the atmospheric state - and thus the visibility - is an indispensable basis for any further weather forecast. In order to overcome the drawbacks of the currently subjective reports from human observers, we present "visIvis", an innovative solution to automatically derive visibility measures from standard cameras by a vision-based approach.

Several research aims already target camera-based visibility estimation. E.g. the weather services of the Netherlands (KNMI [3]) rely on a few manually selected landmarks with known distances to derive the prevailing visibility. A variety of approaches uses physics-based models to derive a visibility measure (e.g. Koschmieder model or other measures of contrast [1] as well as models of light extinction [2]).

II. visIvis-System

Based on a small set of training images with very good and bad visibility conditions, the system visIvis detects fully automatic the best suited image regions for visibility derivation (e.g. buildings or orographic structures). At the same time the optimal parameters (e.g. image features and quality criteria) for the visibility recognition are derived. Assessment of visibility is done separately for each image region and displayed according to customer preferences (e.g. a red-green mask as illustrated in Figure 1)

The system further integrates a pixel-precise distance map by georeferencing the calibrated camera image and exploiting high-resolution elevation data derived from digital surface models (DSM). A statistical analysis of visible and non-visible image regions delivers a representative estimate of the prevailing visibility for the complete camera-covered area.

III. Practical Examples and Discussion

First installations of visIvis are already in use by ANSPs and weather services (e.g. AustroControl and Deutscher Wetterdienst) that proof the system's ability to support automated assessment of the weather situation by standard cameras.

Although the second example for the Hamburg application (Figure 1) is disturbed by raindrops, visIvis can reliable detect the correct prevailing visibility of > 10.000m. The raindrops cause some misclassifications of individual image regions, but due to the assessment of the complete image visIvis is very robust against such artefacts.

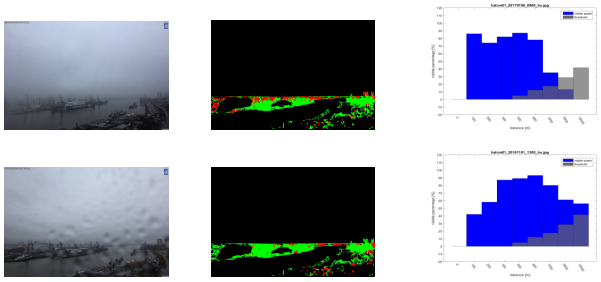


Fig. 1. visIvis results for the location Hamburg: deterministic visibility in top row is 1.500m, bottom row: > 10.000m. Raw image (left), visibility detection (red/green, middle), and corresponding visibility histogram (right), (courtesy of Deutscher Wetterdienst).

Figure 2 illustrates a span of 2 hour comparison between runway visual range sensors (black line), the visIvis result (green) and the officially stated observer report (black dots). It clearly indicates the value of the automated measurements that were used as a tool to check the plausibility of the RVR measurements.

References

[1] K. Du, K. Wang, P. Shi, and Y. Wang, "Quantification of Atmospheric Visibility with Dual Digital Cameras during Daytime and Nighttime," *Atmospheric Measurement Techniques*, vol. 6, pp. 2121–2130, 08 2013.

[2] N. Graves and S. Newsam, "Visibility cameras: Where and how to look," in *Proceedings of the 1st ACM International Workshop on Multimedia Analysis for Ecological Data*, ser. MAED '12. New York, NY, USA: ACM, 2012, pp. 7–12. [Online]. Available: http://doi.acm.org/10.1145/2390832.2390835

[3] W. Wauben and M. Roth, "EXPLORATION OF FOG DETECTION AND VISIBILITY ESTIMATION FROM CAMERA IMAGES," in *WMO Technical Conference on Meteorological and Environmental Instruments and Methods of Observation, CIMO TECO 2016*, no. Session 2, O2 (8), 2016.

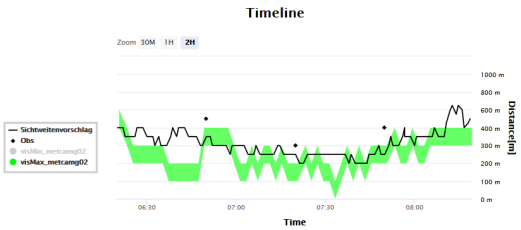


Fig. 2. Comparison of visIvis measurements with observer reports and RVR-sensors (courtesy of AustroControl).

[1]JOANNEUM RESERACH Forschungsgesellschaft mbH, DIGITAL - Institute for Information and Communication Technologies, Steyrergasse 17, 8010 Graz, Austria

[2]MeteoSolutions GmbH, Wilhelminenstraße 2, 64283 Darmstadt, Germany

The authors would like to thank AustroControl and Deutscher Wetterdienst for providing image material and reference data as well as for valuable contributions during the evaluation.

Efficient Multi-Task Learning of Semantic Segmentation and Disparity Estimation*

Robert Harb[1] and Patrick Knöbelreiter[1]

***Abstract*— We propose a jointly trainable model for semantic segmentation and disparity map estimation. In this work we utilize the fact that the two tasks have complementary strength and weaknesses. Traditional depth prediction algorithms rely on low-level features and often have problems at large textureless regions, while for semantic segmentation these regions are easier to capture. We propose a CNN-based architecture, where both tasks are tightly interconnected to each other. The model consists of an encoding stage which computes features for both tasks, semantic segmentation and disparity estimation. In the decoding stage we explicitly add the semantic predictions to the disparity decoding branch and we additionally allow to exchange information in the intermediate feature representations. Furthermore, we set the focus on efficiency, which we achieve by the usage of previously introduced ESP building blocks. We evaluate the model on the commonly used KITTI dataset.**

I. INTRODUCTION

Semantic segmentation and disparity estimation are active fields of research in computer vision. Most state-of-the-art work focuses on single-task models which perform either disparity estimation or semantic segmentation exclusively. In our work we create a multi-task model which performs both tasks simultaneously. This is mainly motivated by the following points: First, a single model performing multiple tasks jointly can be more efficient than using a separate model for each task. Therefore, multi-task learning is especially attractive for applications with limited resources and for real-time processing. Furthermore, it has been shown that the synergy among the individual tasks can boost their individual performances [22]. Except from these performance benefits information about both, the objects in the scene as well as their distance, respectively is important for many real world tasks. Such tasks could be for example autonomous driving or robot navigation.

In this work, we tackle both tasks, semantic segmentation *and* depth estimation, in a joint setting. We therefore start from two individually trained models and step by step fuse them to a joint model. The joint model allows to share computed features and therefore reduces the number of necessary operations yielding an efficient model. Furthermore, this setting allows to exchange information between the two tasks, which helps to increase the individual performances. We train the joint model end-to-end and evaluate the performance on the KITTI dataset.

*This work was supported by the research initiative Intelligent Vision Austria with funding from the AIT and the Austrian Federal Ministry of Science, Research and Economy HRSM programme (BGBl. II Nr. 292/2012)

[1]Institute of Computer Graphics and Vision, Graz University of Technology, Austria {robert.harb@student.tugraz.at, knoebelreiter@icg.tugraz.at}

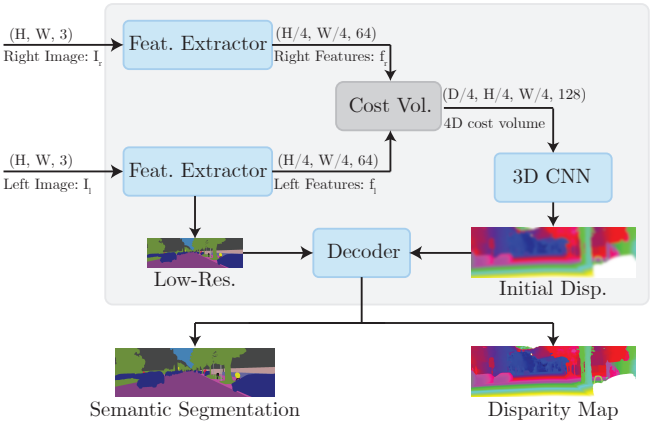


Fig. 1: **Overview** of our proposed architecture. Blue boxes denote learnable building-blocks of our model. The two feature extractor blocks have shared weights and are used for the semantic segmentation and for the disparity computation.

II. RELATED WORK

Semantic Segmentation: Given an input image, a semantic image segmentation algorithm assigns a semantically meaningful label for every pixel in the image. Labels can for example be *car*, *person* or *vegetation*. Applications of semantic segmentation exist in various areas including e.g., autonomous driving [6], [5], brain tumor segmentation [8], or segmentation of satellite images [21]. Recent work utilizes mainly Convolutional Neural Network (CNN) based approaches for semantic segmentation. In the following we divide them into two groups: The first group contains large and resource demanding models which try to improve top benchmark scores on common datasets like KITTI [6] or Cityscapes [5]. Top performing methods of this group are for example Deeplabv3+ [3] or the PSPNet [29]. Those models are able to produce high quality results at the cost of requiring a lot of parameters, memory and computational time. The prior mentioned method Deeplabv3+ uses $\approx$44M parameters and the PSPNet has $\approx$ 66M parameters. This makes them unsuitable for usage in memory restricted environments or for applications where real-time processing is needed.

The second group contains lightweight and therefore efficient models. They try to find optimal trade-offs between computational complexity and performance in terms of segmentation accuracy. Representatives of this group are e.g. MobileNet [10], ShuffleNetV2 [18], ENet [20] or ESPNet [19]. ESPNet is for example $\approx$ 22 times faster while being $\approx$ 180 times smaller than PSPNet. The semantic segmentation performance is still very good. ESPNet achieves a mean-Intersection-over Union (mIoU) score which is only $\approx$ 8% lower than the PSPNet score on Cityscapes benchmark. This

increase in computational efficiency is mainly achieved by incorporating the convolution factorization principle. Due to these benefits, we use parts of the ESPNet as basic building blocks in our model.

Disparity Estimation: Estimation of depth from image data has a wide variety of applications such as e.g. 3D reconstruction and augmented reality. In this work we focus on depth estimation from a rectified pair of stereo images. Similar as for semantic segmentation currently best performing models use CNNs. The first who showed how CNNs can be used for stereo matching have been Zbontar and LeCun [28]. Since then many CNN based stereo methods have been proposed such as PSMNet [2], Edge-Stereo [25] or CNN-CRF [14]. Even though CNN based architectures work well for stereo matching in many cases they still have problems in homogeneous regions, occlusions or reflections. One approach to mitigate this problems is to enlarge the receptive field by using e.g., multiscale features in a 3D CNN regression module [12] or dilated convolutions [17]. Another method is to incorporate semantic scene understanding, this has been done in non CNN based approaches by Ladicky et al. who construct separate random fields for disparity and semantic labeling and optimize them jointly [16]. And more recently by Displets which resolve ambiguities by incorporating object knowledge by modeling 3D vehicles [7], and SegStereo where semantic features are embedded in a cost volume for disparity prediction [27]. SegStereo is the most similar approach to our work; however, there are several fundamental differences. SegStereo uses a Resnet-50 for feature extraction, our architecture is based on ESPNet which is computationally more efficient. They perform a 1D-correlation on the cost volume, whereby we use a 3D-CNN for regularization. Additionally, they add an unsupervised photometric and smoothness component to their loss.

Multi-task learning: The general objective of multi-task learning is to improve generalization by leveraging the domain-specific information contained in the training signals of related tasks [1]. In neural network based architectures this can be done by sharing parts or the whole internal representation among different tasks. When considering efficiency, multi-task learning is attractive. Both memory and computational time can be reduced if we do not need a separate network for each task. This has been shown for example in UberNet [15] where as many as 7 tasks have been trained jointly including among others semantic segmentation and surface normals. Another representant is HyperFace [22] which performs face detection, landmark localization, pose estimation and gender recognition simultaneously while fusing intermediate layers of a deep CNN.

A common challenge in a multi-task setting is that every task has usually a separate domain specific loss function. One way to deal with multiple losses during training is to calculate a weighted linear combination of all losses. This total loss can then be optimized like in a single-task setting. However, due to e.g. different scalings in the task-specific losses weights are usually introduced. One possibility to determine optimal weights is to perform a grid search over the weights. However the search space grows exponentially with the number of tasks and therefore one quickly runs into resource limitations. More sophisticated procedures for the weight calculation are proposed by [4], [11]. The former proposes GradNorm, where an adaptive weight is recalculated at each training step based on each tasks gradient magnitude. The latter introduced uncertainty weighting which considers the homoscedastic uncertainty of each task for its weighting. Another way to train CNN based multi-task architectures was introduced by Ozan et al. [24] who showed how to use gradient-based multiobjective optimization algorithms in such a setting.

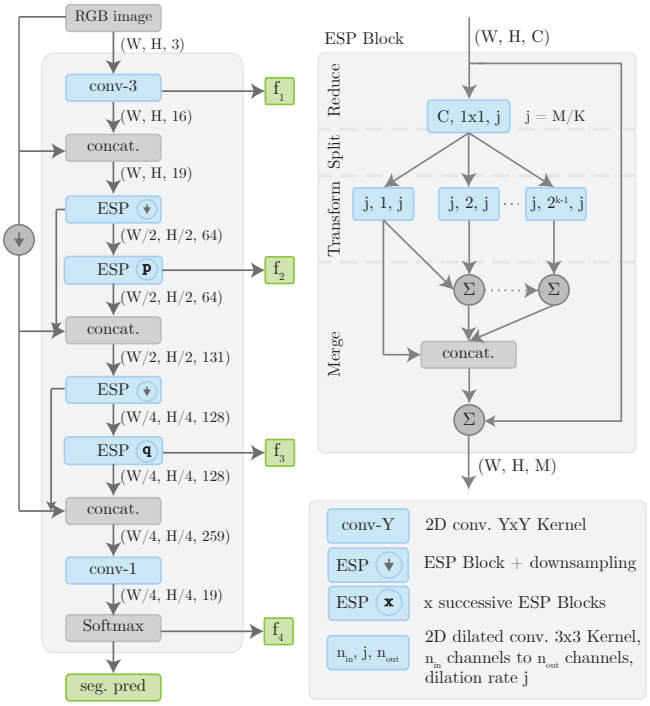


Fig. 2: ESP **feature extraction** module. Left: calculation of the feature maps f_1 to f_4 from an RGB input image. Top-right: Internal structure of one ESP block which maps from C input channels to M output channels. *Seg. pred.* and f_4 have the same content, but are drawn separately for clarity.

III. Joint Segmentation and Disparity Network

Figure 1 gives a brief overview of our proposed architecture. At first the left and right input image I_l and I_r are passed through the feature extractor which calculates for both images the feature maps f_1 to f_4 and a low-resolution semantic segmentation prediction. Weights of both feature extraction modules are shared. Secondly by using pointwise convolutions and concatenation operations new features f_l and f_r are created from the last features f_1 to f_4. Those are aggregated to a 4D cost volume which is then regularized by a 3D CNN to obtain the disparity prediction. Finally the beforehand created low-resolution segmentation prediction and the disparity prediction are passed through a common decoder which refines the segmentation and disparity predictions.

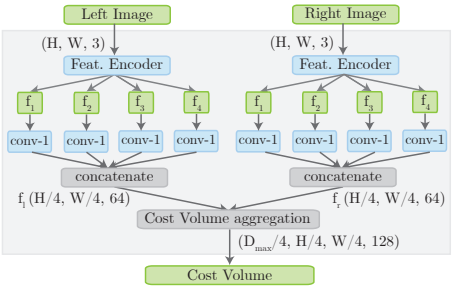


Fig. 3: Construction of the 4D **cost volume** using the left and right input image. conv-1 defines a 2D convolution with a 1x1 Kernel and 16 channels.

A. Semantic feature extraction

Figure 2 shows the structure of our semantic feature extraction module. The module simultaneously calculates a low resolution segmentation prediction, and the features f_1 to f_4 for disparity prediction. The features are extracted at different levels such that they include both high-level and low-level information. E.g. f_4 contains semantic information and f_1 contains edge information. We also use skip connections from the input image to several points in the network. This is also done in ESPNet [19] where it is referred to as input reinforcement. The main building blocks of our feature extraction module are ESP blocks [19]. The internal structure of an ESP block is shown at the top-right of Figure 2. ESP blocks are based upon the convolutional factorization principle which decomposes a standard convolution into a pointwise convolution and a spatial pyramid of dilated convolutions. This enables them to cover a wide effective receptive field while keeping the computational complexity low. A single ESP block performs the following four steps successively. At first a pointwise convolution is used to **reduce** the number of input channels from C to $j = \frac{M}{K}$, whereby K is a hyperparameter called width divider and M is the number of output channels. The resulting j channels are then **split** into K parallel branches. In the **transform** step each of the K branches applies a dilated convolution on its input, whereby for each branch the same kernel-size of $n \times n$, and a different dilation rate taken from 2^{k-1} with $k \in [1, K)$ is used. Finally, the outputs of all K branches are **merged** together. In the ESPNet paper [19] it has been shown that direct summation can introduce unwanted checkerboard or gridding artifacts. To overcome this problem hierarchical feature fusion (HFF) is used. The outputs of the different branches are first hierarchically added before they are concatenated (also shown in Figure 2).

B. Disparity Estimation

For disparity estimation we reuse the features f_1 to f_4 from the left and right input image. At first, all features are rescaled to a common resolution which is $(H/4, W/4)$ in our case. The features f_1 and f_2 have a higher resolution and thus they are downsampled. Then on each feature output $f_{\{1,2,3,4\}}$ a 1×1 convolution is applied to reduce the number of channels to 16. The resulting features are then concatenated into the feature f_l for the left image and f_r for the right image. These features contain information from multiple scales and are therefore well suited for matching. This is done by concatenating f_l together with the corresponding right feature map f_r over all disparity levels. This results in a 4D cost volume of dimensionality $(D_{max}/4, H/4, W/4, 128)$. See Figure 3 for a visualization. Note that the first dimension of the cost volume equals $D_{max}/4$ and not D_{max} since we downsample over all 3 spatial dimensions. To regularize the cost volume we fed it into a 3D CNN with a similar architecture as in GC-Net [12] and PSMNet [2]. The network consists of five consecutive residual blocks containing two 3D convolutional layers with kernels of size $3 \times 3 \times 3$ and 32 channels, followed by a single $3 \times 3 \times 3$ 3D convolutional layer with one output channel. We upsample the output of the 3D CNN from $(D_{max}/4, H/4, W/4)$ to (D_{max}, H, W). This volume contains an estimated cost for each disparity value in the first dimension. A dense disparity prediction could be acquired by e.g. the argmin operation over the first dimension. However a regular argmin operation is non-differentiable and therefore does not allow training with sub-pixel accuracy. Following previous work like GC-Net [12] we use a differentiable SoftArgmin function instead. To calculate the SoftArgmin function we first take the negative values of the cost volume C to convert the cost volume to a probability volume. Then we normalize the probability volume with a softmax across the disparity dimension. Finally, we calculate the sum over all disparity values weighted by their normalized probability. This leads to the following calculation of the SoftArgmin $\hat{d}_i$ for every pixel i:

$$\hat{d}_i = \sum_{d}^{D_{max}} d \cdot \frac{\exp\left(-C_i(d)\right)}{\sum_{d'} \exp\left(-C_i\left(d'\right)\right)}, \tag{1}$$

whereby D_{max} is the maximum disparity respectively the size of the first dimension of the cost volume C. The SoftArgmin is differentiable and thus we can learn sub-pixel accurate disparity maps.

C. Decoder

In order to allow the two tasks to benefit from each other, we propose a decoder which jointly creates the final prediction for both the semantic segmentation and the disparity map. The structure of the decoder is shown in Figure 4. In the decoder the low-resolution disparity and segmentation predictions are upsampled in two separate paths, whereby the two paths are connected at several positions to allow information to flow between the disparity and segmentation path. The inputs of the decoder are a disparity map and a segmentation prediction of size $\left(\frac{H}{4} \times \frac{W}{4}\right)$. Those are then upsampled by the decoder in three consecutive stages, whereby each stage is processing data at the increasing resolutions, i.e. $\left(\frac{H}{4} \times \frac{W}{4}\right) \rightarrow \left(\frac{H}{2} \times \frac{W}{2}\right) \rightarrow (H \times W)$. The input of each stage is either a disparity or segmentation prediction at the respective resolution of the stage, and the output is a refined version of the input. This is done with a residual connection [9] which allows to add finer grained information at the next stage without the need of completely reconstructing the result.

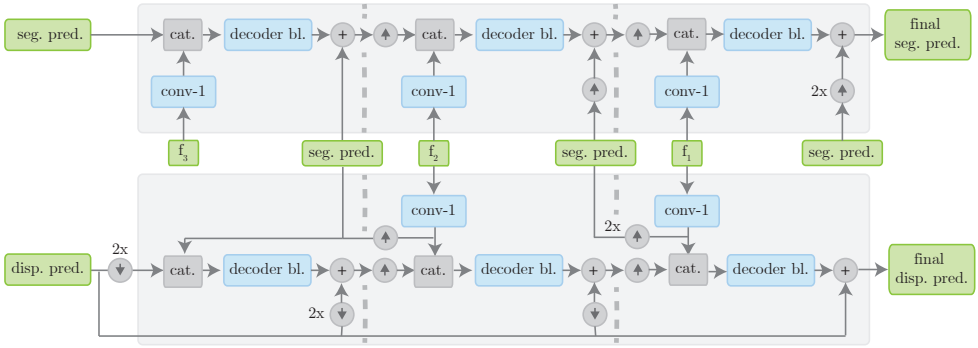


Fig. 4: The **decoder** upsamples the low-resolution segmentation prediction and refines the disparity prediction. Stages of the decoder are separated by vertical dashed lines.

Let us now look into one stage (=one processing resolution) in more detail. To provide fine details which have been lost during the former downsampling and processing we concatenate the encoded feature maps at the respective resolutions. This is similar to the skip connections between the contracting and expanding path in U-Net [23]. We squeeze the concatenated features with a 1×1 convolution to match the number of semantic classes. In the disparity path we additionally concatenate the segmentation prediction.

The result is then processed by a decoder block. The input of a decoder block is first passed through two consecutive ESP blocks with 48 output channels and then through another 1×1 convolution to reduce the output to the 19 channels for the segmentation path and to 1 channel for the disparity path, respectively. Finally, an elementwise addition between the output of the decoder block and the respective prediction of each path is calculated to get the output of the stage.

D. Multi-task loss

To train our model we use a cross-entropy loss for the segmentation task, and a smooth $L1$ loss for the disparity task. We calculate our disparity loss as follows:

$$L_d(d,\hat{d}) = \frac{1}{N_d}\sum_{i=1}^{N_d} \text{smooth}_{L_1}\left(d_i - \hat{d}_i\right), \tag{2}$$

whereby we mask out pixels with invalid disparity label and the smooth L1 loss is given as:

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases}. \tag{3}$$

In the KITTI training set the amount of pixel occurrences of each class varies substantially. To account for this imbalance we weight each class by a different weight w_{class} in the segmentation loss. The weights are calculated as follows:

$$w_{\text{class}} = \frac{1}{\ln(c + p_{\text{class}})}, \tag{4}$$

whereby p_{class} denotes the probability of class occurrence at one pixel in the training set and c is a hyperparameter. We calculate the total loss L as a weighted sum of L_s and L_d :

$$L(\theta_j, \theta_s, \theta_d) = w_s L_s(\theta_j, \theta_s) + w_d L_d(\theta_j, \theta_d), \tag{5}$$

where $\Theta \in \mathbb{R}^N$ contains all N parameters of our model, $\theta_s \subset \Theta$ contains the parameters only used for the segmentation task, $\theta_d \subset \Theta$ the parameters only used for the disparity task and $\theta_j = \theta_s \cup \theta_d$ are the jointly used parameters. The weights for the segmentation and disparity loss are denoted by w_s and w_d. We evaluate how the performance of our model changes if we set the weights w_{disp} and w_{seg} to several fixed values, and if the weights are calculated using adaptive loss balancing with GradNorm [4].

IV. Experiments

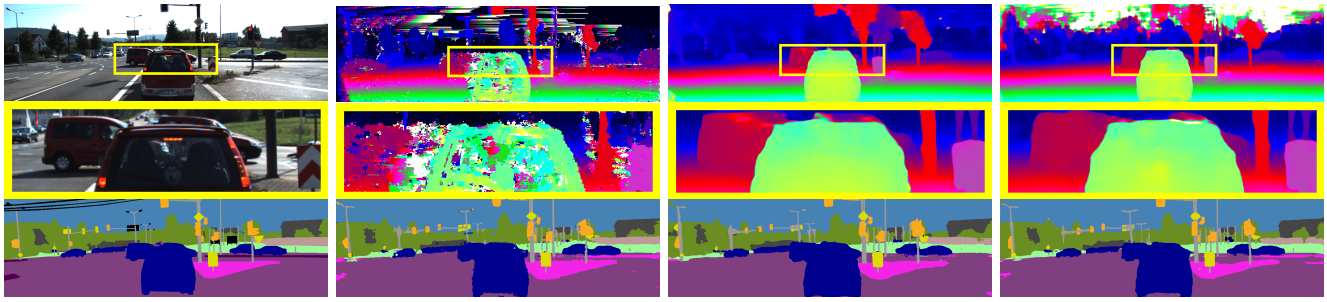
A. Data

We use the KITTI dataset [6] for training and evaluation of our model. It contains ground truth data for both semantic segmentation and disparity. The dataset consists of 200 training and 200 test images of driving scenes, whereby the labels of the 200 test images are not available to the public but hold back. This provides a reliable performance benchmark of different models on a public online leaderboard. To evaluate our model we use a 80/20 random split of the 200 training images. Following most recent work which evaluates segmentation on KITTI [27] we use the most common 19 classes for training and evaluation. To account for the class imbalance in the KITTI we set the hyperparameter c in the calculation of w_{class} to 1.02, as in ENet [20]. This restricts the class weights to be in the interval of $[1, 50]$.

B. Training

All convolutions are followed by a batch normalization layer except the last layer of the 3D regularization module, leaky ReLU units ($\alpha = 0.01$) are used as activation functions, except the 3D regularization module where ReLU units are used for activation. In the ESP feature extraction module the number of successive blocks is set to $p = 5$ and $q = 3$. The width divider K of all ESP blocks is set to 5. We use the Adam [13] optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$) with an initial learning rate of 0.0005 and decay it by a factor of 0.5 at epoch 400 and 500. The complete training is performed end-to-end and no pre-training is done. Following PSMNet [2], we set the maximum disparity to 192. For all experiments a batch size of 2 was used. All input images are normalized, using the mean and variance of the training set.

As input we use the full resolution KITTI images. However to avoid memory problems during training, we perform a 114×128 random crop on the cost volume. The memory



(e) Input/GT (f) Baseline (g) Model A (h) Model B

Fig. 5: Qualitative comparison. (a) shows the input image and the semantic ground-truth. (b-d) show the predictions of our models. Top row: Color-coded disparity maps with green=near and blue=far. Middle row: Close ups of the selected regions. Bottom row: Semantic segmentation. Parameters for *Model A* and *Model B* have been obtained from the same training session but at different epochs.

requirements of our 3D CNN are relatively high, and all recent networks with similar architectures like PSMNet [2] use cropped input images for training. With our approach the feature extraction module can exploit information from the complete input image, while we still save a significant amount of memory in the 3D CNN. It has also been shown by [26] that information from the complete input image is benefical for stereo matching. Note during inference no cropping in the 3D CNN is necessary.

C. Baseline systems

We compare our final results also to simple baseline systems. For disparity estimation we implemented a network similar to the CNN-7 model used in CNN-CRF [14] without the CRF. The network consists of a 7-layer unary CNN with 100 filters in each layer, whereby the filter size in the first layer is (3×3) and in all following layers (2×2), as activation function the *tanh* is used. The unary CNN is then used to extract features from the left and right input image which are passed through a cross-correlation function which outputs the softmax normalized scalar products of corresponding feature vectors. For baseline segmentation we use an 8 layer CNN network, whereby the first 7 layer have 64 filters and a kernel size of (3×3) followed by BatchNormalization and a ReLU activation function. We use a dilation rate of 2 for the layers 3 and 4, and a dilation rate of 4 for the layers 5, 6, 7, and 8. After the second and fourth layer we apply a pooling operation with a factor of 2. The last layer has a kernel size of (1×1) and 19 output filters.

D. Results

During training we monitor the mean Intersection-over-Union (mIoU) and the bad3 pixel error, i.e. the percentage of pixels with disparity error > 3.0 pixels. When only disparity or segmentation is trained, we report the respective score at the best performing epoch. In the multi-task setting, the epoch where one task reaches its best performance is not necessarily the same epoch where the other task performs best. Therefore we need to choose an epoch which gives a good trade-off between performance of both tasks. To give a better insight in our model, we will examine two models both obtained from the same training session ($w_{disp} = 5$ and $w_{seg} = 1$), but parameters are taken at different epochs. One model was chosen with slightly better performance in segmentation and the other in disparity estimation. We will simply refer to them as *Model A* and *Model B*. Figure 5 shows a qualitative comparison of our results. In the top-region *Model B* gives a relative noisy disparity prediction. The segmentation close-ups show that *Model B* is able to predict the car contours better than *Model A*.

Model	NOC	All	Seg.	Params	Time
	bad3	bad3	mIoU	#	sec.
ours - ESP					
Model A	3.16	3.75	47.04	0.64M	0.28
Model B	3.84	4.37	49.43	0.64M	0.28
GradNorm $\alpha = 0.5$	4.07	4.54	46.6	0.64M	0.28
GradNorm $\alpha = 1.5$	3.95	4.68	48.9	0.64M	0.28
Only disp	3.69	4.24	–	0.57M	0.28
Only seg	–	–	50.48	0.21M	0.08
ours - baseline					
Only disp	14.10	15.6	–	0.28M	–
Only seg	–	–	44.13	0.23M	–
other					
PSMNet [2]	2.14	2.32	–	5.23M	0.44
SegStereo [27]	–	2.25	59.10	25.60M[1]	0.60
CNN-CRF [14]	4.84	5.50	–	0.28M	0.76

TABLE I: Results. Note that we report performance on our validation set, and the published results report performance on the retained test set of KITTI.

Table I shows the results of our experiments. Disparity estimation is evaluated on non-occluded (NOC) pixels, i.e. pixels visible in both images, and on all pixels. We have been able to significantly improve upon our baseline systems by incorporating ESP blocks, which shows that they are an efficient building-block for light-weight segmentation and stereo matching networks. When doing multi-task training, performance in segmentation gets slightly worse, but for disparity estimation performance increases upon the single-task setting. Looking at the number of parameters of each model we note that i) all of our models are light-weight compared to most other recent work and ii) only a relative increase in

[1]SegStereo does not report an exact number of parameters for their model, therefore we report the parameters of ResNet-50 which they use as a backbone.

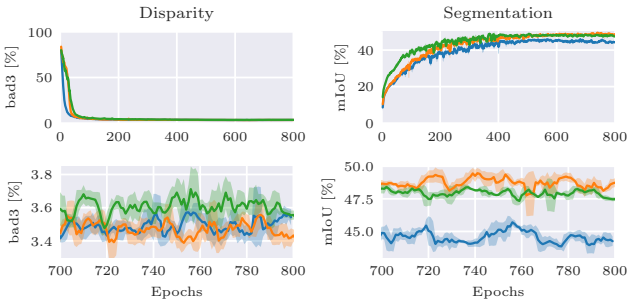


Fig. 6: Training curves, Shaded regions show standard deviation. Left: lower is better. Right: higher is better. Blue: $(w_{disp} = 10, w_{seg} = 1)$, green: $(w_{disp} = 10, w_{seg} = 1)$, orange: $(w_{disp} = 5, w_{seg} = 1)$.

parameters of 12% is needed from our model which only performs disparity estimation to the model which calculates both disparity and segmentation. Overall SegStereo and PSMNet still perform better, we mainly attribute this to the fact that they use significantly more complex models without a strong focus on efficiency. Compared to CNN-CRF we are able to achieve slightly better performance, however it should be noted that their model uses even less parameters than ours.

We also investigated how different weightings of the disparity and segmentation loss influence training. Figure 6 shows training curves for other weightings, one can see that especially the performance in segmentation degrades if its relative weighting is too small. The usage of GradNorm did not help to increase performance. The best results we achieved were obtained by using a fixed weighting of $w_{disp} = 5$ and $w_{seg} = 1$.

V. Conclusion

We have proposed a model which jointly performs semantic segmentation and stereo matching. Compared to performing both tasks separately, our shared architecture reduces runtime and model complexity, while achieving the same performance. Furthermore we carefully designed our model with a focus on efficiency and are able to reduce runtime compared to similar state-of-the art methods.

References

[1] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, 1997.

[2] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[3] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[4] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 2018.

[5] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, 2013.

[7] F. Guney and A. Geiger, "Displets: Resolving stereo ambiguities using object knowledge," in *CVPR*, 2015.

[8] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. Courville, Y. Bengio, C. Pal, P.-M. Jodoin, and H. Larochelle, "Brain tumor segmentation with deep neural networks," *Medical image analysis*, vol. 35, 2017.

[9] K. He, X. Zhang, and S. Ren, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[10] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[11] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[12] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry, "End-to-end learning of geometry and context for deep stereo regression," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[14] P. Knöbelreiter, C. Reinbacher, A. Shekhovtsov, and T. Pock, "End-to-end training of hybrid cnn-crf models for stereo," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[15] I. Kokkinos, "Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[16] L. Ladickỳ, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. H. Torr, "Joint optimization for object class segmentation and dense stereo reconstruction," *International Journal of Computer Vision*, vol. 100, no. 2, 2012.

[17] Z. Li and L. Yu, "Compare stereo patches using atrous convolutional neural networks," in *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, ser. ICMR '18. ACM, 2018.

[18] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[19] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi, "Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[20] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "Enet: A deep neural network architecture for real-time semantic segmentation," *arXiv preprint arXiv:1606.02147*, 2016.

[21] M. Pesaresi and J. A. Benediktsson, "A new approach for the morphological segmentation of high-resolution satellite imagery," *IEEE transactions on Geoscience and Remote Sensing*, vol. 39, no. 2, 2001.

[22] R. Ranjan, V. M. Patel, and R. Chellappa, "Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 1, 2019.

[23] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015.

[24] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," in *NIPS*, 2018.

[25] X. Song, X. Zhao, H. Hu, and L. Fang, "Edgestereo: A context integrated residual pyramid network for stereo matching," *arXiv preprint arXiv:1803.05196*, 2018.

[26] S. Tulyakov, A. Ivanov, and F. Fleuret, "Practical deep stereo (pds): Toward applications-friendly deep stereo matching," in *Advances in Neural Information Processing Systems*, 2018.

[27] G. Yang, H. Zhao, J. Shi, Z. Deng, and J. Jia, "Segstereo: Exploiting semantic information for disparity estimation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[28] J. Zbontar, Y. LeCun, *et al.*, "Stereo matching by training a convolutional neural network to compare image patches." *Journal of Machine Learning Research*, vol. 17, no. 1-32, 2016.

[29] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," *CVPR 2017*, vol. 2017-January, 2017.

6D Object Pose Verification via Confidence-based Monte Carlo Tree Search and Constrained Physics Simulation

Dominik Bauer[1], Timothy Patten[1], Markus Vincze[1]

***Abstract*— Precise object pose estimation is required for robots to manipulate objects in their environment. However, the quality of object pose estimation deteriorates in cluttered scenes due to occlusions and detection errors. The estimates only partially fit the observed scene, or are physically implausible. As a result, robotic grasps based on these poses may be unsuccessful and derived scene descriptions may be unintelligible for a human observer. We propose a hypotheses verification approach that detects such outliers while, at the same time, enforces physical plausibility. On one hand, this is achieved by a tight coupling of hypotheses generation with the verification stage to guide the search for a solution. On the other hand, we integrate a constrained physics simulation into the verification stage to constantly enforce physical plausibility. By constraining the simulated objects to the most confident point correspondences, we prevent the estimated poses from erroneously diverging from the initial predictions. We thereby generate a plausible description of the observed scene. We evaluate our method on the `LINEMOD` and `YCB-VIDEO` datasets, and achieve state-of-the-art performance.**

I. INTRODUCTION

For robots to autonomously operate in the real world, they require a reliable estimate of the pose of objects around them to be able to manipulate objects of interest [17]. In human-robot interaction, it is furthermore essential for the robot to be able to explain why and how its estimates are computed. Thereby, the robot's actions become understandable to the human interaction partner as to build and maintain trust and transparency [3]. Such explanations help to make the robot's belief state and decisions understandable and the human interaction partner more tolerant to observed errors.

The recent SIXD challenge [10] shows advances in object pose estimation; but also current limitations. Pose estimation methods begin to fail when the objects of interest have little textural information, the robot perceives noisy RGB-D data, or the scene exposes a high amount of clutter. Due to detection errors, such as confusing different objects in the observation, the estimated poses can be far-off the actual object pose. Furthermore, the precision of the estimated poses can be low even matching the correct object. Besides falsely matching to parts of an object, not considering global scene consistency is a source for inaccuracies, resulting in estimates that may feature intersecting or floating objects. The computed scene descriptions are, therefore, potentially inconsistent on a global level and physically implausible. In object manipulation, for example, this may result in failed grasps.

Fig. 1: The initial pose hypothesis with the object half-way below the ground plane (left) and the refined pose after the verification stage with the object resting on the plane (right).

Existing hypotheses verification frameworks [2], [1], [21], [14], [15] aim to reduce inconsistencies in the scene description by reasoning about the scene on a global level, i.e., considering all hypotheses at once. Verification is considered a post-processing step in these frameworks, ignoring most of the information gathered during hypotheses generation. These methods construct complex cost functions [2], [1] for which parameters must be tuned for a specific dataset. Creating an exhaustive set of hypotheses [14] or performing search exhaustively [15] are alternative approaches to increase performance in the absence of discriminative information; albeit significantly increasing computation time.

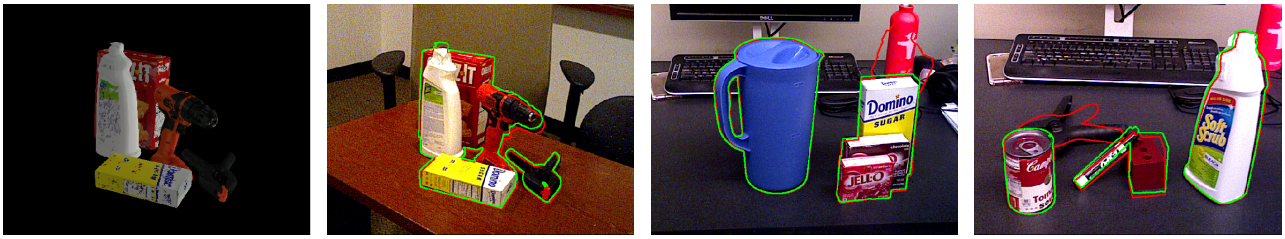
During hypotheses generation, information beyond the pose is generated that enables the discrimination between hypotheses. For example, the ranking, confidence or fitness of a pose hypothesis can be used to consider more promising candidates first. Also, the position and fitness of individual point correspondences with the observation contain information that can reduce the solution space the hypotheses verification has to consider.

To that end, we propose a two-staged framework that closely couples hypotheses generation and verification to leverage available information. We incorporate physics simulation into the verification stage similar to [14] but impose soft-constraints to limit the objects' movements to remain close to the original estimate. This is done by scaling the forces of the point constraints by the confidence of point correspondences. The verification is guided by a heuristic that uses the confidence of the generated hypotheses. The result is a subset of pose hypotheses that best explains the observation on a global scene level.

We propose a hypotheses verification framework based on Monte Carlo tree search (MCTS). By the use of physics simulation during the simulation phases of the tree search,

*Funded by the TU Wien Doctoral College TrustRobots and partially funded by OMRON Corporation.

[1]The authors are with Faculty of Electrical Engineering and Information Technology, Institute of Automation and Control, TU Wien, 1040 Vienna, Austria {bauer,patten,vincze}@acin.tuwien.ac.at



(a) Rendered view of the scene in (b). Note the cap on the model of the bleach bottle that is missing in (b) and (c).

(b) By enforcing the drill to stand on the table, we achieve a plausible and more precise pose.

(c) The bottle in the back is falsely matched to the drill during HG but detected as false positive during HV.

(d) Error case: The clamp is falsely omitted. The predicted foam brick (second from right) moves the pen resting on it.

Fig. 2: Qualitative examples. The red contours show the output of our HG, green contours show the solution after HV.

solutions are generated that are inherently physically plausible. We show improvement over previous work by:

- using confidence-based heuristic to guide verification
- integrating physics simulation, constrained by point correspondences, with hypotheses verification
- detecting false positives that are far-off from the true object pose while refining near-by hypotheses in a physically plausible manner
- achieving state-of-the-art performance on the LINEMOD and YCB-VIDEO datasets.

In the following, we discuss related work in Section II, give a detailed description of our proposed approach in Section III, and present results on LINEMOD and YCB-VIDEO using the ADD, ADD-S and VSD metrics in Section IV. A discussion of the improvements and limitations can be found in Section V.

II. RELATED WORK

A. Object Pose Estimation

Hodan et al. [10] compare several of state-of-the-art methods in object pose estimation and identify four major research directions. Template-based methods [8], [11] precompute different views of the objects of interest observed under a discrete set of rotation angles. During inference, the task is to first locate known objects in the observation and then find the corresponding view to deduce the object rotation. Methods based on Point Pair Features (PPF) [7], [18] are trained by precomputing these features for the objects of interest and storing them in a discretized hash table that is used to match scene points to model points via a voting scheme during pose estimation. For every two model points, such a feature consists of their distance, the angle between their normals and the angle between each normal and the connecting line. Methods based on 3D local descriptors, such as SHOT [16] or PPFH [6], find correspondences between features computed on a model and the observed scene to generate pose candidates that are then refined using ICP.

Finally, several learning-based methods [20], [12], [19] tackle pose estimation using CNNs. Xiang et al. [20] propose an architecture that jointly estimates semantic labels, translation and rotation from RGB images. Li et al. [12] furthermore include depth information in their architecture. Building on this work, Wang et al. [19] propose to fuse appearance features predicted from RGB with geometric features predicted from the depth image into a single pixel-wise embedding. Based on these pixel-wise features, a candidate pose is predicted per pixel. The highest scoring prediction per object instance mask is returned as a pose for this object. We base our hypotheses generation on this work.

B. Hypotheses Verification

Narayanan et al. [15] approach the pose estimation problem by generating an exhaustive set of possible scene configurations and then searching for the best solution in a verification scheme. Each scene configuration is rendered and the resulting depth image is compared to the observed depth image. While the resulting estimates have high accuracy, the authors also report an average runtime of 6.5 minutes.

Aldoma et al. [1] use SHOT to generate a set of pose hypotheses. The authors use Simulated Annealing to search for the subset of hypotheses that best fits the observation on a scene level. To quantify the fitness of a candidate solution, a cost function is constructed based on geometric cues. Additional terms penalize multiple assignments of scene points to different objects and a so-called "clutter term" penalizes hypotheses that only partly fit smooth surface patches. In follow up work [2], the cost function is extended by considering color information and a comparison of different meta-heuristics for finding a solution. Our solution definition is closely related to the formulation presented in this work.

Most related to our hypotheses verification method, Mitash et al. [14] propose the use of MCTS for hypotheses verification and the use of physics simulation to enforce scene consistency. The authors utilize the basic UCT algorithm for MCTS and evaluate their method on a non-public dataset.

III. OBJECT POSE VERIFICATION

Given a RGB-D image as input, object-instance masks and their corresponding class labels are estimated using SegNet [4]. To integrate the information from hypotheses generation (HG) with hypotheses verification (HV), we require a HG method to provide a set of diverse pose hypotheses, information on their quality as well as a set of correspondences that allow us to constrain the object pose in physics simulation. We use DenseFusion, proposed by Wang et al. [19], to

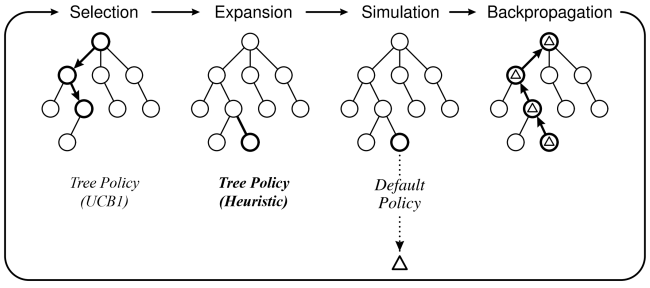


Fig. 3: Overview of the MCTS algorithm. For our method we adapt the tree policy during expansion (bold) and the computation of the reward (triangle). Adapted from [5].

generate such hypotheses; although our approach is able to deal with other HG methods as long as they can provide the required information just mentioned. For example, the PPF-based method by Vidal et al. [18] could be used instead.

The HV stage is tasked with determining the set of hypotheses that globally best explain the observed scene. For this, we initialize MCTS with the generated pose hypotheses. Starting from the empty set, MCTS adds one hypothesis per expansion and evaluates the new solution set by simulating the predicted scene. To determine which hypothesis should be selected next for expansion, we propose a heuristic based on the hypotheses' confidence values. MCTS is stopped after a fixed number of iterations to limit total runtime and the highest-reward solution is returned.

In the following, we describe our approach in more detail.

A. Hypotheses Generation

For each object-instance mask, DenseFusion predicts a pose hypothesis for each corresponding scene point and provides a confidence value for the respective estimate. The highest-confidence pose hypothesis is refined and returned as a pose estimate for this object instance. To generate a set pose hypotheses with more variation, we trigger HG repeatedly with different random samples of the observed object instance. This variation is desired as to generate a set of hypotheses that has a higher chance of containing the true object pose that should be found by HV. As a measure of quality for each pose hypothesis, the confidence value that is returned by the network is used. Finally, as each sampled point predicts a pose hypothesis and reports a confidence value, the 100 most confident hypotheses are selected as soft-constraints that should be satisfied during physics simulation in HV.

B. Hypotheses Verification

The task of HV is to select a subset of hypotheses that best explains the scene. A solution X can be described by $n \cdot N$ binary variables, where n is the number of hypotheses and N the number of object instances. Each $x \in X$ takes a value of 1 if the corresponding hypothesis is used in the solution and 0 otherwise. Exhaustively searching through all possible permutations quickly becomes intractable with increasing number of objects and hypotheses per object. For example, the `YCB-VIDEO` dataset contains scenes with 3 to 9 objects per scene. For 5 hypotheses per object and no further limitations on the solution there are 2^{15} to 2^{45} possible solutions.

1) Monte Carlo tree search: MCTS is an algorithm that is successfully used in game-playing to solve similar tasks with large search spaces and potentially costly state evaluations. An illustration of the basic algorithm is shown in Figure 3. The initial step in each iteration is to *select* the most promising candidate solution for which not all direct children have been expanded. Using the UCB1 policy, the algorithm balances the exploitation of known rewarding solutions with the exploration of new regions of the search tree. Once this node is selected, a new child solution is created and the tree is *expanded*. In the basic version, also used in previous work [14], this new child solution is chosen randomly. We propose to, instead, use a heuristic to expand more promising candidates first. The next step in the algorithm is to evaluate the expected reward of the new solution by *simulating* a rollout. In a rollout, the child solution is expanded using the default policy until a terminal state is reached. The reward for this terminal state is computed and *backpropagated*, starting from the new child solution and following the selection path up to the root node. Thereby, the reward statistics for each node can be updated for use by the UCB1 policy in the next iteration. The search can be stopped any time, offering speed to be traded for solution quality.

2) Solution space: Assuming there can only be one true hypothesis per object, we can reduce the number of possible solutions for the previous example of 3 object instances with 5 hypotheses each from 2^{15} to 915 possibilities. We model the problem as follows: Starting with the empty set as an initial candidate solution, we iteratively activate hypotheses. Activating more than one hypothesis per object instance is not allowed. The hypothesis for an object instance can therefore not change once it has been activated. Other subtrees of the search tree evaluate these alternative hypotheses.

3) Heuristic tree policy: Evaluation individual solutions is nevertheless costly. Guiding the search towards promising regions first allows the search to be stopped early, which saves computation time. We propose to leverage the confidence values of hypotheses compute during HG for this task. Instead of randomly selecting the next hypothesis, the probability of a hypothesis to be chosen is weighted by its confidence value. In a first step, one of the currently inactive object instances is chosen with a probability proportional to the sum of the confidence values of their inactive hypotheses. Next, one of the currently inactive hypotheses of this object instance is randomly chosen, again weighted by their confidence value. As a result, the search is guided towards the more promising objects and their hypotheses first.

4) Physics simulation: To determine the reward of a new candidate solution, we apply a physics simulation, render the object models in the resulting poses and compare the rendered depth values with the observed ones. However, naïvely applying physics simulation to a candidate solution has two main problems: First, the unknown mass distribution of the objects may cause unexpected ways. Without, for

example, considering the heavy battery pack of a power drill, the object easily topples over in simulation. Second, as the solution may include intersecting objects, the resulting repulsion forces can push close-by objects away or over.

The center of mass of the objects is moved close to their bottom, with respect to their canonical pose, to tackle the problem of unknown mass distribution. By computing a convex decomposition of the object meshes used as colliders, the effect of intersecting objects is lessened. In addition, soft-constraining the movement of the objects to their highest-confidence point correspondences limits the effect of the repulsion forces. Per point and confidence value, we set a constrain between the point on the object and its predicted position in world space. Each constrain can apply a maximal force that is proportional to its confidence value. The constrained physics simulation is able to generate plausible poses while circumventing the problem of exaggerating the instability resulting from estimation inaccuracy.

5) Reward function: After the physics simulation, the final step of evaluating a candidate solution consists of rendering the corresponding depth image and comparing it to the observed depth. We mask-out all points that have neither a depth value in the rendering nor the observation. The reward r is then computed as follows:

$$r = \sum \delta(R,S) \tag{1}$$

$$\delta_1(R,S) = \begin{cases} 1, & \text{if } |d_R - d_S| < \tau \text{ for } d_R, d_S \in R,S \\ -1, & \text{otherwise} \end{cases}$$
$$\delta_2(R,S) = \begin{cases} 1 - \frac{|d_R - d_S|}{\tau}, & \text{if } |d_R - d_S| < \tau \text{ for } d_R, d_S \in R,S \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where R is the masked rendered depth and S the masked observed depth. We use δ_1 with $\tau = 1cm$ on LINEMOD and δ_2 with $\tau = 3cm$ on YCB-VIDEO.

IV. RESULTS

For our evaluation, we want to compare the proposed HV pipeline (*Ours*) against two state-of-the-art pose estimation methods, the baseline method DenseFusion [19] (*Baseline*) and *PoseCNN* [20]. To ensure comparability with the two methods, we use the trained weights from [19] and the pre-computed segmentation masks provided by [20] for evaluation on the YCB-VIDEO dataset. Both methods also provide results for the LINEMOD dataset. However, the classes 3 and 7 of LINEMOD are not included in the pre-trained network and are thus missing from our evaluation. The evaluation uses the same test set as Wang et al. [19].

On the LINEMOD dataset, we evaluate the non-symmetric objects on the ADD [8], the symmetric objects on the ADD-S [20] metrics and all objects the VSD metric [10], [9]. The results on the ADD(-S) metric are reported in terms of recall score for a threshold of 0.1 times the object's diameter ($\leq 0.1d$). For the VSD metric, the threshold is set to 0.3. On the YCB-VIDEO dataset, all objects are evaluated on the ADD-S metric. We report the Area under Curve (AUC) as well as the recall scores for thresholds of $2cm$ and $1cm$. Qualitative results on this dataset are shown in Figure 2. Following Wang et al. [19], we compare against PoseCNN with DeepIM [20], [13] (*PoseCNN + DeepIM*) on LINEMOD and against PoseCNN with ICP (*PoseCNN + ICP*) on YCB-VIDEO.

A. Results on LINEMOD

For the LINEMOD dataset, we are able to report a significantly increased performance over the baseline method on both the ADD(-S) and the VSD metrics. Since the LINEMOD dataset features only one target object per scene, the results only show the performance of the constrained physics simulation and reward function. The MCTS has no impact on the solution as it will exhaustively try all solutions. We set the number of hypotheses per object instance to 10 and the threshold τ in our reward function to 1cm. The HG takes approximately 20ms per pose hypothesis and the HV takes approximately 60ms to evaluate a candidate solution. The total runtime, including the segmentation stage and additional processing, is 1–2s per test target.

Table I shows the results on the LINEMOD dataset for the ADD(-S) metric in the left sub-table and for the VSD metric in the right sub-table. Note that we do not report values for classes 3 and 7 as our HG does not include them.

B. Results on YCB-VIDEO

The YCB-VIDEO dataset features 3 to 9 objects per scene. In the test set used in [19] and [20], there are 3 to 6 objects per scene. Moreover, the pre-computed segmentation masks contain misdetected object instances. This allows us to evaluate the performance of our MCTS-based HV as false positives have to be omitted and different pose hypotheses may influence one-another.

Table II shows the results on the YCB-VIDEO dataset for the ADD-S metric. The results for *PoseCNN + ICP* are re-computed from the provided result poses [20]. The results for *Baseline* are taken from the same run as *Ours*. Overall, we report a slight increase of our HV method over previous work. As the results with a threshold of $2cm$ are already saturated for many classes, we also use a stricter threshold of $1cm$, showing an improvement of up to 3.3% on individual classes over the baseline. For the AUC measure, in accordance with [19], we set the highest threshold on the ADD-S to 10cm. The results on this measure show a slight improvement of *Ours* over the baseline of up to 0.5%. However, *PoseCNN + ICP* is the best performing method on several classes in terms of AUC.

The number of hypotheses per object instance is set to 5, the threshold τ in our reward function to 3cm and the number of iterations in the MCTS was limited to 300. This means that, out of all possible combinations of hypotheses for the 3 to 6 objects in the test set, at most 300 solutions are evaluated. We motivate this choice as follows: In the worst case of 6 objects with 5 hypotheses each, we spend 30 iterations on the initial expansion and need at least 105

TABLE I: Results on LINEMOD. Symmetric objects in italics, best results are highlighted bold. The difference between *Baseline* and *Ours* is given in parentheses. Left: Recall scores in percent for the ADD-S metric [20] for symmetric objects, ADD metric [8] otherwise. Right: Recall scores in percent for the VSD metric ([10], [9]) computed with $\tau = 0.02$, $\delta = 0.015$ and $\theta = 0.3$.

Class	PoseCNN+DeepIM [20], [13] $\leq 0.1d$	Baseline $\leq 0.1d$	Ours $\leq 0.1d$	Baseline $\theta < 0.3$	Ours $\theta < 0.3$
01 ape	77.0	91.3	**94.1** (+2.8)	96.5	**97.7** (+1.2)
02 vise	**97.5**	92.1	95.6 (+3.5)	83.4	**90.9** (+7.5)
04 camera	93.5	92.9	**97.1** (+4.2)	91.8	**96.2** (+4.4)
05 can	**96.5**	92.8	95.8 (+3.0)	90.2	**94.5** (+4.3)
06 cat	82.1	96.1	**97.1** (+1.0)	94.7	**97.0** (+2.3)
08 driller	**95.0**	87.4	90.6 (+3.2)	80.0	**87.3** (+7.3)
09 duck	77.7	91.6	**95.4** (+3.8)	97.5	**99.0** (+1.5)
10 eggbox	97.1	99.7	**99.7** (–)	81.7	**93.0** (+11.3)
11 glue	99.4	**100.0**	**100.0** (–)	86.5	**92.0** (+5.5)
12 puncher	52.8	89.9	**96.0** (+6.1)	90.0	**95.4** (+5.4)
13 iron	98.3	96.1	**98.5** (+2.4)	92.7	**97.0** (+4.3)
14 lamp	**97.5**	95.5	97.4 (+1.9)	89.9	**93.0** (+3.1)
15 phone	87.7	90.8	**95.4** (+4.6)	84.5	**91.7** (+7.2)
Overall	88.6	93.6	**96.4** (+2.8)	89.2	**94.2** (+5.0)

iterations to get to a full candidate solution of 6 hypotheses. To allow for different candidate solutions to be considered, and to add some safety in expectation of additional false positives while still keeping computation time reasonable, we increase this to a maximum number to 300 iterations. The hypothesis generation takes 20ms per pose hypothesis. The runtime per MCTS iteration including evaluation of the candidate solution is 60ms on average, resulting in a total runtime of about 18s.

Although we choose a tighter threshold than previous work [19], the performance on the dataset is saturated for some classes – a stricter metric would be required to further differentiate between methods. This is due to the ADD-S metric ignoring rotation errors around symmetry axes.

Another aspect that we observed is that the solution after the verification can be worse than simply using the first generated hypothesis per object. On one hand, this is due to the limited number of 300 solutions we explore during the verification. The search may, as a result, only return a local minimum. On the other hand, we rather strictly evaluate candidate hypotheses. The resulting reward seems to underestimate the quality of certain hypotheses or to aggressively classify them as false positive. To lessen this effect, we augment the final solution with high-confidence hypotheses after the search is stopped.

V. DISCUSSION

As discussed in Section IV, the performance on the LINEMOD and YCB-VIDEO datasets is already saturated. We plan to evaluate our method on more challenging datasets with, for example, heavier occlusion or texture-less objects. Stricter metrics and thresholds may also further differentiate between methods. For our goal of achieving a physically plausible scene description, a stricter metric could penalize intersecting objects or missing support relations.

However, a general problem when using physics simulation is the need for knowing the direction of gravity and, ideally, the supporting plane and all supporting objects. Without gravity, we can only resolve intersections. Without support, objects simply fall to the ground. How these cases can be handled would be a route for future improvements to the presented method. For example, further constraining object movement based on the observed evidence could remove the need to know all interacting objects a-priori.

The performance of our HV method is currently limited by the amount of solutions we can consider. The results on the YCB-VIDEO dataset, presented in Section IV-A, illustrate this problem: Since we only allow for 300 iterations, MCTS might not be able to explore better performing areas of the tree. A suboptimal candidate solution is returned in such cases. This problem could be tackled by constraining the search space further or constructing a more informative reward function. However, the main limiting factor is the runtime of an individual MCTS iteration.

In our current implementation, one iteration takes approximately 60ms. While this is an improvement over the runtime of 200ms per iteration reported by [14], it is still restrictive for use in a robotic application. The main part of these 60ms are due to the physics simulation with 30ms and the read-back of the rendered image with 20ms. Especially the read-back could be accelerated by computing the reward on the GPU and returning one value instead of rendered images.

VI. CONCLUSION

We presented a hypotheses verification method for 6D object pose estimation. Our approach is based on closely integrating information from hypotheses generation with verification. The confidence values per hypotheses are used to guide verification while the confidence values and positions of point correspondences are used to constrain physics simulation. This physics simulation is an integral step of our verification scheme that uses Monte Carlo tree search to find a solution that fits the observation in a globally consistent and plausible manner. Using our proposed hypotheses verification method, we are able to achieve state-of-the-art performance on both the LINEMOD and the YCB-VIDEO dataset.

TABLE II: Results on YCB−VIDEO (ADD-S [20]). Symmetric objects in italics, best results are highlighted bold. The difference between *Baseline* and *Ours* is given in parentheses.

Class	PoseCNN + ICP [20]			Baseline			Ours		
	AUC	$< 2cm$	$< 1cm$	AUC	$< 2cm$	$< 1cm$	AUC	$< 2cm$	$< 1cm$
002_master_chef_can	95.8	**100.0**	99.5	96.4	**100.0**	**100.0**	**96.5** (+0.1)	**100.0** (–)	**100.0** (–)
003_cracker_box	92.7	91.6	84.8	95.8	99.4	96.7	**96.3** (+0.5)	**99.7** (+0.3)	**98.6** (+1.9)
004_sugar_box	**98.2**	**100.0**	**100.0**	97.6	**100.0**	**100.0**	97.7 (+0.1)	**100.0** (–)	**100.0** (–)
005_tomato_soup_can	**94.5**	**96.9**	96.8	**94.5**	**96.9**	96.8	**94.5** (–)	**96.9** (–)	**96.9** (+0.1)
006_mustard_bottle	**98.6**	**100.0**	98.9	97.3	**100.0**	97.8	97.6 (+0.3)	**100.0** (–)	**99.4** (+1.6)
007_tuna_fish_can	97.1	**100.0**	97.7	97.1	**100.0**	99.4	**97.2** (+0.1)	**100.0** (–)	**99.7** (+0.3)
008_pudding_box	**97.9**	**100.0**	**100.0**	95.9	99.5	98.6	96.2 (+0.3)	99.5 (–)	99.1 (+0.5)
009_gelatin_box	**98.8**	**100.0**	**100.0**	98.0	**100.0**	**100.0**	98.1 (+0.1)	**100.0** (–)	**100.0** (–)
010_potted_meat_can	**92.7**	**93.6**	83.3	90.7	92.8	87.1	90.3 (–0.4)	92.4 (–0.4)	**87.3** (+0.2)
011_banana	**97.1**	99.7	95.0	96.2	99.7	98.4	96.7 (+0.5)	**100.0** (+0.3)	**99.7** (+1.3)
019_pitcher_base	**97.8**	**100.0**	99.7	97.5	**100.0**	99.7	97.7 (+0.2)	**100.0** (–)	**100.0** (+0.3)
021_bleach_cleanser	**96.9**	99.4	95.1	95.9	99.9	99.1	96.2 (+0.3)	**100.0** (+0.1)	**99.7** (+0.6)
024_bowl	81.0	54.9	42.9	89.4	94.6	55.4	**89.6** (+0.2)	**96.1** (+1.5)	**55.7** (+0.3)
025_mug	95.0	99.8	97.6	96.7	**100.0**	98.1	**96.8** (+0.1)	**100.0** (–)	**99.1** (+1.0)
035_power_drill	**98.2**	**99.6**	**99.3**	96.0	99.0	97.4	96.5 (+0.5)	**99.6** (+0.6)	98.7 (+1.3)
036_wood_block	87.6	80.2	74.4	92.8	**100.0**	88.0	**93.3** (+0.5)	**100.0** (–)	**91.3** (+3.3)
037_scissors	91.7	95.6	68.0	92.0	**100.0**	71.8	**92.4** (+0.4)	**100.0** (–)	**73.5** (+1.7)
040_large_marker	97.2	99.7	97.1	97.6	**100.0**	**100.0**	**97.7** (+0.1)	**100.0** (–)	**100.0** (–)
051_large_clamp	**75.2**	74.9	**67.3**	72.5	**78.7**	33.3	72.5 (–)	**78.7** (–)	33.7 (+0.4)
052_extra_large_clamp	64.4	48.8	**38.4**	**69.9**	74.8	17.2	**69.9** (–)	**75.1** (+0.3)	17.2 (–)
061_foam_brick	**97.2**	**100.0**	99.7	92.0	**100.0**	99.7	92.0 (–)	**100.0** (–)	**100.0** (+0.3)
Overall	93.0	93.2	**89.6**	93.2	96.7	88.9	**93.3** (+0.1)	**96.8** (+0.1)	89.5 (+0.6)

REFERENCES

[1] A. Aldoma, F. Tombari, L. Di Stefano, and M. Vincze, "A global hypotheses verification method for 3d object recognition," in *European conference on computer vision*. Springer, 2012, pp. 511–524.

[2] ——, "A global hypothesis verification framework for 3d object recognition in clutter," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 7, pp. 1383–1396, 2016.

[3] P. Andras, L. Esterle, M. Guckert, T. A. Han, P. R. Lewis, K. Milanovic, T. Payne, C. Perret, J. Pitt, S. T. Powers, *et al.*, "Trusting intelligent machines: Deepening trust within socio-technical systems," *IEEE Technology and Society Magazine*, vol. 37, no. 4, pp. 76–83, 2018.

[4] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[5] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A survey of monte carlo tree search methods," *IEEE Transactions on Computational Intelligence and AI in games*, vol. 4, no. 1, pp. 1–43, 2012.

[6] A. G. Buch, D. Kraft, and D. Odense, "Local point pair feature histogram for accurate 3d matching," in *Proceedings of British Machine Vision Conference*, 2018.

[7] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: Efficient and robust 3d object recognition," in *Proceedins of IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 998–1005.

[8] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Proceedings of Asian Conference on Computer Vision*, 2012, pp. 548–562.

[9] T. Hodaň, J. Matas, and Š. Obdržálek, "On evaluation of 6d object pose estimation," in *Proceedings of European Conference on Computer Vision*, 2016, pp. 606–619.

[10] T. Hodan, F. Michel, E. Brachmann, W. Kehl, A. GlentBuch, D. Kraft, B. Drost, J. Vidal, S. Ihrke, X. Zabulis, *et al.*, "Bop: benchmark for 6d object pose estimation," in *Proceedings of European Conference on Computer Vision*, 2018, pp. 19–34.

[11] T. Hodaň, X. Zabulis, M. Lourakis, Š. Obdržálek, and J. Matas, "Detection and fine 3d pose estimation of texture-less objects in rgb-d images," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015, pp. 4421–4428.

[12] C. Li, J. Bai, and G. D. Hager, "A unified framework for multi-view multi-class object pose estimation," in *Proceedings of European Conference on Computer Vision*, 2018, pp. 254–269.

[13] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "Deepim: Deep iterative matching for 6d pose estimation," in *Proceedings of European Conference on Computer Vision*, 2018, pp. 683–698.

[14] C. Mitash, A. Boularias, and K. E. Bekris, "Improving 6d pose estimation of objects in clutter via physics-aware monte carlo tree search," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2018, pp. 1–8.

[15] V. Narayanan and M. Likhachev, "Perch: Perception via search for multi-object recognition and localization," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2016, pp. 5052–5059.

[16] S. Salti, F. Tombari, and L. Di Stefano, "Shot: Unique signatures of histograms for surface and texture description," *Computer Vision and Image Understanding*, vol. 125, pp. 251–264, 2014.

[17] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield, "Deep object pose estimation for semantic robotic grasping of household objects," in *Proceedings of Conference on Robot Learning*, 2018, pp. 306–316.

[18] J. Vidal, C.-Y. Lin, and R. Martí, "6d pose estimation using an improved method based on point pair features," in *Proceedings of International Conference on Control, Automation and Robotics*, 2018, pp. 405–409.

[19] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "Densefusion: 6d object pose estimation by iterative dense fusion," *arXiv preprint arXiv:1901.04780*, 2019.

[20] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes," *arXiv preprint arXiv:1711.00199*, 2017.

[21] W. Zhou, C. Ma, and A. Kuijper, "Hough-space-based hypothesis generation and hypothesis verification for 3d object recognition and 6d pose estimation," *Computers & Graphics*, vol. 72, pp. 122–134, 2018.

Quantile Filters for Multivariate Images

Martin Welk[1]

Abstract— **Median filtering is known as a simple and robust procedure for denoising and aggregation of data. Its generalisation to arbitrary quantiles is straightforward, yielding a class of robust (rank-order) filters for univariate data. Motivated by earlier work from image processing on generalisations of median filtering to multivariate images, we study in this paper possible quantile filtering procedures for multivariate images. Discussions of multivariate quantile generalisations in the statistics literature suggest that the position parameter of a multivariate quantile should not be chosen from an interval as in the univariate case but from a unit ball in data space. This allows to derive multivariate quantile definitions from multivariate median concepts. We investigate quantile counterparts of several multivariate medians and explore their properties under the aspect of possible use as robust image filters.**

I. INTRODUCTION

Filters for multivariate (such as colour) images are often designed as generalisations of well-known filters for scalar images. When following this approach, it is important to thoroughly analyse the essential properties of the underlying univariate concepts and to choose an appropriate generalisation that retains those properties which are crucial for the application to images. This work is part of a theoretical effort to devise adequate multivariate generalisations of robust and efficient image filters based on statistical measures.

In this paper, we will first recall multivariate median concepts in $\mathbb{R}^n$, with emphasis on $\mathbb{R}^2$. We will then discuss the principal idea of a multivariate quantile and, on this basis, quantile concepts associated to some multivariate median concepts. We will compare basic properties of these quantile filters with the possible application for the filtering of multivariate images as goal.

This work presented here is driven by the motivation to systematically explore possible filters for multivariate images and to close gaps in the toolbox of fundamental image filters. At the current stage, this is therefore a mainly theoretical contribution, the practical application potential of which is to be investigated further by future efforts.

II. UNIVARIATE RANK-ORDER FILTERS

In this section, we recall basic concepts about rank-order filters for univariate (grey-value) signals and images. The median filter has been introduced for the processing of univariate signals by Tukey [17]. For an image, median filtering proceeds by shifting a sliding window (e.g. a 3×3 square patch) across the image. At each pixel position, the median of the given grey-values within the window is computed, and it becomes the grey-value of the central pixel in the filtered image. This filter can be iterated. Despite its simplicity, the iterated median filter has remarkable properties: It is capable of denoising images degraded by certain types of heavy-tailed noise (e.g. salt-and-pepper noise or impulse noise with uniform distribution of noise values within some interval). At the same time, it preserves sharp edges. Edges can be dislocated, however; on one hand, their exact localisation in the filtered image may be influenced by noise details of the original image; on the other hand, edges tend to be straightened after several iterations of median filtering. Provided that the filtering window is large enough, a characteristic rounding of corners is observed as a result of iterative median filtering.

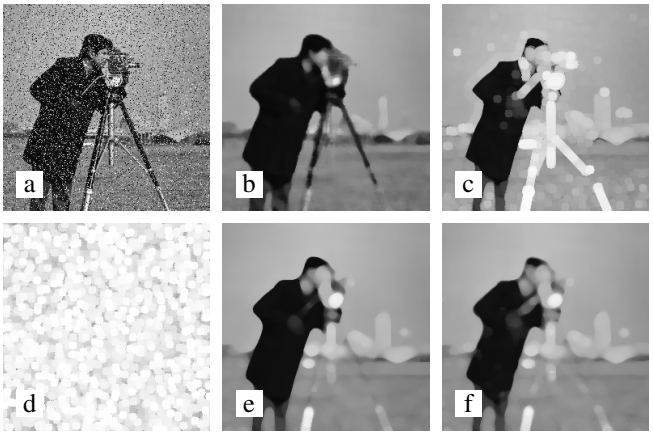


Fig. 1. Univariate rank-order filtering. **(a)** Test image *cameraman* degraded by impulse noise: 20% of the pixels have been replaced with random values uniformly distributed in $[0,255]$. – **(b)** Median filtering of (a), 1 iteration. – **(c)** Morphological dilation of the original *cameraman* image. – **(d)** Morphological dilation of the noisy image (a). – **(e)** 0.75-quantile filter applied to the original *cameraman* image. – **(f)** Same quantile filter applied to the noisy image (a).

The edge-preserving denoising and corner-rounding effect of median filtering is demonstrated in Figure 1 (a) and (b). For the median filter, we have used a disc-shaped sliding window which includes all pixels with Euclidean distance less or equal to 5 from the central pixel. We will use the same sliding window in all further image filtering experiments in this paper.

Due to its simplicity, its favourable denoising capabilities and edge-preserving behaviour, the median filter continues to be an indispensable tool in signal and image processing to date.

Whereas the median filter is initially designed in the dis-

*This work was not supported by any organization

[1]Private University of Health Sciences, Medical Informatics and Technology, Eduard-Wallnöfer-Zentrum 1, 6060 Hall/Tyrol, Austria martin.welk@umit.at

crete domain, it is straightforward to define median filtering of continuous signals or images: The finite collection of discrete values within a window is then replaced by the density of values within a continuous window, i.e. a compact neighbourhood of the location to be filtered. The median of a density is a well-defined quantity. We will, however, not discuss the continuous setting further in this paper.

As the median of univariate data is their $1/2$-quantile, an obvious generalisation of the median filter is an α-quantile filter in which the filtered value is the α-quantile of the values selected by the sliding window, with prescribed $\alpha \in (0,1)$. One can also link the fundamental morphological operations [13] of dilation (taking the maximum of values) and erosion (taking the minimum of values within the window) to this concept by considering the limit cases $\alpha \to 1$ and $\alpha \to 0$.

At first glance, an α-quantile filter with $\alpha \neq 1/2$ can be understood as a somewhat biased modification of the median filter. One should, however, be aware that an iterated quantile filter for any $\alpha > 1/2$ will in the long run converge (pointwise) to a homogeneous image with the maximum intensity of the original image; for $\alpha < 1/2$ it will converge to the minimum intensity instead. From a practical image filtering perspective, α-quantile filters for $\alpha \neq 1/2$ are therefore robust alternatives to morphological operators as they combine denoising properties with dilation or erosion behaviour, compare e.g. the application in [2]. We illustrate the robust dilation-like effect of a 0.75-quantile filter in Fig. 1 (c)–(f). Note that morphological dilation reacts very sensitively to noise whereas the quantile filter achieves similar structural filtering of the image while being much more robust towards noise.

III. MULTIVARIATE MEDIAN FILTERING

Attempts to generalise the median concept to multivariate data can be traced back to Hayford's 1902 work [9]. In 1909 Weber [19] introduced what is now known as the L^1 median, which became popular in the statistics literature since the 1920s–1930s [7], [8]. The univariate median is known [10] to minimise the sum of absolute differences to the given numbers $a_1, \ldots, a_N \in \mathbb{R}$,

$$m((a)_N) = \operatorname*{argmin}_{x \in \mathbb{R}} \frac{1}{N} \sum_{i=1}^{N} |x - a_i| \,, \tag{1}$$

where we have used the abbreviation $(a)_1^N$ for the sequence $(a_1, \ldots, a_N)$ of numbers. Analogously, we will write $(\boldsymbol{a})_1^N$ for a sequence of data points $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N \in \mathbb{R}^n$.

We remark that the minimiser in (1) is non-unique if N is even. This is generally the case for all argmin formulations of medians and quantiles discussed in the following. Formally one can consider the minimisers as set-valued, or use some additional heuristics to enforce uniqueness. We will not discuss this issue any further here because in the multivariate case the problem is restricted mostly to non-generic data configurations and parameters.

A. L^1 Median

The L^1 median generalises (1) by defining the median of points from $\mathbb{R}^n$ as the point with minimal sum of Euclidean distances to the given points $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N \in \mathbb{R}^n$,

$$\boldsymbol{m}_{L^1}((\boldsymbol{a})_1^N) := \operatorname*{argmin}_{\boldsymbol{x} \in \mathbb{R}^n} \frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{x} - \boldsymbol{a}_i\| \,. \tag{2}$$

The L^1 median is now well-understood, and efficient algorithms for its computation in arbitrary dimension are available [18]. Unlike the univariate median, the L^1 median will often attain values that are not among the input values (but still in their convex hull). The same is true for the other multivariate median concepts discussed in the following.

B. Oja Median

The L^1 median is equivariant only w.r.t. similarity transforms of the data space (i.e. if the input data undergo a similarity transform, their L^1 median changes by the same transform), unlike the univariate median that is equivariant under arbitrary strictly monotone mappings of $\mathbb{R}$. To overcome this limitation, alternatives to the L^1 median have been discussed in literature since the 1970s, with the interest to achieve at least affine equivariance, see the overview [15].

An interesting concept is Oja's simplex median [14] which generalises (1) in a different way: Interpreting $|b-a|$ for $a, b \in \mathbb{R}$, $a < b$ as the length of the interval $[a,b]$, i.e. a one-dimensional simplex, one can define a median in $\mathbb{R}^n$ as the minimiser of the sum of volumes of n-dimensional simplices with the median and n of the data points as vertices,

$$\boldsymbol{m}_{\mathrm{Oja}}((\boldsymbol{a})_1^N) := \operatorname*{argmin}_{\boldsymbol{x} \in \mathbb{R}^n} \frac{1}{\binom{N}{n}} \sum_{1 \le i_1 < \ldots < i_n \le N} \left| [\boldsymbol{x}, \boldsymbol{a}_{i_1}, \ldots, \boldsymbol{a}_{i_n}] \right| \,. \tag{3}$$

Albeit theoretically elegant, and obviously affine equivariant, the Oja median suffers from its computational complexity that increases with dimension.

C. Half-Space Median

Further affine equivariant median concepts in literature are motivated by geometric combinatorial ideas. We mention here the half-space median [12] which defines the half-space depth of a point $\boldsymbol{p} \in \mathbb{R}^n$ w.r.t. data points $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N \in \mathbb{R}^n$ as the minimal number of data points that can lie on one side of a hyperplane through $\boldsymbol{p}$. By a slight reformulation, we can define the half-space potential

$$V_{\mathrm{HS};(\boldsymbol{a})_1^N}(\boldsymbol{p}) := \max_{\boldsymbol{v} \in \mathrm{S}^{n-1}} V_{\mathrm{HS};(\boldsymbol{a})_1^N}(\boldsymbol{p}, \boldsymbol{v}) \tag{4}$$

where the maximisation over n-dimensional unit vectors, i.e. the unit sphere S^{n-1}, is applied to directional half-space potentials

$$V_{\mathrm{HS};(\boldsymbol{a})_1^N}(\boldsymbol{p}, \boldsymbol{v}) := \frac{N_- - N_+}{N_- + N_+} \,, \tag{5}$$

$$N_+ := \#\{i \in \{1, \ldots, N\} \mid \langle \boldsymbol{a}_i - \boldsymbol{p}, \boldsymbol{v} \rangle > 0\} \,, \tag{6}$$

$$N_- := \#\{i \in \{1, \ldots, N\} \mid \langle \boldsymbol{a}_i - \boldsymbol{p}, \boldsymbol{v} \rangle \le 0\} \,. \tag{7}$$

Eventually the half-space median can be stated as

$$\boldsymbol{m}_{\mathrm{HS}}((\boldsymbol{a})_1^N) := \operatorname*{argmin}_{\boldsymbol{x}\in\mathbb{R}^n} V_{\mathrm{HS};(\boldsymbol{a})_1^N}(\boldsymbol{x})\,. \tag{8}$$

Despite the formal similarity of (2), (3) and (8) as minimisations of certain potentials it should be noticed that the objective functions of the L^1 and Oja medians are convex (that of the L^1 median even strictly convex if the data are not degenerated, i.e. they span $\mathbb{R}^n$), whereas the objective function of the half-space median is a piecewise constant jump function. It is, however, unimodal, and its sub-level sets $L_s(V_{\mathrm{HS};(\boldsymbol{a})_1^N}) := \{\boldsymbol{p}\in\mathbb{R}^n \mid V_{\mathrm{HS};(\boldsymbol{a})_1^N}(\boldsymbol{p}) \le s\}$ are convex by construction.

In the processing of multivariate images, early attempts to establish median filters were directed at median concepts that would always select among the given data values [1], [5]. Later on mainly the L^1 median was adopted e.g. for colour images [16], for matrix-valued images [22] and for colour images using a transform between a colour space and symmetric matrices [11]. More recently, also the Oja median has been proposed for image filtering [20], however, due to its complexity practical problems remain.

IV. MULTIVARIATE QUANTILES

An attempt to establish also multivariate quantile filters in image processing has been made in [22] in the context of matrix-valued images.

For $\alpha\in(0,1)$, univariate α-quantiles q_α can be described by a modification of the minimisation property (1) of the univariate median. Weighting positive and negative differences differently, one has

$$q_\alpha((a)_1^N) = \operatorname*{argmin}_{x\in\mathbb{R}} \frac{2-2\alpha}{N}\sum_{\substack{i=1\\ a_i<x}}^{N}|x-a_i| + \frac{2\alpha}{N}\sum_{\substack{i=1\\ a_i>x}}^{N}|x-a_i| \tag{9}$$

(the boundary cases $\alpha=0$, minimum, and $\alpha=1$, maximum, can be included by taking limits). For the following, it is interesting that this characterisation of the α-quantile can be rewritten in terms of the objective function $V(x) = \frac{1}{N}\sum_{i=1}^N|x-a_i|$ from the median characterisation (1) as

$$\begin{aligned}\boldsymbol{q}_\alpha((a)_1^N) &= \operatorname*{argmin}_{x\in\mathbb{R}}\big(V(x)-(2\alpha-1)x\big)\\ &= \big(\partial V\big)^{-1}(2\alpha-1)\end{aligned} \tag{10}$$

where ∂V is the subgradient of V.

A. Multivariate Quantile Parameter

In order to transfer (9) to multivariate data, the crucial question is how the differences $\boldsymbol{x}-\boldsymbol{a}_i$ are to be distributed to the "positive" and "negative" part. Such a separation of sums will generally have to be based on some preferred direction. In the context of matrix-valued images such an attempt was made in [22] (later transferred to colour images in [21]) where the asymmetric weights were applied to positive and negative eigenvalues of symmetric matrices. Clearly, this choice relies on a directional preference for eigenvalues, which made sense in this particular case.

In more general multivariate data, however, there is not always a natural preferred direction; all directions should be treated equally. It has therefore been proposed in [3], [4], [6] that the quantile parameter itself should have a magnitude and a direction. Rescaling the parameter $\alpha\in[0,1]$ of real-valued quantiles to $r := 2\alpha-1\in[-1,1]$, one sees that this range is the unit ball in $\mathbb{R}$. Analogously, the appropriate parameter range for quantiles in $\mathbb{R}^n$ is the unit ball B_n, such that one aims at defining $\boldsymbol{r}$-quantiles with

$$\boldsymbol{r}\in\mathrm{B}_n := \big\{\boldsymbol{x}\in\mathbb{R}^n \mid |\boldsymbol{x}\le 1\big\}\,. \tag{11}$$

In this parametrisation, $\boldsymbol{r}=0$ always refers to the median.

B. L^1 Quantiles

Extending (10), it is further proposed in [3], [4], [6] to use the derivative (or subdifferential, depending on the precise setting) of the function minimised by some multivariate median as quantile parameter. For the L^1 median this amounts to the L^1 $\boldsymbol{r}$-quantile

$$\begin{aligned}\boldsymbol{q}_{\boldsymbol{r}}^{L^1}((\boldsymbol{a})_1^N) &:= \operatorname{argmin}_{\boldsymbol{x}\in\mathbb{R}^n}\frac{1}{N}\sum_{i=1}^N\big(\|\boldsymbol{x}-\boldsymbol{a}_i\| - \langle\boldsymbol{r},\boldsymbol{x}-\boldsymbol{a}_i\rangle\big)\\ &= \operatorname{argmin}_{\boldsymbol{x}\in\mathbb{R}^n}\frac{1}{N}\sum_{i=1}^N\|\boldsymbol{x}-\boldsymbol{a}_i\| - \langle\boldsymbol{r},\boldsymbol{x}\rangle\end{aligned} \tag{12}$$

as worked out in [6]. It should be noted that for non-degenerate input data $\boldsymbol{a}_1,\dots,\boldsymbol{a}_N$ (spanning $\mathbb{R}^n$) the gradient of $M(\boldsymbol{x})$ approaches unit norm only asymptotically when $\boldsymbol{x}$ goes to infinity. L^1-quantiles even of bounded input data therefore extend infinitely.

C. Oja Quantiles

Quantiles associated to the Oja median have been discussed in [3], [4]. We restrict ourselves for the time being to the bivariate case ($n=2$) with $\boldsymbol{x}=(x,y)^{\mathrm{T}}$, $\boldsymbol{a}_i=(a_i,b_i)^{\mathrm{T}}$. Translating from the terminology used in [3] (in which the roles of quantile parameter and quantile are interchanged, such that $\boldsymbol{r}$ is called quantile of $\boldsymbol{q}_{\boldsymbol{r}}$), one considers derivatives of the objective function

$$V_{\mathrm{Oja}}(\boldsymbol{x}) = \frac{2}{N(N-1)}\sum_{1\le i<j\le N}\frac{1}{6}\big|D(\boldsymbol{x},\boldsymbol{a}_i,\boldsymbol{a}_j)\big|\,, \tag{13}$$

$$D(\boldsymbol{x},\boldsymbol{a}_i,\boldsymbol{a}_j) := \det\begin{pmatrix}1 & 1 & 1\\ \boldsymbol{x} & \boldsymbol{a}_i & \boldsymbol{a}_j\end{pmatrix} \tag{14}$$

of the bivariate Oja median. However, the gradient

$$\boldsymbol{\nabla}V_{\mathrm{Oja}}(\boldsymbol{x}) = \frac{\sum_{1\le i<j\le N}\operatorname{sgn}\big(D(\boldsymbol{x},\boldsymbol{a}_i,\boldsymbol{a}_j)\big)\cdot\begin{pmatrix}b_i-b_j\\ a_j-a_i\end{pmatrix}}{3N(N-1)} \tag{15}$$

is not normalised to the admissible range B_2; depending on the input data, they might cover a substantially larger or smaller range. This range is always bounded, since obviously $|\mathrm{d}V_{\mathrm{Oja}}/\mathrm{d}x| \le B_x := \sum_{1\le i<j\le N}|b_j-b_i|/(3N(N-1))$, $|\mathrm{d}V_{\mathrm{Oja}}/\mathrm{d}y| \le B_y := \sum_{1\le i<j\le N}|a_j-a_i|/(3N(N-1))$. An additional normalisation $\boldsymbol{u}(\boldsymbol{x}) := C\boldsymbol{\nabla}V_{\mathrm{Oja}}(\boldsymbol{x})$ with a suitable factor C should therefore ensure that $\boldsymbol{r} := \boldsymbol{u}(\boldsymbol{x}) = \big(u(\boldsymbol{x}),v(\boldsymbol{x})\big)^{\mathrm{T}}$

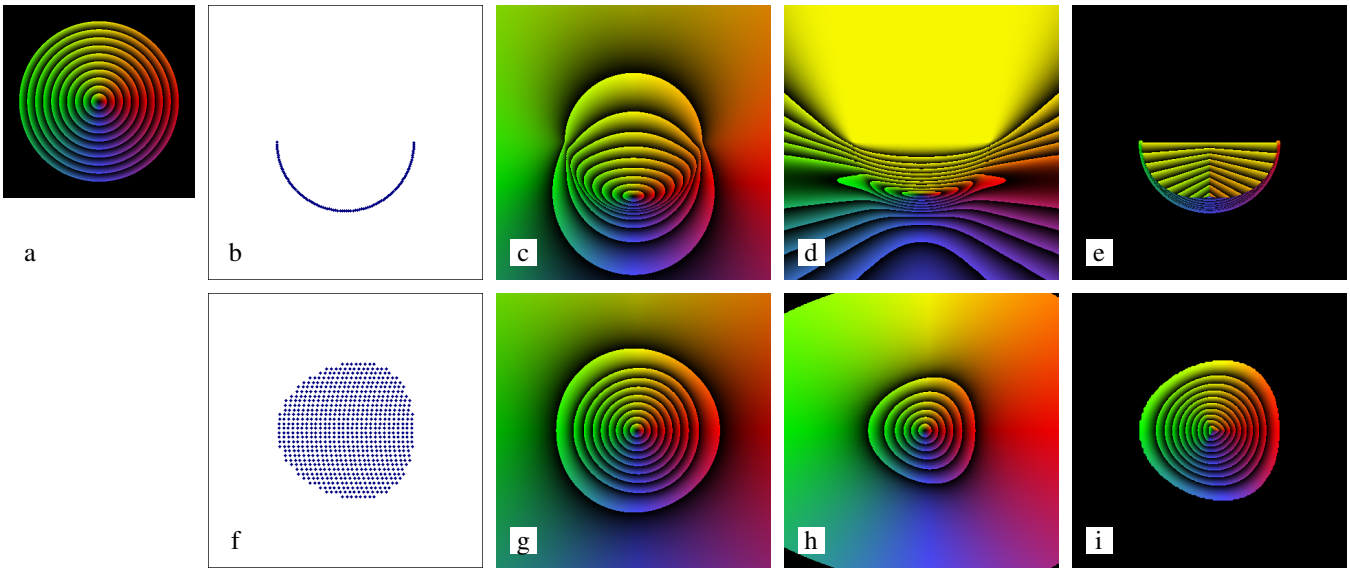


Fig. 2. Quantile parameter maps. **(a)** Colour assignment for the unit disc B_2, the parameter range of bivariate quantiles. – **(b)** Data set of 100 points arranged on a half-circle. – **(c)** Quantile parameter map for bivariate L^1 quantiles of the data set (b). – **(d)** Quantile parameters of Oja quantiles of (b). – **(e)** Quantile parameters of half-space quantiles of (b). – **(f)** Data set of 716 points. – **(g)** Quantile parameter map for bivariate L^1 quantiles of the data set (f). – **(h)** Quantile parameters of Oja quantiles of (f). – **(i)** Quantile parameters of half-space quantiles of (f).

takes values in B_2. The desired Oja quantiles could then be obtained by inverting the function $\boldsymbol{x} \mapsto \boldsymbol{u}(\boldsymbol{x})$ as

$$\boldsymbol{q}_{\boldsymbol{r}}^{\mathrm{Oja}}((\boldsymbol{a})_1^N) := \boldsymbol{u}^{-1}(\boldsymbol{r}) \,. \tag{16}$$

One possible choice for the normalisation, which we will use in the experimental section, is $C = 1/\sqrt{B_x^2 + B_y^2}$. With this choice, $\boldsymbol{u}$ takes values in a subset of B_2.

A severe disadvantage of the so normalised Oja quantiles is that in general substantial regions of B_2 are missing in the range of quantile parameters. This problem is mitigated but not eliminated with $C = 1/\max_{\boldsymbol{x}\in\mathbb{R}^2} \|\nabla V_{\mathrm{Oja}}(\boldsymbol{x})\|$.

D. Half-Space Quantiles

Unlike the objective functions of L^1 and Oja median, the half-space potential $V := V_{\mathrm{HS};(\boldsymbol{a})_1^N}$ of a sequence of input values as defined in (4) is not differentiable. It is not even continuous; instead it is piecewise constant with jumps along a network of straight line segments. The previous definitions of $\boldsymbol{r}$-quantiles can therefore not be translated straightforwardly. However, the convexity of sub-level sets of V_{HS}, and the fact that V takes its values in $[0,1]$, open another option: We define as the half-space $\boldsymbol{r}$-quantile of $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N \in \mathbb{R}^n$ the extreme point of the convex sub-level set $L_s(V)$ with $s = |\boldsymbol{r}|$ in the direction of $\boldsymbol{r}$. Since the minimum of V is greater than 0 in some configurations, we define the quantile to be the median if $L_s(V)$ is empty. Summarising, we have

$$\boldsymbol{q}_{\boldsymbol{r}}^{\mathrm{HS}}((\boldsymbol{a})_1^N) := \begin{cases} \operatorname*{argmax}_{\boldsymbol{x}\in L_{|\boldsymbol{r}|}(V)} \langle \boldsymbol{x}, \boldsymbol{r}\rangle \,, & |\boldsymbol{r}| > \min_{\boldsymbol{x}\in\mathbb{R}^n} V(\boldsymbol{x}) \,, \\ \boldsymbol{m}_{\mathrm{HS}}((\boldsymbol{a})_1^N) & \text{otherwise.} \end{cases} \tag{17}$$

In contrast to the L^1 and Oja quantiles, half-space quantiles are always located in the convex hull of the input data. Quantiles with $|\boldsymbol{r}| = 1$ are located on the boundary of the convex hull. This is an advantageous property for image filtering because it guarantees that a so-defined filter does not extend the intensity range of images being filtered.

V. Evaluation

A. Quantile Parameter Maps

In a first series of experiments, see Figure 2, we visualise the quantile parameters for bivariate quantiles of the three types discussed. In Fig. 2 (a), we show a colour encoding for the unit disc B_2, i.e. the parameter range of bivariate quantiles. Here, hue represents the orientation of parameter vectors $\boldsymbol{r}$, whereas the intensity is increased from zero to maximum in each of ten concentric zones to indicate $|\boldsymbol{r}|$. In the top row, Fig. 2 (b) shows an exemplary data set of 100 points equally spaced along a half-circle. In Fig. 2 (c)–(e) the distribution of quantile parameters $\boldsymbol{r}$ in the same planar region as represented in frame (b) is shown for the L^1, Oja and half-space quantiles, respectively. Each point $\boldsymbol{x} \in \mathbb{R}^2$ of the plane is coloured with the colour representing the $\boldsymbol{r}$ for which the respective quantile $\boldsymbol{q}_{\boldsymbol{r}}$ equals $\boldsymbol{x}$. Although the data set is bounded, L^1 quantiles for $\boldsymbol{r}$ cover the entire plane $\mathbb{R}^2$, and the mapping between $\boldsymbol{r}$ and $\boldsymbol{x}$ is continuous.

Regarding the Oja quantile, we notice first that some quantile parameters (like $\boldsymbol{r} = (r,0)^{\mathrm{T}}$ for $|r| \gtrsim 0.5$) do not occur at all (the 0.5-level line of $\boldsymbol{r}$ depicted by the margin of the fifth colour zone decomposes to two branches extending to infinity), so the corresponding quantiles do not exist. Remarkably, in the case of the Oja quantile some sub-level sets for $|\boldsymbol{r}|$ are non-convex.

In contrast, half-space quantiles are strictly constrained to the convex hull of the data set; the outer region shown in black does not contain quantiles. However, it can be seen that

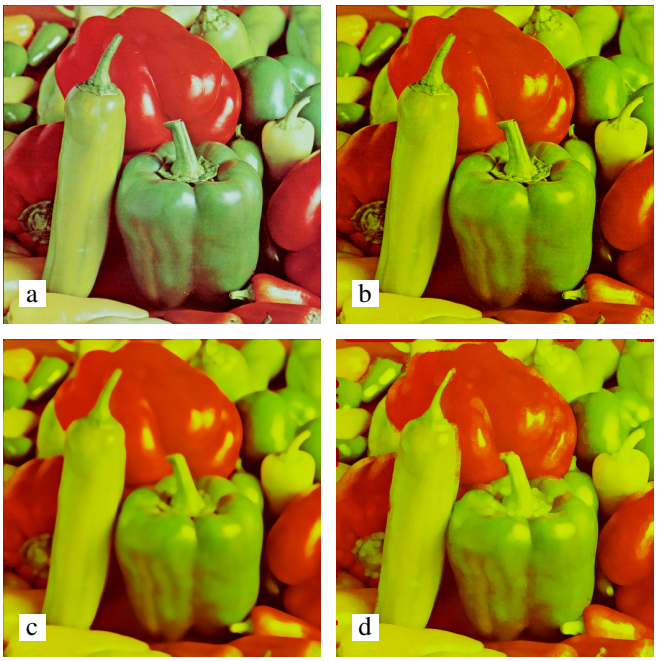


Fig. 3. Quantile filtering of a bivariate test image. **(a)** RGB colour image *peppers*, 512×512 pixels. – **(b)** Bivariate test image obtained from (a) by using only the red and green channels. – **(c)** Result of L^1-quantile filtering of (b), $\boldsymbol{r} = (0.9, 0)^{\mathrm{T}}$, 1 iteration. – **(d)** Result of halfspace-quantile filtering of (b), $\boldsymbol{r} = (0.9, 0)^{\mathrm{T}}$, 1 iteration.

there are kinks in the level lines of $|\boldsymbol{r}|$. At these locations, the mapping $\boldsymbol{x} \mapsto \boldsymbol{r}$ jumps; it is set-valued, and multiple quantile parameters $\boldsymbol{r}$ yield the same quantile. However, the parameters yielding the same quantile value form a connected region.

The bottom row, Fig. 2 (f)–(i) shows another data set with the corresponding quantile parameters. To obtain the data points (x, y) of this data set, 716 points (ξ, η) were sampled from a regular grid within B_2 and transformed via $(\xi, \eta) \mapsto (\xi + 0.2\eta^2, \eta)$. This kind of data set is representative for data selected from a smooth bivariate image within a disc-shaped window, as it occurs within image filtering (see next subsection). The general properties of the quantile maps are similar as for the previous data set. Discontinuities in the half-space quantile map are less prominent here, however. Sub-level sets of the Oja quantile parameter are not concave in this case but it is evident that already quantiles with $|\boldsymbol{r}| \approx 0.7$ lie far outside the convex hull of the data, and quantile parameters of larger magnitude cannot be realised at all.

B. Quantile Filtering of Bivariate Images

Our second series of experiments, see Figures 3 and 4, is targeted at the application of quantiles for image filtering. Whereas L^1 quantiles can be computed efficiently in arbitrary dimension using a straightforward modification of the algorithm from [18], efficient algorithmics for the half-space quantiles, especially in dimensions greater than 2, is still a topic of ongoing research. For this reason, we restrict ourselves here to the filtering of two-channel images and defer an extension to RGB images for future work.

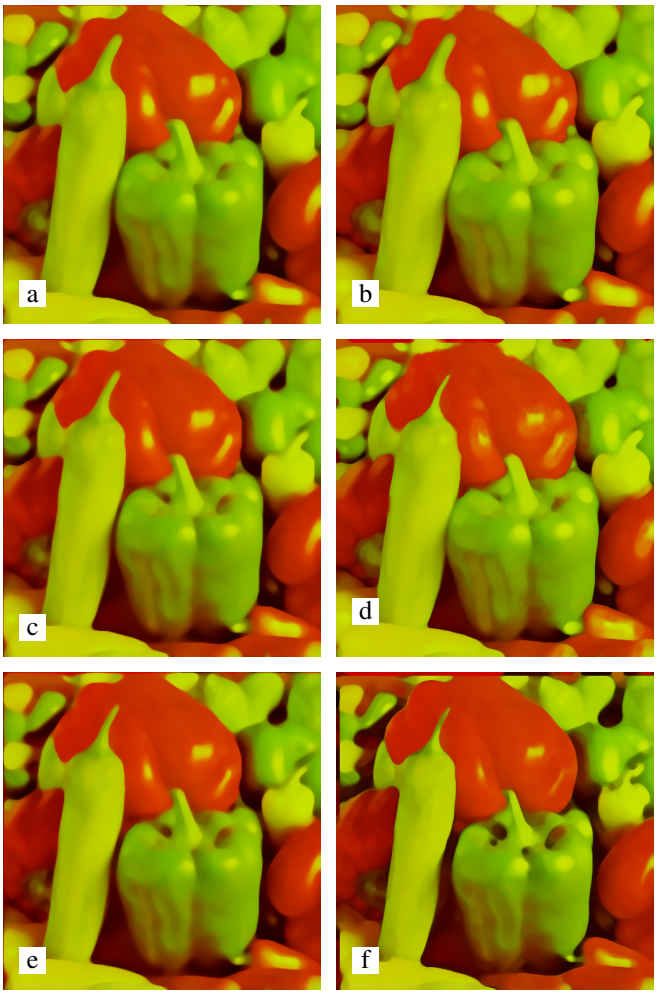


Fig. 4. Quantile filtering of a bivariate test image, continued from Fig. 3. **(a)** Result of L^1-quantile filtering of Fig. 3 (b), $\boldsymbol{r} = (0, 0.25)^{\mathrm{T}}$, 5 iterations. – **(b)** Halfspace-quantile filter, $\boldsymbol{r} = (0, 0.25)^{\mathrm{T}}$, 5 iterations. – **(c)** L^1-quantile filter, $\boldsymbol{r} = (0.25, 0)^{\mathrm{T}}$, 5 iterations. – **(d)** Halfspace-quantile filter, $\boldsymbol{r} = (0.25, 0)^{\mathrm{T}}$, 5 iterations. – **(e)** L^1-quantile filter, $\boldsymbol{r} = (0.177, -0.177)^{\mathrm{T}}$, 5 iterations. – **(f)** Halfspace-quantile filter, $\boldsymbol{r} = (0.177, -0.177)^{\mathrm{T}}$, 5 iterations.

Moreover, from the previous theoretical and experimental findings it must be concluded that Oja quantiles as defined in Section IV-C are unsuitable for establishing an image filter. This leads us to comparing L^1 and half-space quantile filters.

Fig. 3 (a) shows the *peppers* test image which, by the dominance of red and green colours, is chosen as a suitable candidate to demonstrate two-channel image filtering. In Fig. 3 (b) we show its two-channel version consisting only of the red and green colour channels. For all quantile filters in the following, we use the same disc-shaped sliding window of radius 5 as mentioned in the Introduction.

Fig. 3 (c) and (d) show the results of single iterations of L^1 and halfspace quantile filtering, respectively, with a quantile parameter $\boldsymbol{r}$ of magnitude 0.9, thus close to the boundary of the admissible range B_2. Visually the filtering effect of the halfspace quantile is more pronounced and resembles a morphological dilation, in agreement with our comment on univariate quantile filtering at the end of Section II. However,

the L^1 filter exceeds the intensity range of the input image: Whereas the maximal value of the red channel in the original image is 231, it is 255 in the filtered image (cut off by the image file format with 1 byte per channel). This is not the case for the half-space quantile result, thus confirming our previous findings. For the practical application of quantiles this implies that in L^1 quantile filtering results artificial colours are to be expected, which is generally an undesired effect in colour image processing.

In the following, we choose quantile parameters with smaller magnitude for which both quantile filters do not exceed the input intensity range. As the effect of a single filter iteration is hard to notice with the smaller quantile parameters, we increase the iteration count to 5 from now on. In Fig. 4 (a) and (b) we use a quantile parameter directed to bright green values. We can indeed observe a dilation-like behaviour for green structures, i.e. the green structures grow in size at the cost of darker and red regions; note particularly the stems of peppers.

Surprising at first glance, just using a filter parameter in positive red direction, Fig. 4 (c) and (d), dilates only some of the red structures. The reason is that less saturated green structures in this image are often overall brighter than adjacent red structures, such that even their red-channel values are greater. A more pronounced filtering in favour of red structures can be obtained by specifying a quantile parameter with positive red and negative green component, see Fig. 4 (e) and (f).

VI. CONCLUSIONS

In this contribution, we have explored image filters for multivariate images based on multivariate quantiles. Building on previous work from the statistics community, which suggested that multivariate quantiles should have multidimensional quantile parameters within the unit ball in data space, and provided definitions for quantiles derived from the L^1 and Oja medians, we extended the concept to a quantile derived from the half-space median.

In a theoretical discussion supported by numerical experiments, we demonstrated that these quantile concepts differ substantially regarding properties that are highly relevant for image filtering. In particular, Oja quantiles for given parameters may not exist for certain data configurations which makes them unusable for image filtering at the present stage of investigation. Algorithmically efficient L^1 quantiles yield values exceeding the intensity range of the input image. Half-space quantiles are theoretically appealing and yield values within the convex hull of the input data.

In an experiment on a bivariate (red–green) test image we have demonstrated that quantile filters with suitable choice of parameters can be employed for a colour-selective dilation of image structures.

There are several topics and open questions for ongoing research. For space limitation, equivariance properties of quantiles have not been discussed here; however, none of the quantiles discussed here is affine equivariant. Devising theoretically well-founded affine equivariant quantiles will be an interesting goal. Moreover, it remains an open question whether the shortcomings of Oja quantiles can be overcome by modifying the concept.

Regarding numerics, it will be important to design efficient algorithms for halfspace quantile filtering, especially in higher dimensions. On the practical side, it will be interesting to study the usability of multivariate quantile filters for colour-selective structure enhancement in relevant application problems.

REFERENCES

[1] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," *Proceedings of the IEEE*, vol. 78, no. 4, pp. 678–689, 1990.

[2] A. Bengtsson and H. Bengtsson, "Microarray image analysis: background estimation using quantile and morphological filters," *BMC Bioinformatics*, vol. 7, no. 96, pp. 1–15, 2006.

[3] B. M. Brown and T. P. Hettmansperger, "Affine invariant rank methods in the bivariate location model," *Journal of the Royal Statistical Society B*, vol. 49, no. 3, pp. 301–310, 1987.

[4] ——, "An affine invariant bivariate version of the sign test," *Journal of the Royal Statistical Society B*, vol. 51, no. 1, pp. 117–125, 1989.

[5] V. Caselles, G. Sapiro, and D. H. Chung, "Vector median filters, inf-sup operations, and coupled PDE's: Theoretical connections," *Journal of Mathematical Imaging and Vision*, vol. 8, pp. 109–119, 2000.

[6] P. Chaudhuri, "On a geometric notion of quantiles for multivariate data," *Journal of the Americal Statistical Association*, vol. 91, no. 434, pp. 862–872, 1996.

[7] L. Galvani, "Sulla determinazione del centro di gravita e del centro mediano di una popolazione, con applicazioni alla popolazione italiana censita il 1 dicembre 1921," *Metron*, vol. 11, pp. 17–48, 1933.

[8] C. Gini and L. Galvani, "Di talune estensioni dei concetti di media ai caratteri qualitativi," *Metron*, vol. 8, pp. 3–209, 1929.

[9] J. F. Hayford, "What is the center of an area, or the center of a population?" *Journal of the American Statistical Association*, vol. 8, no. 58, pp. 47–58, 1902.

[10] D. Jackson, "Note on the median of a set of numbers," *Bulletin of the American Mathematical Society*, vol. 27, pp. 160–164, 1921.

[11] A. Kleefeld, M. Breuß, M. Welk, and B. Burgeth, "Adaptive filters for color images: median filtering and its extensions," in *Computational Color Imaging*, ser. Lecture Notes in Computer Science, A. Trémeau, R. Schettini, and S. Tominaga, Eds. Cham: Springer, 2015, vol. 9016, pp. 149–158.

[12] R. Y. Liu, "On a notion of data depth based on random simplices," *The Annals of Statistics*, vol. 18, no. 1, pp. 405–414, 1990.

[13] G. Matheron, *Eléments pour une théorie des milieux poreux*. Paris: Masson, 1967.

[14] H. Oja, "Descriptive statistics for multivariate distributions," *Statistics and Probability Letters*, vol. 1, pp. 327–332, 1983.

[15] C. G. Small, "A survey of multidimensional medians," *International Statistical Review*, vol. 58, no. 3, pp. 263–277, 1990.

[16] C. Spence and C. Fancourt, "An iterative method for vector median filtering," in *Proc. 2007 IEEE International Conference on Image Processing*, vol. 5, 2007, pp. 265–268.

[17] J. W. Tukey, *Exploratory Data Analysis*. Menlo Park: Addison–Wesley, 1971.

[18] Y. Vardi and C.-H. Zhang, "A modified Weiszfeld algorithm for the Fermat-Weber location problem," *Mathematical Programming A*, vol. 90, pp. 559–566, 2001.

[19] A. Weber, *Über den Standort der Industrien*. Tübingen: Mohr, 1909.

[20] M. Welk, "Multivariate median filters and partial differential equations," *Journal of Mathematical Imaging and Vision*, vol. 56, pp. 320–351, 2016.

[21] M. Welk, A. Kleefeld, and M. Breuß, "Quantile filtering of colour images via symmetric matrices," *Mathematical Morphology: Theory and Applications*, vol. 1, no. 1, pp. 136–174, 2016.

[22] M. Welk, J. Weickert, F. Becker, C. Schnörr, C. Feddern, and B. Burgeth, "Median and related local filters for tensor-valued images," *Signal Processing*, vol. 87, pp. 291–308, 2007.

Motion Artefact Compensation for Multi-Line Scan Imaging

Nicole Brosch[1], Svorad Štolc[1], Simon Breuss[1], Doris Antensteiner[1]

***Abstract*— This work focuses on the compensation of transport synchronization artefacts that may occur during multi-line scan acquisitions. We reduce these motion artefacts by a warping function that stretches/squeezes line frames in the scanning domain that were acquired too early/late. The estimation of the warping function is controlled by comparing light field views and enforce uniform spacing between line acquisitions. This approach enables multi-line scan systems to perform multi-line scan light field imaging largely independent from the transport and trigger quality.**

I. INTRODUCTION

Line scan imaging is a popular choice when performing industrial quality inspection [4]. However, when capturing moving objects *motion artefacts* may arise when the transport velocity of the object is not perfectly synchronized with the camera [4], [6], [8]. While in conventional line scanning (i.e., single line) such artefacts are not distinguishable from the correct signal, they become visible in light fields acquired with a multi-line scan system [6] (Figure 1). The standard solution to motion artefacts in line scan imaging, is to use high-end hardware components, such as high-precision transport stages and motion sensors [4]. However, we have observed that despite such hardware, acquisitions might sill suffer from such artefacts especially at high magnifications. The importance of compensating for motion artefacts was stressed by existing line scan imaging approaches (e.g., [4], [6], [8]) and addressed in a multi-line scenario in [1], i.e., the approach spotlighted in this paper. Related works outside the realm of line scanning, include motion compensation based on explicitly recorded reference patterns [5], [7].

II. ALGORITHM DESCRIPTION

A light field acquired with [6] is stored in an EPI stack $V \in \mathbb{R}^{n\times m\times r}$ (Figure 1). In $V(x_i, v_k, y_j)$ a moving object was captured at n space instances and with m camera lines that consist of r pixels, where $1 \le i \le n$, $1 \le k \le m$ and $1 \le j \le r$. If the transport velocity is not perfectly synchronized with the multi-line scan camera, the distance between successive acquired lines is not constant. This leads to distortions of the assumed integer indices x_i and true sub-pixel indices $\tilde{x}_i$. To compensate for motion artefacts, i.e., the discrepancy between x_i and $\tilde{x}_i$, we first determine $\tilde{x}_i$, and then unwarp pixels in V to generate a new EPI stack, with uniform distances between its position indices $\tilde{x}_i$. To find the true sub-pixel indices $\tilde{x}_i$, that correspond to each observed index x_i, we formulate an energy function,

$$\min_{\tilde{x}} \frac{1}{2}\|E_d(\tilde{x})\|^2 + \frac{\lambda_1}{2}\|E_x(\tilde{x})\|^2 + \frac{\lambda_2}{2}\|E_s(\tilde{x})\|^2. \quad (1)$$

[1]AIT Austrian Institute of Technology, Giefingasse 4, 1210 Vienna, Austria firstname.lastname@ait.ac.at

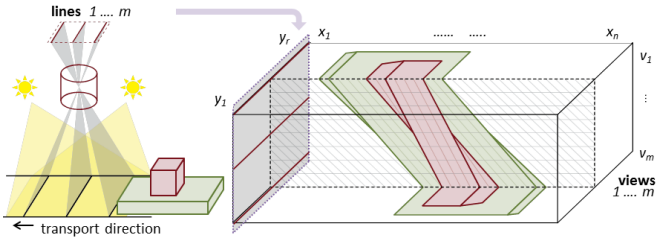


Fig. 1. Illustration of multi-line scan image acquisition setup [6] (left). At each space instance a set of m lines is captured, then the object is moved by a linear transport stage. Multi-line frames with position index x_i are acquired simultaneously. Each camera line captures the object under a different viewing angle and, over the time, contributes to a different view of the object. Each view v_k, consists of line acquisitions with indices x_i. The views compose a linear light field, which is stored in an *epipolar plane image* (EPI) stack $V_{y_j}(x_i, v_k)$ [3] (right). Figure taken from [6].

consisting of a *disparity term* E_d, an *identity term* E_x and a *smoothness term* E_s, which will be discussed in more detail below. Here, λ_1 and λ_2 are used to balance the energy terms.

The *disparity term* is based on the observation that motion artefacts become visible in 3D reconstructions (e.g., Figure 2, a) from light fields acquired with [6]. An object point and an entire multi-line frame associated with x_i that was performed too early/late, causes a smaller/larger disparity than expected. In order to determine a true index $\tilde{x}_i$, we use estimated disparities to locate corresponding multi-line frames in different views and adjust the position of the i-th mutli-line frame from x_i to $\tilde{x}_i$. More precisely, this adjustment is based on balanced forward and backward disparities between two views, i.e., forward disparities between views v_k and v_{k+1} and backward disparities between views v_k and v_{k-1} (Figure 3). In order to speed up the approach, we determine the mean forward disparity $d_{k,i}$ and the mean backward disparity $\bar{d}_{k,i}$ in each index x_i. Given $d_{k,i}$ and $\bar{d}_{k,i}$ for each position index x_i, we infer the true indices $\tilde{x}_i$ with:

$$E_d(\tilde{x}) = D'\tilde{x}, \quad (2)$$

where for each position index x_i for which both forward and backward disparities exist, we form one line in matrix $D' \in \mathbb{R}^{n\times n}$. The corresponding set of linear equations in Eq. (1) for any given view v_k $k \in \{2,...,m-1\}$ are:

$$\begin{aligned} -2\tilde{x}_i + \tilde{x}_{i+\lfloor d_{k,i}\rfloor}(1 - d_{k,i} + \lfloor d_{k,i}\rfloor) + \tilde{x}_{i+\lceil d_{k,i}\rceil}(d_{k,i} - \lfloor d_{k,i}\rfloor) \\ + \tilde{x}_{i+\lfloor \bar{d}_{k,i}\rfloor}(1 - \bar{d}_{k,i} + \lfloor \bar{d}_{k,i}\rfloor) + \tilde{x}_{i+\lceil \bar{d}_{k,i}\rceil}(\bar{d}_{k,i} - \lfloor \bar{d}_{k,i}\rfloor) \\ \approx 0, \forall i \in \{1,...,n\}. \quad (3) \end{aligned}$$

E_d may be generalized e.g., by including calibration information or when exchanging the L2 with an L1 penalization.

The *identity term* assumes that the actual movement is similar to the assumed ideal movement of the transport stage

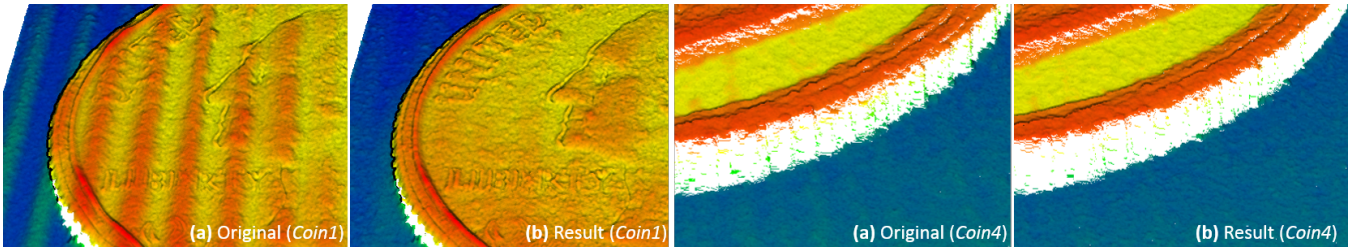


Fig. 2. Visual comparison of 3D reconstructions generated from original (a) and undistorted (b) light field acquisition of coins. Ripples due to motion artefacts in (a) are significantly reduced in (b). For the left example, a camera in free-running mode was used to increase the visibility of motion artefacts.

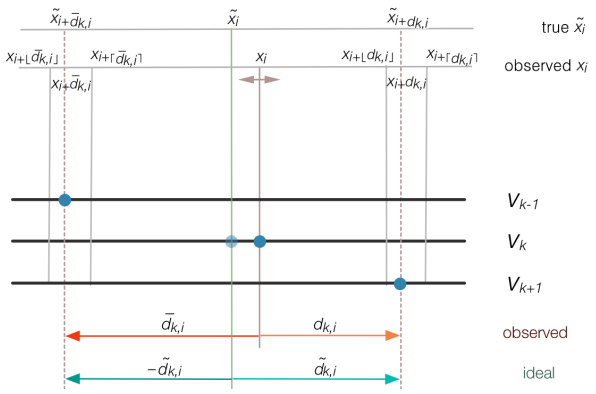


Fig. 3. Illustration of E_d. Disparities relate corresponding object points (dark blue) in different views (v_{k-1}, v_k, v_{k+1}). In presence of transport issues, points are acquired too early/late (light vs. dark blue point in v_k) and the observed index x_i differs from its true index $\tilde{x}_i$. In an ideal EPI stack (perfect transport synchronization) fore- and backward disparities are balanced ($\tilde{d}_{k,i}$, green arrows). In a distorted EPI stack with observed x_i and disparities $\bar{d}_{k,i}$ and $d_{k,i}$ (red and orange arrow) this is not the case. For x_i, $\tilde{x}_i$ can be determined by enforcing the balance between $\tilde{x}_{i+\bar{d}_{k,i}}$ and $\tilde{x}_{i+d_{k,i}}$.

up to a non-accumulative normally distributed error:

$$E_x(\tilde{x}) = x - \tilde{x}. \tag{4}$$

Hence in Eq. (1), we also solve the set of equation:

$$x_i - \tilde{x}_i \approx 0, \quad \forall i \in \{1,...,n\}. \tag{5}$$

The *smoothness term* ensures a smooth solution by penalizing abrupt changes between neighboring $\tilde{x}_i$:

$$E_s(\tilde{x}) = \Delta\tilde{x}, \tag{6}$$

where Δ denotes the Laplacian operator, which is implemented in form of a convolution filter with the kernel $[1,-2,1]$. Thus, we form another set of linear equations:

$$-2\tilde{x}_i + \tilde{x}_{i-1} + \tilde{x}_{i+1} \approx 0, \quad \forall i \in \{1,...,n\}. \tag{7}$$

We express the energy term in Eq. (1) as an overdetermined linear system of equations, which can be solved approximately using a standard least squares solver.

III. EXPERIMENTAL RESULTS

We perform evaluations on EPI stacks acquired with [6]. To increase the visibility of motion artefacts, our experiments are also performed on a *free running dataset*, for which camera and transport are not synchronized via trigger. In Table I, we compare disparity maps that were generated from the original and from our undistorted EPI stacks. Since disparity values in such a (assumed flat) region should be constant, the standard deviation can be used as a quality measure, where low values indicate less artefacts than larger ones. Table I also contains an analogue evaluation for depth maps obtained with the calibrated system [2] and our compensation approach which includes the calibration information. In our dataset, the proposed compensation approach significantly reduces the motion artefacts (Figure 2, Table I).

TABLE I
EVALUATION: STANDARD DEVIATIONS OF DISPARITIES/DEPTHS IN FLAT REGION BEFORE (ORIGINAL) AND AFTER (RESULT) COMPENSATION

free-running camera			**synchronization via trigger**		
Data set	Original	Result	Data set	Original	Result
Note1	0.3061	0.0986	*Coin3*	0.0686	0.0617
Note2	0.2896	0.1153	*NoObject1*	0.0612	0.0481
PCB1	0.3069	0.1283	**calibrated system** (depth)		
PCB2	0.3361	0.1319	Data set	Original	Result
Coin1	0.2954	0.0992	*Coin4*	0.0147	0.0137
Coin2	0.2771	0.0912	*NoObject2*	0.0161	0.0133

IV. CONCLUSION

The compensation of motion artefacts allows multi-line scan light field imaging when the transport cannot be controlled with high precision. The compensation approach constrained the transport position indices according the information obtained by comparing multiple views and enforced uniform spacing between line acquisitions. The undistorted views were computed according to a warping function, which significantly reduced artefacts in our test data.

REFERENCES

[1] N. Brosch et al., "Warping-based motion artefact compensation for multi-line scan light field imaging", *EI'18*, 2018, pp.273-1-273-6.
[2] B. Blaschitz et al., "Geometric calibration and image rectification of a multi-line scan camera for accurate 3D reconstruction", *EI'18*, 2018, pp.240-1-240-6.
[3] R. Bolles et al., "Epipolar-Plane Image Analysis: An approach to determining structure from motion", *IJCV'87*, 1, 1, 1987.
[4] R. Davies, *Machine Vision: Theory, Algorithms, Practicalities.* Morgan Kaufmann Publishers Inc., 2004.
[5] B. Krolla et al., "Light field from smartphone-based dual video", *ECCV'14*, 2014, pp. 600-610.
[6] S. Štolc et al., "Depth and all-in-focus imaging by a multi-line-scan light-field camera," *Journal of Electronic Imaging*, 23, 5, 2014.
[7] T. Weise et al., "Fast 3D scanning with automatic motion compensation", *CVPR'07*, 2007, pp. 1-8.
[8] H. Yuet al., "An antivibration time-delay integration CMOS image sensor with online deblurring algorithm", *TCSVT*, 26, 8, 2016.

impress: Forensic Footwear Impression Retrieval

Manuel Keglevic[1] and Robert Sablatnig[2]

***Abstract*— Footwear impressions are a valuable source of evidence for criminal investigations. By comparing them, forensic experts can show that a footwear impression was made by a specific shoe or impressions at different crime scenes were made by the same suspect. However, this process is very cumbersome and the current software solution used by the Austrian Police uses an annotation based search that is very subjective and thus are not accurate enough. Therefore, the goal of the *impress* project is a system that helps searching through databases with thousands of footwear impression images by automatically computing image similarities.**

I. INTRODUCTION

Footwear impressions are frequently found at various crime scenes. They are easily detected, processed and interpreted, and are therefore a valuable source of evidence for criminal investigations. Especially the combination with other types of forensic evidence, e.g. DNA, toolmarks, fingerprints, etc. offers great potential for solving a crime. Additionally to an estimate of the shoe size, the unique patterns of footwear impressions contain clues to the model and brand of the footwear, which in turn help to limit the number of possible suspects. Further, similar footwear impressions at different crime scenes indicate that the crime was committed by the same suspect. This way, criminal acts committed by serial offenders can be identified. For instance burglaries are a great unease for society and are mostly committed by serial offenders. Solving those cases is a crucial factor in improving the subjective sense of security of the people.

In case a suspect is apprehended, the individual features of the footwear can proove that a footwear impression was made by a specific shoe. For this, forensic experts investigate the model characteristics and individual wear, damages and manufacturing marks. If multiple matching features can be found, the forensic evidence can support the prosecution in court. However, for this investigation the actual shoe has to be retrieved, from either the suspect or the evidence locker, and compared to the footwear impression. Since this process is time consuming and cumbersome, a limitation of the number of necessary comparisons to the most similar footwear impressions is desired by the forensic experts.

Therefore, an automated system that helps searching through databases with thousands of footwear impression images is needed. However, the software solution currently used by the Austrian Police is ill equipped to solve this problem. The main issue is, that the footwear impressions have to be classified by the forensic expert by hand. This is done by describing the patterns of the impressions using a set of predefined classes. However, this process is very subjective and therefore the resulting list of similar impressions is not able to accurately depict the footwear impressions that were made by the same shoe. To alleviate this problem the goal of this project is an automated system, which implements an efficient image comparison methodology to find similar footwear impressions in huge databases of images. Further, to allow an identification of the shoe model and brand a footwear impression reference database, i.e. shoe catalog, is created using the huge amount of shoe sole images freely available in the internet.

[1]Computer Vision Lab, TU Wien, 1040 Vienna, Austria keglevic@cvl.tuwien.ac.at

II. RELATED WORK

Earlier approaches for the forensic comparison of footwear impressions used for instance frequency analysis [2], [3] or local descriptors like Hu-Moments [1] and Scale-Invariant Feature Transform [11], [10]. In 2014 Luostarinen and Lehmussola [6] published a review paper comparing such approaches. They evaluated the influence of the overall footwear impression quality, variations in the orientation, and impact of partial footprints and showed that even the best performing approaches at that time, by Gueham et al. [3] and Nibouche et al. [7], were not suitable for comparing footwear impressions from real criminal cases. These are especially challenging due to background noise, blur and partial and overlapping impressions. In Figure 1 this is visualized on an example.

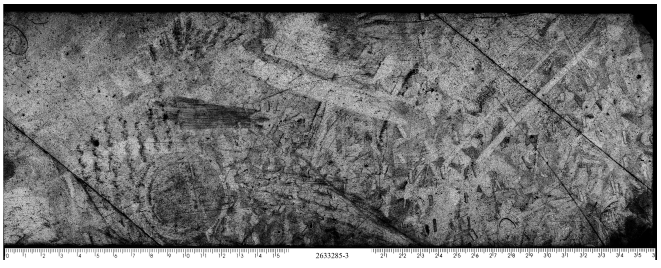


Fig. 1. Footwear impression(s) from a real crime scene collected using a gelatin foil lifter.

More recently, Wang et al. [12] proposed a combination of Wavelets and the Fourier Transform for comparing real impressions. Unfortunately, their approach and all others mentioned above are hard to evaluate, since no results on publicly available datasets were provided. A recent survey from Rida et al. [9] summarizes 21 approaches by listing their published results. According to the listed dataset sizes these results are spread over at least 15 different datasets,

which makes it hard to assess the actual performance of these approaches.

In 2014 Kortylewski et al. [5] published an approach which models the impressions using primitive patterns and contains the first publicly available dataset of footwear impressions. It includes 300 impressions from real criminal cases and 1175 reference impressions. The current state of the art on this dataset is achieved by Kong et al. [4] with an approach based on deep learning; top-1% cumulative match score of 79.7% and a top-5% score of 86.3%.

Another dataset was published in 2017 by Richetelli et al. [8] containing crime scene-like impressions. They used 18 pairs of shoes to create impressions on 4 different substrates using dust or human blood. Their dataset contains in total 180 such impressions and was used in their work to compare methods based on the Fourier-Mellin Transform, phase-only correlation and local interest points. Yet, Zang et al. [13] using established techniques like fine-tuning and data augmentation, show that deep learning based approaches outperform these traditional methods also on this dataset.

III. PROJECT OBJECTIVES

As shown in the previous section, deep learning based approaches are the current state of the art for computing similarities between footwear impression images. However, the two publicly available datasets are not well suited for training such methods. The biggest dataset by Kortylewski et al. [5] contains only 300 crime scene images and 1175 reference impressions. Further, it was not designed to capture the variations among different impressions created by the same shoe which limits the applicable machine learning approaches; e.g. one-shot learning based methods. Additionally, the ground truth is not based on a specific shoe, but on the shoe model and the resolution of the images is less than 1 megapixel. This prohibits a comparison of the individual characteristics which would allow for an exact identification of a specific shoe. In contrast to that, Richetelli et al.'s dataset [8] provides high resolution 600dpi scans of multiple impressions per shoe, but only for 18 different pairs of shoes. Therefore, our first step in the project is the creation of a public dataset with the following properties:

1) *Size*: at least 1000 different shoes
2) *Variance*: multiple modalities (gelatin foil lifters, reference impressions, 3D molds, etc.) and multiple substrates (wood, paper, etc.)
3) *Image quality*: high quality scans of at least 300dpi
4) *Annotations*: ground truth with pixel-precise registration to allow the training of local image similarities

We already started the work on this dataset by creating an acquisition line where participants walk along a given path in order to create predefined footwear impressions. This provides an efficient way to create crime scene-like footwear impressions and using this approach, we already collected multiple impressions of over 300 different pairs of shoes in addition to shoe sole images and model descriptions.

Even though the manual collection of crime scene impressions is inevitable, an automated way to collect reference impressions is desired in order to keep up with the ever changing catalog of shoe models. Therefore, as a second part of the project, it is planned to develop a strategy to automatically download and process images of shoe soles from online retailers like Zalando to create and update a reference database.

Yet, in order to utilize this database, the methodology for the automatic comparison of footwear impressions has to be able to not only handle different modalities of crime scene images, but also images of shoe soles. Furthermore, this comparison of the shoe model characteristics is only the first step to filter the number of possible results. The final goal is to not only identify model characteristics but also individual characteristics, like blemishes or wear, to allow an identification of the specific shoe used to create a footwear impression.

ACKNOWLEDGMENT

This work has been funded by the Austrian security research programme KIRAS of the Federal Ministry for Transport, Innovation and Technology (bmvit) under Grant 867028. We would like to thank the forensic experts of the Criminal Intelligence Service Austria for their help. The Titan X used for this research was donated by the NVIDIA Corporation.

REFERENCES

[1] G. AlGarni and M. Hamiane, "A novel technique for automatic shoeprint image retrieval," *Forensic Science International*, vol. 181, no. 1–3, pp. 10–14, 2008.

[2] P. de Chazal, J. Flynn, and R. B. Reilly, "Automated Processing of Shoeprint Images Based on the Fourier Transform for Use in Forensic Science," *IIEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 341–350, 2005.

[3] M. Gueham, A. Bouridane, D. Crookes, and O. Nibouche, "Automatic Recognition of Shoeprints Using Fourier-Mellin Transform," in *Proceedings of the 2008 NASA/ESA Conference on Adaptive Hardware and Systems*, ser. AHS '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 487–491.

[4] B. Kong, J. Supancic, D. Ramanan, and C. C. Fowlkes, "Cross-Domain Image Matching with Deep Feature Maps," *International Journal of Computer Vision*, pp. 1–13, jan 2019.

[5] A. Kortylewski, T. Albrecht, and T. Vetter, "Unsupervised Footwear Impression Analysis and Retrieval from Crime Scene Data," in *ACCV (Conference on Computer Vision), Workshop on Robust Local Descriptors*, 2014.

[6] T. Luostarinen and A. Lehmussola, "Measuring the Accuracy of Automatic Shoeprint Recognition Methods," *Journal of Forensic Sciences*, vol. 59, no. 6, pp. 1627–1634, 2014.

[7] O. Nibouche, A. Bouridane, M. Gueham, and M. Laadjel, "Rotation Invariant Matching of Partial Shoeprints," in *2009 13th International Machine Vision and Image Processing Conference*, sep 2009, pp. 94–98.

[8] N. Richetelli, M. C. Lee, C. A. Lasky, M. E. Gump, and J. A. Speir, "Classification of footwear outsole patterns using Fourier transform and local interest points," *Forensic Science International*, vol. 275, pp. 102–109, jun 2017.

[9] I. Rida, S. Bakshi, X. Chang, and H. Proenca, "Forensic Shoe-print Identification: A Brief Survey," jan 2019. [Online]. Available: http://arxiv.org/abs/1901.01431

[10] H. Su, D. Crookes, and A. Bouridane, "Shoeprint Image Retrieval by Topological and Pattern Spectra," in *Proceedings of the International Machine Vision and Image Processing Conference*, ser. IMVIP '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 15–22.

[11] H. Su, D. Crookes, A. Bouridane, and M. Gueham, "Shoeprint Image Retrieval Based on Local Image Features," in *Proceedings of the Third International Symposium on Information Assurance and Security*, ser. IAS '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 387–392.

[12] X. Wang, H. Sun, Q. Yu, and C. Zhang, "Automatic Shoeprint Retrieval Algorithm for Real Crime Scenes," in *Computer Vision – ACCV 2014: 12th Asian Conference on Computer Vision, Singapore, Singapore, November 1-5, 2014, Revised Selected Papers, Part I*, D. Cremers, I. Reid, H. Saito, and M.-H. Yang, Eds. Cham: Springer International Publishing, 2015, pp. 399–413.

[13] Y. Zhang, H. Fu, E. Dellandréa, and L. Chen, "Adapting Convolutional Neural Networks on the Shoeprint Retrieval for Forensic Use," in *Biometric Recognition*, J. Zhou, Y. Wang, Z. Sun, Y. Xu, L. Shen, J. Feng, S. Shan, Y. Qiao, Z. Guo, and S. Yu, Eds. Cham: Springer International Publishing, 2017, pp. 520–527.

Machine Vision Solution for a Turnout Tamping Assistance System

Gerald Zauner[1], Tobias Mueller[2], Andreas Theiss[2], Martin Buerger[2], Florian Auer[2]

***Abstract*— In order to guarantee safe and comfortable train travel, the tracks must be in the correct geometric position. For this reason, so-called tamping machines are used worldwide to perform this important task of track maintenance. Turnout-tamping is a complex procedure to improve and stabilize the track situation in turnout-areas, which is usually only carried out by experienced operators. This application paper presents the current state of development of a 3D laser line scanner-based sensor system for a new tamping assistance system, which should support and relieve the operator in complex tamping areas. In this context, semantic segmentation is used to fully automatically identify essential and critical areas in the generated 3D depth images and process them for subsequent machine control.**

I. INTRODUCTION

A. Tamping process

When a train drives along the railway, it generates enormous forces. The entire track consisting of rails, sleepers and ballast is an elastic system that deforms and then returns to its original position. In the end, this high load leads to a deterioration of the track geometry. This can lead to anomalies, because of which the ideal geometry of the track can no longer be guaranteed. In these areas, for example, temporary speed restrictions must be imposed. To avoid such a situation, tracks have to be maintained at regular intervals. This ensures that the ideal geometry of the track is restored. In this context, the so-called track tamping represents the most common maintenance task on railway tracks. Lining refers to correcting the horizontal and vertical alignment of the track, and lifting to the compaction and displacement of the substructure with complete removal of cavities under the sleepers. The combined lifting-lining unit works with a measuring system, gripping the track, raising the track to a predetermined height, correcting for vertical misalignment and simultaneously pivoting the track to correct horizontal alignment. Subsequently, the tamping units are lowered and the tamping tines dip into the ballast. The tamping unit vibrates to fluidize the ballast so that it can rearrange and settle in a dense matrix. Thereafter, the tamping machine moves forward to the next sleeper and the process is repeated. Finally, behind the tamping machine, the result is a track at the correct geometric level, on a homogeneous ballast bed and with restored elasticity [1].

B. Turnout Tamping Assistant

The purpose of the turnout-tamping assistant is to develop an automatic assistance system comparable to level 3 of the SAE J3016 standard (which was originally defined to characterize the autonomous driving of road-bound motor vehicles). Generally, the focus is on the automated support of tamping in difficult environments such as turnout areas and crossings (but not restricted to). At this level of automation, the system creates action recommendations that the operator can confirm prior to each action. The aim is to relieve the operator, to increase the working speed and to stabilize the quality of work at a consistently high level. Basically, the tamping assistance system is also suitable for higher degrees of autonomy [2,3].

Fig. 1. Tamping machine with a roof-mounted 3D laser scanner.

II. 3D DEPTH IMAGE ACQUISITION

A. Relevant object information from 3D scanner image data

The environment (i.e., mainly the superstructure directly in front of the tamping machine) is scanned with a rotating 3D laser scanner mounted on the tamping machine roof (Fig.1). The scanner itself delivers single line scans with millimeter depth accuracy, which are then continuously merged into a depth image with a typical resolution of approximately 4000 x 1000 pixels, where different gray values correspond to different distances to the sensor (i.e. the brighter the image pixels, the closer). The scanner head is mounted in front of the train whereas the actual tamping unit is located approximately in the middle of the machine. Thus, due to moving of the vehicle, there is a small time offset between the scanning of a certain region and the actual tamping process at this particular position, which provides a time window of about 10 seconds for all necessary data processing tasks. Additionally, the raw line scans have to be geometrically corrected as the scanning laser spot moves in a helix-like trajectory along the railway tracks. This correction is of course speed-dependent. The actual working speed during tamping is approx. 1000m/h, which leads to a lateral scan resolution of approx. 2mm. This is sufficient to create detailed scan images that allow visualization even of small objects (such as fasteners, etc., Fig.2).

[1]Gerald Zauner, School of Engineering, Upper Austria University of Applied Sciences, 4600 Wels, Austria
[2]Plasser & Theurer, Export von Bahnbaumaschinen GmbH, Linz

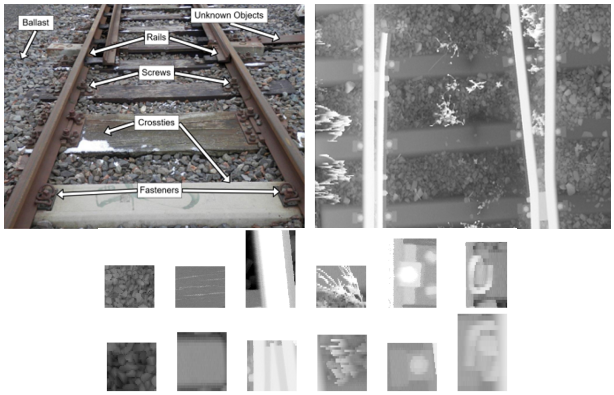


Fig. 2. Picture of the relevant track area (top); typical 3D scan depth image (middle); 3D scan image patches (bottom).

III. Semantic Segmentation of 3D-Scanner Data

Semantic segmentation generally plays a crucial role in computer vision and enables a computer to not only recognize objects in images, but also to locate them pixel-exactly. The recognition and exact delineation of objects in the image is achieved by the classification of each individual pixel, i.e. each pixel is assigned a defined object class (Fig.3). Our original segmentation approach was based on a Fully Convolutional Network (FCN) [4], a popular algorithm for semantic segmentation.

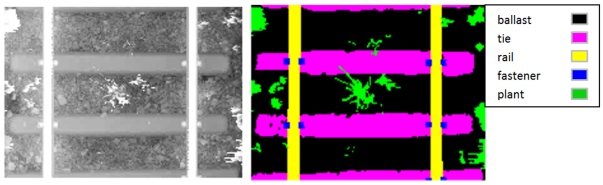


Fig. 3. 3D scanner image (left). Desired result image (right) with pixel-exactly segmented areas representing the relevant image contents such as ballast, rails, plants, etc.

A general challenge in deep learning is the large amount of learning data needed to produce good results i.e. enough annotated images must be provided to train the network. Especially, in semantic segmentation the according effort is considerable, since a pixel-precise marking of the image objects is necessary. A further challenge is the fact that we have to deal with a very imbalanced dataset. The images typically consist of large areas of ballast structures whereas e.g. only very few pixels represent objects like rail screws. Besides real world data we also used artificially generated depth images from a virtual simulation environment (this simulator was originally intended for machine operator training purposes thus providing very realistic 3D scenarios). In this case, no manual labelling of the images was necessary and we were able to provide large quantities of images very quickly (and we could even vary image structures specifically, such as different gravel sizes, etc.). Furthermore, with the help of the simulator, we were also able to intensively test and improve the functionality of the entire assistance system (digital twin).

IV. Results

From the segmentation result relevant information for the tamping process is derived, e.g. exact ballast areas (into which the tamping tines can penetrate) but also sleeper positions and orientations, which are important for correct control of the tamping units. Additionally, the beginning and ending of turnout sections are identified automatically. Also, special equipment along the rail track (like switch rods, etc.) can be identified robustly (Fig.4).

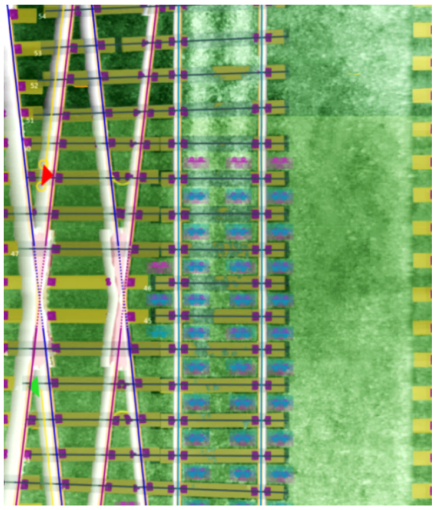


Fig. 4. Typical result of the segmentation process. Different segmented classes such as ballast (green), sleepers (yellow), tracks (white), screws (purple) and plants (green) are shown as a half transparent overlay on the depth image.

First tests in real operation confirm the excellent suitability of the method described. In conclusion, deep learning based semantic segmentation enables the practical realization of very robust outdoor applications under very harsh conditions. The algorithms used are constantly being improved - for example, a new generation of network architecture with an improved segmentation approach is currently being worked on which promises to further improve the detection properties even for very small or thin objects (such as cables, etc.).

ACKNOWLEDGEMENT

We thank our colleagues from the department Digital Track Systems (Plasser & Theurer) for their support.

References

[1] Homepage Plasser & Theurer: https://www.plassertheurer.com/en/machines-systems/tamping.html

[2] F. Auer et al.: Smart Tamping Anwendungsmglichkeiten des Weichenstopf-Assistenzsystems, ZEVrail 2018, Nr. 6.

[3] M. Buerger et al.: Hilfe beim Stopfen Entwicklung eines Weichenstopf-Assistenzsystems, Eisenbahn-Ingenieur 2017, Nr. 6

[4] Jonathan Long et al.: Fully Convolutional Networks for Semantic Segmentation, 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); Electronic ISBN: 978-1-4673-6964-0

Longitudinal Finger Rotation - Deformation Detection and Correction

Bernhard Prommegger[1] , Christof Kauba[1], Michael Linortner[1], and Andreas Uhl[1]

Abstract— **Finger vein biometrics is becoming more and more popular. However, longitudinal finger rotation, which can easily occur in practical applications, causes severe problems as the resulting vein structure is deformed in a non-linear way. These problems will become even more important in the future, as finger vein scanners are evolving toward contact-less acquisition. This paper provides a systematic evaluation regarding the influence of longitudinal rotation on the performance of finger vein recognition systems and the degree to which the deformations can be corrected. It presents two novel approaches to correct the longitudinal rotation, one based on the known rotation angle. The second one compensates the rotational deformation by applying a rotation correction in both directions using a pre-defined angle combined with score level fusion and works without any knowledge of the actual rotation angle. During the experiments, the aforementioned approaches and two additional are applied: one correcting the deformations based on an analysis of the geometric shape of the finger and the second one applying an elliptic pattern normalization of the region of interest. The experimental results confirm the negative impact of longitudinal rotation on the recognition performance and prove that its correction noticeably improves the performance again.**

ACKNOWLEDGEMENTS

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Program under Grant 700259, and in part by the FFG KIRAS Project AUTFingerATM under Grant 864785.

The original paper is published as:

Bernhard Prommegger, Christof Kauba, Michael Linortner, Andreas Uhl. Longitudinal Finger Rotation - Deformation Detection and Correction. IEEE Transactions on Biometrics, Behavior, and Identity Science 1:2, pp. 123-138, 2019.

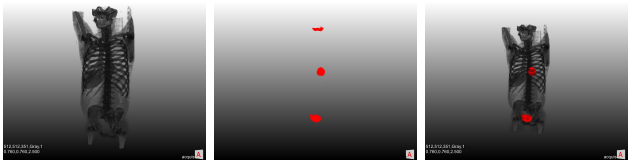
[1]Bernhard Prommegger, Christof Kauba, Michael Linortner, and Andreas Uhl are with the Department of Computer Sciences, University of Salzburg, Jakob-Haringer-Str. 2, 5020 Salzburg, Austria uhl@cosy.sbg.ac.at

Learning from the Truth: Fully Automatic Ground Truth Generation for Training of Medical Deep Learning Networks*

Christina Gsaxner[1,2,3], Peter M. Roth[1], Jürgen Wallner[2,3] and Jan Egger[1,2,3]

I. PROBLEM STATEMENT AND MOTIVATION

Automatic medical image analysis has become an invaluable tool in the different treatment stages of diseases. Especially medical image segmentation plays a vital role, since segmentation is often the initial step in an image analysis pipeline. Convolutional neural networks (CNNs) have rapidly become a state of the art method for many medical image analysis tasks, such as segmentation. However, in the medical domain, the use of CNNs is limited by a major bottleneck: the lack of training data sets for supervised learning. Although millions of medical images have been collected in clinical routine, relevant annotations for those images are hard to acquire. Generally, annotations are created (semi-)manually by experts on a slice-by-slice basis, which is time consuming and tedious. Therefore, available annotated data sets are often too small for deep learning techniques.



(a) CT data (b) PET data (c) PET/CT

Fig. 1: 3D image data obtained from CT, PET, and combined PET/CT. In CT data (a), contrast for soft tissue is poor. PET data in (b) shows metabolical active regions. A PET/CT scan (c) allows to properly assign active regions anatomically.

II. METHOD OVERVIEW

To overcome these problems, we proposed a novel method to automatically generate ground truth annotations by exploiting positron emission tomography (PET) data acquired simultaneously with computed tomography (CT) scans in combined PET/CT systems [3], [4]. PET/CT scanning combines functional information from PET with anatomical information from CT. Soft tissue, which exhibits limited contrast in CT, shows up distinctively in a PET scan if it is metabolically active (see Figure 1). However, PET scanning increases radiation exposure for the patient and is not as widely available as CT, making approaches which detect significant structures in CT scans alone compelling. We utilized the high contrast in PET scans to extract ground truth segmentations for corresponding structures of interest in CT data, enabling automatic detection of these structures in CT alone by training CNNs with the generated data. As a structure of interest we chose the urinary bladder, since the radio-tracer used for PET imaging always accumulates in it. The ground truth is acquired fully automatically from PET by a thresholding algorithm. Furthermore, affine transformations and noise are applied to the generated data for data augmentation [2]. Using these data, we trained and tested different CNN architectures for image segmentation, which are based on fully convolutional networks [5] and Deeplab [1].

III. RESULTS AND DISCUSSION

Qualitative segmentation results predicted with our best performing architecture are shown in Figure 2. Quantitatively, we achieve a maximal mean Dice coefficient of 81.9%. This results are very satisfactory, considering that no manually annotated training data was used in our studies. Our approach presents a promising tool for automatic CT analysis and can be generalized to all applications of PET/CT. In particular, in future research we aim to extend our method to tumor detection.

Fig. 2: Qualitative segmentation results. The prediction is shown in red, while the ground truth is outlined in green.

*Supported by FWF KLI 678-B31 (enFaced), COMET K-Project 871132 (CAMed) and the TU Graz Lead Project (Aortic Dissection).

[1]Institute of Computer Graphics and Vision, TU Graz, Austria
[2]Computer Algorithms for Medicine Laboratory, Graz, Austria
[3]Department of Oral and Maxillofacial Surgery, Medical University of Graz, Austria

REFERENCES

[1] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE PAMI*, 40(4):834–848, 2018.

[2] C. Gsaxner, B. Pfarrkirchner, L. Lindner, N. Jakse, J. Wallner, D. Schmalstieg, and J. Egger. Exploit 18F-FDG enhanced urinary bladder in PET data for deep learning ground truth generation in CT scans. In *Proc. SPIE Medical Imaging*, 2018.

[3] C. Gsaxner, B. Pfarrkirchner, L. Lindner, A. Pepe, J. Wallner, P. M. Roth, and J. Egger. PET-Train: Automatic Ground Truth Generation from PET Acquisitions for Urinary Bladder Segmentation in CT Images using Deep Learning. In *Proc. BMEiCON*, 2018.

[4] C. Gsaxner, P. M. Roth, J. Wallner, and J. Egger. Exploit fully automatic low-level segmented PET data for training high-level deep learning algorithms for the corresponding CT data. *PLOS ONE*, 14(3):1–20, 2019.

[5] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Proc. CVPR*, 2015.

PRNU-based Finger Vein Sensor Identification in the Presence of Presentation Attack Data

Babak Maser[1,*], Dominik Söllinger[1,*] and Andreas Uhl[1]

Abstract— **We examine the effectiveness of the Photo Response Non-Uniformity (PRNU) in the context of sensor identification for finger vein imagery. Experiments are conducted on eight publicly-available finger vein datasets. We apply a Wiener Filter (WF) in the frequency domain to enhance the quality of PRNU estimation and noise residual, respectively, and we use two metrics to rank PRNU similarity, i.e. Peak-to-Energy (PCE) and Normalized Cross Correlation (NCC). In the experiments, we include a dataset consisting of both real finger vein data and captured artifacts produced to assess presentation attacks. We investigate the impact of this situation on sensor identification accuracy and also try to discriminate spoofed images from non-spoof images varying decision thresholds. Results of sensor identification for finger vein imagery is encouraging, the obtained scores for classification accuracies are between 97% to 98% for different settings. Interestingly, selecting particular decision thresholds, it is also possible to discriminate real data from artificial data as used in presentation attacks.**

I. INTRODUCTION

Human identification is one of the main goals of biometric technology and the corresponding research area. Biometric systems utilize a human's physical or behavioral characteristics for authentication. Not only companies or governmental organizations do rely on biometric technology to provide secure authentication, but also everyday technology (e.g. smartphones, laptops, entrance systems) applies this technology to an increasing extent. Nevertheless, biometric traits also set new challenges in terms of maintaining the security and integrity of biometric data. While the (cryptographic) key material used in traditional authentication methods like PINs, passwords, smart-cards etc. can usually be changed once compromised, a person's biometric trait usually remains stable. Therefore, once biometric features are leaked, stolen or adopted many different attack scenarios become realistic and many corresponding attack vectors have been identified.

In order to capture a certain biometric trait, we need digital hardware which is typically termed "sensor" that has the technological capability to acquire the corresponding data suited to uniquely identify humans, which is a near-infrared (NIR) camera used with NIR illumination to visualize the structure and vascular pattern of human finger veins. The underlying imaging principle relies on NIR light absorption of human blood, thus, vessels appear dark in such images. For the security of a biometric system, the integrity of the authentication process is of vital interest. In this context, it is required to ascertain that imagery used for authentication has been indeed captured by the proper sensor, and has not entered the system in the context of an injection attack. At this point, we encounter passive media security techniques termed "digital image forensics" which can be used for this purpose. Similar to bullet scratches that allow forensic experts to match a bullet to a particular barrel with high reliability to be accepted even in court, these techniques can be eventually used to identify a sensor which has captured a finger vein image.

In this paper, we use an approach which is based on the photo-response non-uniformity (PRNU) [1] method. PRNU is an intrinsic property of every digital sensor caused by different sensitivity of pixels to light due to inhomogeneity of silicon wafers and imperfections during the sensor manufacturing process. PRNU can be interpreted as the telltale of "scratches" in images which can identify the originating sensor and discriminates images taken by different sensor instances.

Prior work in biometric sensor identification has shown that the PRNU method can be considered a well-suited method to identify a sensor in different fields of biometrics, so far considered for fingerprint [2] as well as iris [3], [4], [5], [6] sensors, respectively.
Prior work shows that the PRNU method can be considered a well-suited method to identify a sensor in different fields of biometric, e.g. Bartlow *et al.* studied the application of hardware fingerprinting based on PRNU noise analysis of biometric fingerprint devices for sensor identification [2], also PRNU has been used in the context of iris sensors in [3], [7], [8]. Alternatively to PRNU, also classical texture-oriented features have been used to identify a particular sensor model in the context of iris recognition [4], [5], [6]. Finally, in [9], Schuch *et al.* studied the applicability of a CNN-based and conventional approach on database bias as distinguishing property for the origin of a fingerprint. Also Marra, Francesco, *et al.* in [10] proposed a CNN-based algorithm improve the iris sensor model identification for benefit of the sensor interoperability.

As biometric authentication becomes a standard replacement for the traditional way of authentication in many areas, various attacks have been used to fool sensors with prerecorded data or artifacts. One of the approaches to mislead and deceive biometric sensors is the so-called "presentation attack" or sensor spoofing [11].

[1] Babak Maser, Dominik Söllinger, and Andreas Uhl are with the Department of Computer Sciences, University of Salzburg, Jakob-Haringer-Str. 2, 5020 Salzburg, Austria uhl@cosy.sbg.ac.at

[*] Both authors contributed equally

In this attack, a copy of a biometric trait is fabricated artificially and presented to the sensor. Intensive work has been done to develop techniques to detect presentation attacks [12], and for evaluation purposes, datasets consisting of artificial biometric data resulting from sensing such artifacts have been established and published.

This work is organized as follows: Section II gives an overview of techniques used for PRNU extraction and enhancement. Section III introduces the datasets, explains the different experimental settings as well as the evaluation workflow in detail. Finally, experimental results are provided in section V followed by a conclusion (section VII).

II. Technical Approach

There are different ways how to compute the PRNU, we used a method proposed by Fridrich in [13], the method describes how to estimate the PRNU image from set of images taken by the same camera, the PRNU estimator is derived using maximum likelihood estimator (MLE),the MLE is modeled from the simplified sensor output model [13]. Thus the PRNU factor is obtained as follow:

$$\hat{K} = \sum_{i=1}^{N} R_i I_i \Big/ \sum_{i=1}^{N} I_i^2 \tag{1}$$

where PRNU factor is denoted by $\hat{K}$ which is noise-like signal responsible for the PRNU. I_i is an image and R_i is the noise residual of an image which is obtained by (eq. 2), note that i stands for the i *th* image out of N images which have been taken from a particular sensor.

The residual image R_i can be calculated by subtracting an original image from a denoised image obtained using e.g. a wavelet denoising filter from an original image:

$$R_i = I_i - F(I_i) \tag{2}$$

where F denotes the denoising method, in our case the denoised image is obtained in the wavelet domain applying a 4-Level Wavelet decomposition using the Daubechies 8-tap wavelet filter, we empirically set $\sigma_0 = 3$. Eventually, a Wiener Filter (WF) [14] is applied additionally.

To detect whether the Residual of an image I (R_I) is taken by the sensor with PRNU estimator $\hat{K}$, we use normalized cross-correlation (NCC):

$$\rho_{[R_I, I\hat{K}]} = NCC(R_I, I\hat{K}) \tag{3}$$

NCC has been also proposed in [13]. Apart from NCC, we investigate the effect of using the Peak Correlation Energy (PCE) as another similarity metric [13] in this paper.

III. Experimental Design

A. *Datasets*

In this paper, we have assembled the following publicly available datasets to evaluate the performance and effectiveness of the proposed approach. The number of images in each dataset is not equal, thus to keep the sample dataset balance we choose an equal number of images from each dataset. Hence we have chosen the first 120 images from each dataset for our experiments. The following listing provides a description of the datasets.

- **SDUMLA-HMT (SDUMLA)** - Images are selected from the first 20 clients, images of the dataset [15] are stored in BMP format with 320×240 pixels in size.
- **IDIAP VERA (IDIAP-REAL)** - Images of the dataset [16] are stored in PNG format with a size of 250×665. The images are taken from 60 clients of the IDIAP-REAL sub-dataset.
- **IDIAP VERA (IDIAP-SPOOF)** - Images of the dataset [16] are stored in PNG format with a size of 250×665. The images are taken from 60 individuals of the IDIAP-Spoof sub-dataset.
- **FV-USM** - Images of the dataset [17] are stored in JPEG format with a size of 480×640. The selected subset is taken from the first 30 clients.
- **MMCBNU_6000 (MMCBNU)** - Images of the dataset [18] are stored in BMP format with a size of 640×480. The selected subset is chosen from the first 20 clients.
- **PLUS-FV3-Laser-Palmar (Palmar)** - Images of the dataset [19] are stored in PNG format with a size of 600×1024. The selected subset has been chosen from the first 20 clients.
- **THU-FVFDT** - Images of the dataset [19] are stored in PNG format with a size of 600×1024. The selected subset is composed of images from the first 20 clients.
- **UTFVP** - Images of the dataset [20] are stored in PNG format with a size of 672×380. The selected subset is composed of images of the first 20 clients.
- **HKPU-FV** - Images of the dataset [21] are stored in BMP format with a size of 513×256. The selected subset is composed of images of the first 60 clients.

B. *Cropping*

The primary goal of this work is not only to study the general applicability of PRNU-based sensor identification for finger vein images but also to investigate the effect of the presence of spoofed images resulting from a presentation attack on the sensor identification performance.
We assume that fingerprints generated from uncorrelated data (in order to facilitate the out-averaging of image-content related high-frequency content) are better suited for sensor identification than fingerprints generated from

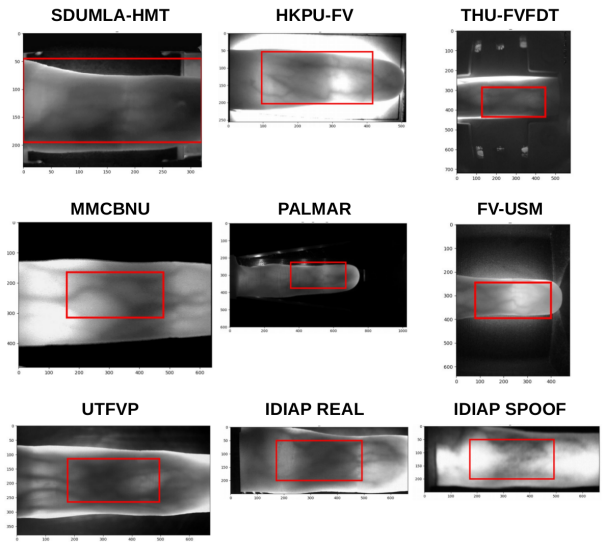


Fig. 1. Sample Patches: Center 320×150

correlated data. Therefore, we expect good performance for regions containing the biometric trait due to a better variability of the image content instead of image regions covered by sensor parts only. Thus, we decided to focus our experiment on a region which contains the biometric trait. The selected cropping is termed as **Center 320×150** as the image region was taken from the center and contains mostly finger vein texture (except for some data sets like SDUMLA-HMT and PALMAR, see Figure 1 for example croppings).

IV. Workflow and Scenario

We applied a 4 fold cross-validation framework for all eight datasets to examine the proposed methods. In each fold, we feed $3/4$ of the query dataset (i.e. 90 images) to the model to determine the PRNU estimation by MLE ($\hat{K}$), subsequently, the estimated PRNU will be enhanced by WF or no enhancement will be applied. The images of the other datasets, as well as $1/4$ of the query dataset, are fed into the model to compute the residuals (R_i), and again, either WF is applied to the residuals or *No Enhancement* is considered. The estimated PRNU, as well as the residuals, are fed into the classification unit, and as it is mentioned in Section II we use two similarity metrics for sensor identification (NCC and PCE).

Recall that IDIAP-Real dataset and IDIAP-Spoof dataset are captured with the same sensor, the difference is that for the former, human fingers are imaged, while for the latter, presentation artifacts are imaged. The AUC-ROC score and the Precision-Recall score for the IDIAP-Real data are obtained by estimating $\hat{K}$ from images of the IDIAP-Real dataset only, while the residuals are taken from both the IDIAP-Real dataset and the IDIAP-Spoof dataset respectively. The IDIAP-Spoof results are generated in the same manner, but $\hat{K}$ is computed from IDIAP-Spoof data only.

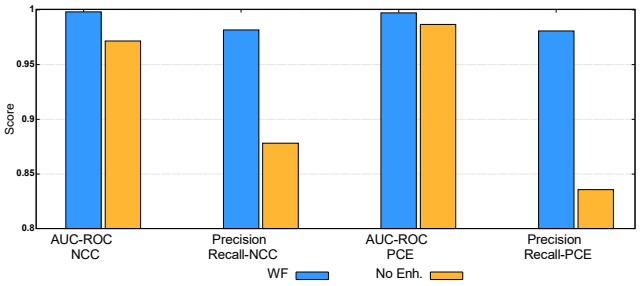


Fig. 2. Impact of applying Wiener filter and No Enhancement on AUC-ROC and Precision-Recall scores using NCC and PCE similarity metrics

Similarity metric	Performance measurement	WF	No Enh.
NCC	AUC ROC	0.998	0.971
NCC	AUC Precision Recall	0.982	0.878
PCE	AUC ROC	0.997	0.986
PCE	AUC Precision Recall	0.980	0.835

TABLE I
IMPACT ON AUC ROC AND PRECISION RECALL USING NCC AND PCE BY APPLYING WF AND NO ENH.

V. Experiments and Results

Our aim is to investigate the following topics as our primary interest:

- Feasibility of PRNU-based sensor identification using the proposed method for finger vein imagery,
- Comparison between PCE and NCC as similarity assessment methods in the context of finger vein data,
- Assessment of the influence of applied WF and
- Investigation of the influence of the presence of presentation attack data.

To evaluate and analyze the proposed method we provide the AUC-ROC score and the Precision-Recall score for all sensors/data sets.
In Figure 2 and Table I, we display the achieved AUC-ROC and the Precision-Recall scores by taking the average over all sensor class scores.

We find that the Wiener Filter plays a significant role in sensor identification accuracy[1]. This behavior was somehow expected because the WF suppresses periodic artifacts and it has been observed on other data that the resulting PRNU and residuals have higher quality. We observe the same behavior in Figures 3, 4, 5, 6 which show non-averaged but per-sensor results.

When comparing PCE and NCC we find that for data after the application of WF there is hardly any difference. When considering non-enhanced data, there are some differences, but these are not consistent when considering AUC-ROC scores and Precision-Recall scores.

[1]In this paper, we use the term *accuracy* for the AUC ROC score and the AUC ROC Precision-Recall score

Figures 3, 4, 5, 6, now detailing results per dataset, confirm that the PRNU-based approach is well suited to identify sensors overall. However, there are some results where certain configurations turn out not to deliver satisfying performance.

Figure 3 displays stable results with excellent accuracy in case of WF application, while scores are down to 0.75 for one sensor (i.e. THU-FVFDT) in case no enhancement is applied.

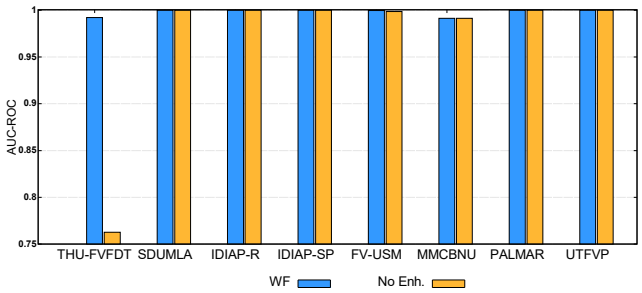


Fig. 3. AUC ROC for NCC

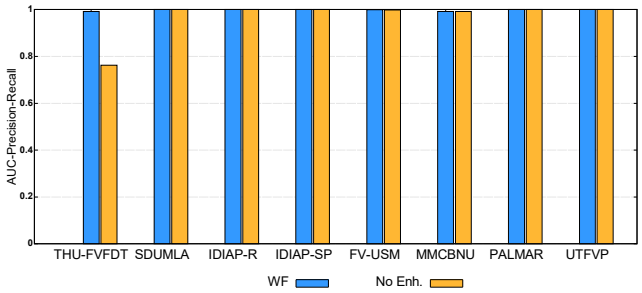


Fig. 4. AUC Precision-Recall for NCC

This result is confirmed in Figure 4, which shows the results of AUC-Precision-Recall for all sensors. Again the worst result is seen for sensor THU-FVFDT in case of no enhancement is applied. We observe that the score for the MMCBNU sensor is slightly improved compared to AUC-ROC scores in Figure 3.

When looking at PCE results (Figure 5 and Figure 6), these seem to less stable as compared to their NCC counterparts at first sight. However, in Figure 5 the scale on the y-axis is fairly different as the minimum score value is 0.965. So basically all these results are excellent and the differences do not matter. Figure 6 reveals a very poor result in case of UTFVP and no enhancement applied. Here is the score is down to 0.3! In this setting again the superiority of applying WF is confirmed.

Overall, we note that there are some lower score values in all settings but applying WF enhancement and using NCC as similarity measure prevents significant inaccuracies in any case.

VI. Detection of spoofed images

As described in section III-A, the IDIAP dataset consists of real finger vein images as well as of spoof images which

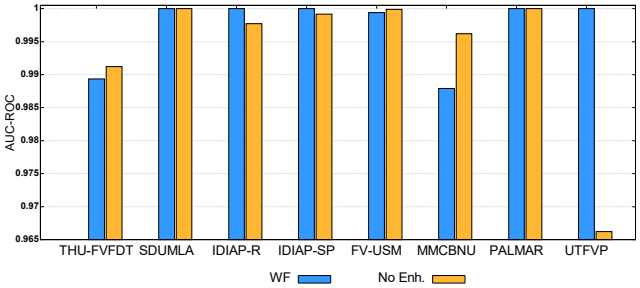


Fig. 5. AUC ROC for PCE

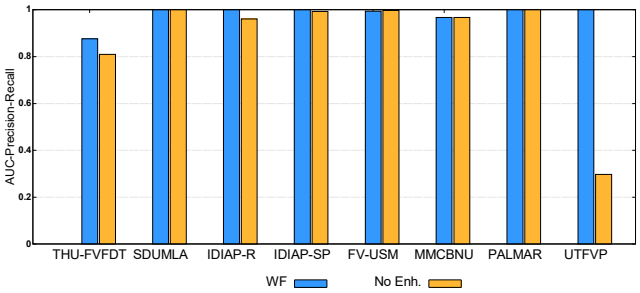


Fig. 6. AUC Precision-Recall for PCE

were generated by imaging presentation attack artifacts with the same sensor. The results of the previous section clearly demonstrate that these spoof images do not prevent a correct classification of the sensor as both, real and spoof images can be discriminated well from finger vein images acquired by one of the other sensors. However, there might still be subtle differences present, which might lead to slight differences in the PRNU which could be exploited to detect a presentation attack using corresponding artifacts. To understand whether PRNU can be also used for this purpose, we analyze if there is a certain NCC-threshold that allows us to discriminate between IDIAP-Real and IDIAP-Spoof images.

Table II shows the accuracy of assigning IDIAP-Real and IDIAP-Spoof images to the appropriate class for different thresholds. The PRNU computed from IDIAP-Real images is computing NCC to the respective residuals. We can observe that for a low threshold, all IDIAP-Real images are classified correctly. When we increase the threshold to 0.5, we classify 98% of IDIAP-Real images correctly and IDIAP-Spoof images are still hardly correctly classified (7%). For threshold 0.6, IDIAP-Real images achieve 92% accuracy while IDIAP-Spoof accuracy is 48%. When increasing the threshold further, accuracy for IDIAP-Real images are further reduced while IDIAP-Spoof image classification accuracy is almost perfect.

Table III again shows the accuracies of classifying IDIAP-Real and IDIAP-Spoof images for different thresholds, respectively. However, in this case, IDIAP-Spoof images are used to compute the PRNU.

As we can see, when choosing a low threshold at 0.3, 0.4 and 0.5, IDIAP-Spoof images are correctly classified, but the accuracy for IDIAP-Real images is almost 0. However, once

Threshold (NCC)	IDIAP-REAL	IDIAP-SPOOF
0.3	1.0	0.0
0.4	1.0	0.01
0.5	0.98	0.07
0.6	0.92	0.48
0.7	0.34	0.99
0.8	0.0	1.0
0.9	0.0	1.0

TABLE II
DISCRIMINATION AMONG IDIAP-REAL AND IDIAP-SPOOF IMAGES FOR DIFFERENT THRESHOLDS BASED ON IDIAP-REAL PRNU

the threshold increases to 0.6, it can be observed that most (99%) IDIAP-spoof images are still treated correctly, while also 74% of the IDIAP-Real images are correctly classified. Furthermore, if we increase the threshold to 0.7 we can see that 100% of IDIAP-Real images are correctly classified while 44% of the IDIAP-Spoof images are detected correctly as spoof images. Increasing the threshold further entirely disables classification for IDIAP-spoof images. Overall, we cannot find any threshold in Tables II and III to perfectly discriminate both datasets, but for some settings, a PRNU-based distinction seems to be realistic.

Threshold (NCC)	IDIAP-REAL	IDIAP-SPOOF
0.3	0.00	1.00
0.4	0.00	1.00
0.5	0.02	1.00
0.6	0.74	0.99
0.7	1.00	0.44
0.8	1.00	0.00

TABLE III
DISCRIMINATION AMONG IDIAP-REAL AND IDIAP-SPOOF IMAGES FOR DIFFERENT THRESHOLDS BASED ON IDIAP-SPOOF PRNU

VII. CONCLUSION

This work studies the applicability of PRNU-based sensor identification methods for finger vein images in the context of biometric systems. The result clearly shows that this approach is well-suited, in particular, the Wiener filter is used as an enhancement technique. Finally, we observe that the PRNU-based approach might be also suited for presentation attack, aka sensor spoofing, detection.

ACKNOWLEDGEMENTS

This work has received funding from the European Union's Horizon 2020 research and innovation program under grant agreements No. 700259 (PROTECT) and No. 690907 (IDENTITY), respectively. The work was also funded by the Austrian Research Promotion Agency, FFG KIRAS project AUTFingerATM under grant No. 864785.

REFERENCES

[1] Jan Lukas, Jessica Fridrich, and Miroslav Goljan. Digital camera identification from sensor pattern noise. *IEEE Transactions on Information Forensics and Security*, 1(2):205–214, 2006.

[2] Nick Bartlow, Nathan Kalka, Bojan Cukic, and Arun Ross. Identifying sensors from fingerprint images. In *Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on*, pages 78–84. IEEE, 2009.

[3] Luca Debiasi and Andreas Uhl. Blind biometric source sensor recognition using advanced prnu fingerprints. In *Signal Processing Conference (EUSIPCO), 2015 23rd European*, pages 779–783. IEEE, 2015.

[4] Christof Kauba, Luca Debiasi, and Andreas Uhl. Identifying the origin of iris images based on fusion of local image descriptors and prnu based techniques. In *Biometrics (IJCB), 2017 IEEE International Joint Conference on*, pages 294–301. IEEE, 2017.

[5] Luca Debiasi, Christof Kauba, and Andreas Uhl. Identifying iris sensors from iris images. *Iris and Periocular Biometric Recognition*, 5:359, 2017.

[6] Susan El-Naggar and Arun Ross. Which dataset is this iris image from? In *WIFS*, pages 1–6, 2015.

[7] Nathan Kalka, Nick Bartlow, Bojan Cukic, and Arun Ross. A preliminary study on identifying sensors from iris images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 50–56, 2015.

[8] Andreas Uhl and Yvonne Höller. Iris-sensor authentication using camera prnu fingerprints. In *Biometrics (ICB), 2012 5th IAPR International Conference on*, pages 230–237. IEEE, 2012.

[9] Patrick Schuch, Jan Marek May, and Christoph Busch. Estimating the data origin of fingerprint samples. In *2018 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–6. IEEE, 2018.

[10] Francesco Marra, Giovanni Poggi, Carlo Sansone, and Luisa Verdoliva. A deep learning approach for iris sensor model identification. *Pattern Recognition Letters*, 113:46–53, 2018.

[11] Amrit Pal Singh Bhogal, Dominik Söllinger, Pauline Trung, Jutta Hämmerle-Uhl, and Andreas Uhl. Non-reference image quality assessment for fingervein presentation attack detection. In *Scandinavian Conference on Image Analysis*, pages 184–196. Springer, 2017.

[12] Sébastien Marcel, Mark S. Nixon, and Stan Z. Li, editors. *Handbook of Biometric Anti-Spoofing - Trusted Biometrics under Spoofing Attacks*. Advances in Computer Vision and Pattern Recognition. Springer, 2014.

[13] Jessica Fridrich. Digital image forensics. *IEEE Signal Processing Magazine*, 26(2), 2009.

[14] M Kivanc Mihcak, Igor Kozintsev, and Kannan Ramchandran. Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising. In *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, volume 6, pages 3253–3256. IEEE, 1999.

[15] Y. Yin, L. Liu, and X. Sun. SDUMLA-HMT: A Multimodal Biometric Database. In *The 6th Chinese Conference on Biometric Recognition (CCBR 2011)*, volume 7098 of *Springer Lecture Notes on Computer Science*, pages 260–268, 2011.

[16] Pedro Tome, Matthias Vanoni, and Sébastien Marcel. On the vulnerability of finger vein recognition to spoofing. In *IEEE International Conference of the Biometrics Special Interest Group (BIOSIG)*, September 2014.

[17] M. S. M. Asaari and B. A. Rosdi S. A. Suandi. Fusion of band limited phase only correlation and width centroid contour distance for finger based biometrics. *Expert Systems with Applications*, 41(7):3367–3382, 2014.

[18] Yu Lu, Shan Juan Xie, Sook Yoon, Zhihui Wang, and Dong Sun Park. An available database for the research of finger vein recognition. In *Image and Signal Processing (CISP), 2013 6th InternationalCongress on Image and Signal Processing (CISP 2013)*, volume 1, pages 410–415. IEEE, 2013.

[19] Christof Kauba, Bernhard Prommegger, and Andreas Uhl. Focussing the beam - a new laser illumination based data set providing insights to finger-vein recognition. In *Proceedings of the IEEE 9th International Conference on Biometrics: Theory, Applications, and Systems (BTAS2018)*, pages 1–9, Los Angeles, California, USA, 2018.

[20] B.T. Ton and R.N.J. Veldhuis. A high quality finger vascular pattern dataset collected using acustom designed capturing device. In *International Conference on Biometrics, ICB 2013*. IEEE, 2013.

[21] Ajay Kumar and Yingbo Zhou. Human identification using finger images. *IEEE Transactions on Image Processing*, 21(4):2228–2244, 2012.

GMM Interpolation for Blood Cell Cluster Alignment in Childhood Leukaemia

Roxane Licandro[1,2], Konstantin Miloserdov[2], Michael Reiter[2] and Martin Kampel[2]
licandro@cvl.tuwien.ac.at

***Abstract*— The accurate quantification of cancer (blast) and non-cancer cells in childhood leukaemia (blood cancer) is a key component in assessing the treatment response and to guide patient specific therapy. For this classification task, cell specific biomarker expression levels are estimated by using flowcytometry measurements of multiple features of single blood cells. For the automated distinction between blasts and non-blasts a main challenge are data shifts and variations in the high-dimensional dataspace caused by instrumental drifts, inter patient variability, treatment response and different machine characteristics. In this work we present a novel alignment scheme for stable (non-cancer) cell populations in flowcytometry using Gaussian Mixture Models (GMM) as data representation format for the cell clusters' probability density function and a Wasserstein interpolation scheme on the manifold of GMM. The evaluation is performed using a dataset of 116 patients with acute lymphoblastic leukaemia at treatment day 15. Classification results show an improved normalization performance using Wasserstein metric compared to two other metrics with a mean sensitivity of 0.97 and mean f-score of 0.95.**

I. INTRODUCTION

For the assessment of leukaemia (blood cancer) treatment response in the clinical routine, cell specific immunophenotypes of blood or bone marrow samples are measured using the FlowCytoMetry (FCM) technique [1], [4]. Therefor, in a staining step a combination of specific fluorescence-labeled antibodies is used to mark antigens on the surface of a cell. The expression of antigens on a surface varies among different blood cell types, which is used to identify blasts and non-blasts. In a subsequent step, every single cell is put in a fluid stream and FCM lasers of different wavelengths are used to measure physical properties (cell size and granularity) and the fluorescence pattern of the antibodies attached to the cell [12]. In Figure 1 a sample obtained by an FCM is illustrated. Every dot represents a cell (red cells refer to blast cells), where its position is determined by the features measured (in this case) for the expression of the antibodies CD 10 and CD 45. Up to 30 features can be measured and observed at once. Blasts are manually identified using a gating hierarchy, which consists of a defined procedure observing specific combinations of features in a two-dimensional representation and by drawing polygons (gates) to extract and mark cell clusters of interest in every step. The identified events of interest of a gating step serve as input of the subsequent gate in the hierarchy. This manual procedure strongly depends on the experience and skills of the operator, is highly time-consuming and subjective. [7]

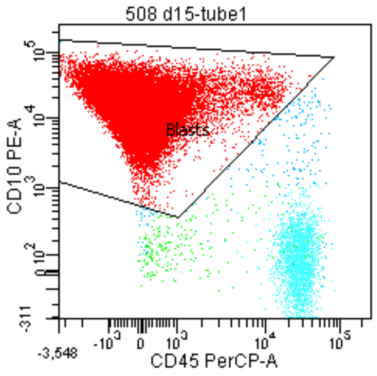


Fig. 1. Illustration of a flowcytometer sample. The labeling is obtained by manual gating (drawing polygons (gates) around populations of interest) in the two dimensional feature space (CD10, CD45). [Best viewed in color]

A. Challenges

The main challenge in gating FCM data is the detection of small cell populations, e.g. cancer cells in late therapy stages composing around 0.1% of all cells observed ($3*10^5$ - 10^6). This is further aggravated by the influence of treatment or age on the regeneration status of bone marrow precursors [5]. A further challenge forms shifts and deformations of the clusters observed in the high-dimensional feature space caused by different machine types, corresponding instrumental drifts and lab varying standard operating procedures [10]. Thus, it is particularly important to obtain a normalized representation of samples to be able to compare and identify clusters of interest and to perform patient or population based longitudinal analysis automatically over therapy time and multiple centers.

B. Related Work

A key issue for multi-center studies as well as in studies where data is acquired with different flowcytometers, is the normalization and alignment of cell population clusters to improve the automated blast identification process. In recent approaches ([8], [11], [7]) Gaussian Mixture Models are

[1] Medical University of Vienna, Department of Biomedical Imaging and Image-guided Therapy, Computational Imaging Research (CIR).
[2] TU Wien, Institute of Visual Computing and Human-Centered Technology, Computer Vision Lab (CVL).
*This work has been supported by the European Commission Project AutoFLOW (http://www.autoflow-project.eu/) and the Computer Vision Club. We thank Markus Diem and Florian Kleber at the CVL for their valuable input. We thank Michael Dworzak at St. Anna Childrens' Hospital Vienna and Angela Schumich and Susanne Suhendra at LabDIA GmbH, for their kind support and providing the flowcytometry data and annotations.

used to model the probability density function of blood cell clusters to assign a biologically meaningful label to every observed cell automatically. Various approaches for the registration and interpolation of GMM distributions have been proposed in the computer vision and pattern recognition community: Jian et al. [2] present a robust registration scheme of GMM for point sets using a L2 distance metric, since it provides a closed-form expression for GMM and needs no specification of tuning parameters. However, no intermediate evolution step guarantees that the interpolation between two GMMs lies in the manifold of K-component GMM as well as minimizing the L2 distance for registration is prone to instability due to many local optima. Kim et al. [6] address this problem by introducing cross-entropy instead of the L2 distance and provide a closed form solution with no numerical difficulties. In [9] Orlova et al. propose an alignment approach, based on quadratic formcluster matching *QF Match* for flow and mass cytometry data. They provide a multivariate extension of the quadratic form distance for cell cluster alignment. In [10] Orlova et al. used the Earth Mover's Distance (also known as Wasserstein Distance (WD)) as a metric for comparing the expression levels of biomarker in cell populations measured by FCM. Using WD as a metric allows to distinguish small shifts caused by e.g. instrument drifts from biologically significant differences. A key benefit of WD over other metrics (quadratic form, bipartite matching) is that the WD score between two samples involves the magnitude of change and the proportion of cells whose antigen expression has changed.

C. Contribution

The contribution of this work is three fold: Firstly, we provide an optimal transport technique for FCM data on the manifold of Gaussian Mixture Models using a Wasserstein metric. Kim et al. [6] propose an interpolation scheme on the manifold on k-component GMM using the Kullback Leibler Divergence (KLD) as a metric. We extended the GMM interpolation approach by introducing the Wasserstein metric as a distance measure between two GMMs. We were inspired in doing this by the approach of Chen et al. [3], who provided an interpolation scheme based on the Wasserstein metric for GMM on the space of Gaussian distributions, so it can be seen as a discrete measure and retains the Gaussian mixture structure. This formulation showed efficiency for high dimensional GMMs with a small number of components, but (to our knowledge) has not been used for the alignment of cell distributions acquired by FCM. Secondly, we provide an extensive evaluation scheme against two different metrics. Thirdly, we used the proposed alignment strategy on FCM data to improve the non-blast alignment in childhood Acute Lymphoblastic Leukaemia (ALL).

This paper gives an overview of the methodology proposed in Section II. The dataset used and evaluation results are presented in Section III and the conclusion of this work and possibilities for future work are summarized in Section IV.

II. METHODOLOGY

In this section the proposed normalization strategy for FCM data is presented. For the representation of the probability density function $p(x|.)$ of non blast cells x a N component Gaussian Mixture Model (GMM) $S(x)$ is used as a weighted sum of N Gaussian distributions $S_i(x) = \mathcal{N}(x|\mu_i, \sigma_i)$ as expressed in Equation 1. θ is the set of parameters required to parameterize a Gaussian component i in a GMM (cf. Equation 2), where w_i is the Gaussian weighting parameter, μ_i the mean (cf. Equation 3) and Σ_i the covariance (cf. Equation 4).

$$S(x) = p(x|\theta) = \sum_{i=1}^{N} w_i S_i(x) \tag{1}$$

$$\theta = \{w_i, \mu_i, \Sigma_i\}_{i=1}^{N} \tag{2}$$

$$\mu_i = \mathbb{E}_{S_i(x)}[x] \tag{3}$$

$$\Sigma_i = \mathbb{E}_{S_{(x)}}[(x-\mu_i)-(x-\mu_i)^T] \tag{4}$$

A. GMM based Optimal Transport

The optimal transport problem is the transport of a source distribution S(x) of a mass x on a manifold $\mathcal{M}_{GMM}$ of N component GMMs, in a way that it is transformed into the target distribution $T(x) \in \mathcal{M}_{GMM}$ without loss of mass [3]. Therefore we can define an optimization function and transport plan γ. Optimizing γ refers to the finding of an optimal interpolation path on $\mathcal{M}_{GMM}$ by minimizing distances D between the Gaussian components $S_i(x), T_j(x)$ in $S(x)$ and $T(x)$ (cf. Equation 5). In Figure 2 a schematic illustration of the GMM based optimal transport problem for FCM data is visualised. Two physical features (SS INT and FS INT) are used as dimensions for visualisation. Cyan ellipses correspond to Gaussian components of the source GMM and black ellipses to the target GMM. In the optimal case, components of the source GMM overlay with the target GMM components after alignment (left image). Only non-cancer cells are visualised.

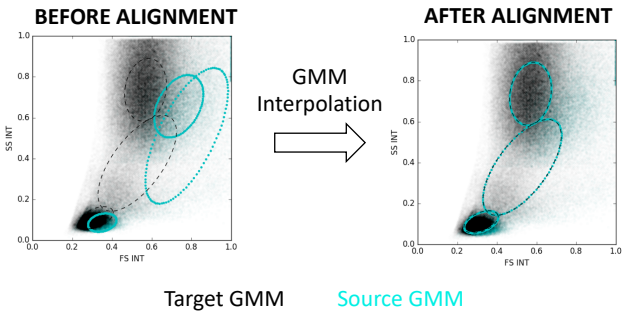


Fig. 2. Schematic illustration of the GMM based Optimal Transport problem with flowcytometry data. Only 2 measured features (SS INT and FS INT) are visualised. Black ellipses correspond to Gaussian components of the target GMM and cyan to the source GMM. Only non-cancer cells are visualised. [Best viewed in color]

$$S^* = argmin_{S(x)\in\mathscr{M}_{GMM}} \sum_{j=1}^{M} D(S(x)||T_j(x)) \tag{5}$$

In this work the Wasserstein metric (cf. Equation 6 [3]) is minimized between the Gaussian components of the source and target GMMs. In the following, i = 1,...N and j = 1,...,M denote the numbers of Gaussian components in the source and in the target GMM respectively. We evaluate the performance of the WD against the Kullback Leibler Divergence (KLD) (cf. Equation 7) and Bhattacharyya Bound (BB) (cf. Equation 8).

$$W^2 = \|\mu_i - \mu_j\|^2 + tr[\Sigma_i + \Sigma_j - 2\sqrt{(\sqrt{\Sigma_i})\Sigma_j(\sqrt{\Sigma_i})}] \tag{6}$$

$$KL = \frac{1}{2}k\log 2\pi + \log|\Sigma_j| + tr[\Sigma_j^{-1}\Sigma_i] + (\mu_i - \mu_j)^T\Sigma_j^{-1}(\mu_i - \mu_j) \tag{7}$$

$$BB = \frac{1}{8}(\mu_j - \mu_i)^T(\frac{\Sigma_i + \Sigma_j}{2})^{-1}(\mu_j - \mu_i) + \frac{1}{2}\ln\frac{|\frac{\Sigma_i+\Sigma_j}{2}|}{|\sqrt{|\Sigma_i||\Sigma_j|}} \tag{8}$$

Kim et al. [6] propose Expectation Maximization (EM) to estimate the parametrization θ of the transported GMM S^* by minimizing the KL-divergence. We adapted this approach for FCM data by estimating the likelihood of the WD (*responsibility*) between every Gaussian component $S_i(x)$ and $T_j(x)$ (cf. Equation 9) in the E-step to obtain the transportation matrix $\gamma_{ij} \in \mathbb{R}^{N\times M\times d}$, where d refers to the number of features observed per cell.

$$\gamma_{ij} = \frac{\delta_{i,j}}{\tau_i} \text{ where } \delta_{ij} = w_j \exp -W^2\left(S_i(x), T_j(x)\right) \text{ and } \tau_i = \sum_{j=1}^{M}\delta_{ij} \tag{9}$$

In the M-step the parameter set θ is updated using Equations 10, 11 and 12.

$$w_j = \frac{w_j}{\sum_{j=1}^{M}\sum_{i=1}^{N} w_j} \text{where } w_j = \sum_{i=1}^{N} w_i\gamma_{ij} \tag{10}$$

$$\mu_j = \sum_{i=1}^{N}\pi_i\mu_i \text{ where } \pi_i = \frac{w_i\gamma_{ij}}{\sum_i w_i\gamma_{ij}} \text{for fixed j} \tag{11}$$

$$\Sigma_j = \sum_{i=1}^{N}\pi_i\Sigma_i + \sum_{i=1}^{N}\pi_i(\mu_i - \mu_j)(\mu_i - \mu_j)^T \tag{12}$$

III. EVALUATION RESULTS

In this section first the dataset used is introduced, as well as the parametrisation of the algorithm and non-cancer cell classification procedure is described.

A. Dataset

In this work we use a FCM dataset of 116 patients with B-ALL acquired at treatment day 15. The patient's therapy was guided by the AIEOP-BFM 2009 protocol[1]. At the national diagnostic reference center all samples have been prepared and annotated according to the international standard operating procedure for 6 color FCM. For every cell ten FCM based features are measured (3 optical features (FSC-A, FSC-W, SSC-A), 7 fluorescence based features (CD20, CD10, CD45, CD34, SYTO41, CD19, CD38). For every patient in average 300 000 cells are measured. In FCM partial overlapping of fluorescence spectra of different fluorochromes is removed by spillover compensation. The preprocessing concludes with a normalization of the measured parameter values between 0 and 1.

B. Experimental Setup

The methodology proposed is evaluated in the following way: As comparable performance measures for non-cancer cell identification after data normalization, the mean sensitivity, accuracy, precision, accuracy and f-score over all patients in the test fold are computed. We used a 4 fold cross validation setup, where the test fold consisted of 29 patients and the training fold of 87. The measured cells in the training data were subsampled in a random way, by extracting 100 000 non blast cells per patient, resulting in a total of $8.7*10^6$ non blast cells. Subsequently we learned a GMM representation of the non-blast cells of the training set (Source) and test set (Target) separately, using the toolbox *mixture* integrated in the python framework *sklearn*[2]. The parameter set of the transformed Source GMM is estimated by using 1000 iterations of Expectation Maximization (EM) and the methodology introduced in Section II. Subsequently, the transformed model is used to predict a score for every cell of a test patient's sample to identify non-cancer cells.

C. Discussion

In Table I the non-blast classification performance is summarized using three different alignment metrics and 3 different setups of component numbers to represent the source GMM and target GMM. Non-blasts are classified based on WD best with a precision of 0.9534, accuracy of 0.9402 and f-score of 0.9585 in the experiment with a number of components (3,4) and (4,6) compared to the KLD and BB metric. This confirms the observations of Chen et al. [3], that an GMM interpolation scheme based on WD is efficient for high-dimensional GMM with a small number of components. For (4,5) number of components the KL divergence performs better compared to WD and BB with an accuracy of 0.9366 and f-score of 0.9582. The worst results are achieved by the BB metric with an average difference of -0.2 of the f-score and -0.3 difference in the sensitivity for N_{NB} = 3 and M = 4 Gaussian components.

IV. CONCLUSION

We propose an optimal transport scheme for GMM representation of FCM data by minimizing the Wasserstein

[1]AIEOP-BFM 2009 is a conducted randomized clinical trial for ALL between age 1-18 years in 10 countries in- and outside Europe, with approximately 1000 patients observed per year (Dworzak, 2013)) https://bfminternational.wordpress.com/ [accessed 2019-03-15]

[2]PythonToolbox:https://scikit-learn.org/stable/modules/generated/sklearn.mixture.GaussianMixture.html#sklearn.mixture.GaussianMixture [accessed 2019-03-15]

TABLE I

ILLUSTRATION OF THE NON-BLAST CLASSIFICATION PERFORMANCE USING GMM INTERPOLATION WITH WASSERSTEIN DISTANCE (WD), KULLBACK LEIBLER DIVERGENCE (KLD) AND BHATTACHARYYA BOUND (BB) FOR DIFFERENT THERAPY TIME POINTS, AND WITH DIFFERENT NUMBER OF COMPONENTS N_{NB} FOR THE SOURCE NON BLAST GMM, AND M FOR THE TARGET SAMPLE.

Experiment	N_{NB}	M	Sensitivity	Precision	Accuracy	f-score
KLD	3	4	**0.9780**	0.9488	0.9339	**0.9564**
WD	3	4	0.9732	**0.9505**	**0.9347**	0.9558
BB	3	4	0.6904	0.9441	0.6798	0.7188
KLD	4	5	**0.9821**	0.9489	**0.9366**	**0.9582**
WD	4	5	0.9702	**0.9503**	0.9316	0.9540
BB	4	5	0.8587	**0.9506**	0.8274	0.8703
KLD	4	6	**0.9814**	0.9495	0.9376	**0.9585**
WD	4	6	0.9755	**0.9534**	**0.9402**	**0.9585**
BB	4	6	0.8711	0.9499	0.8419	0.8835

distance between GMMs using Expectation Maximization. Results suggest that the WD performs best for non-blast identification and normalization compared to KLD and BB, for a low number of Gaussian mixture components. The focus of this work is to align stable populations (non cancer cells), which is a limit of the approach. For future work we plan to extend this approach for normalized non-cancer cell populations to perform anomaly detection for blast identification of multi-center data and of data from different treatment time points.

REFERENCES

[1] G. Basso, M. Veltroni, M. Valsecchi, M. Dworzak, R. Ratei, D. Silvestri, A. Benetello, B. Buldini, O. Maglia, G. Masera, *et al.*, "Risk of relapse of childhood acute lymphoblastic leukemia is predicted by flow cytometric measurement of residual disease on day 15 bone marrow," *Journal of Clinical Oncology*, vol. 27, no. 31, pp. 5168–5174, 2009.

[2] Bing J. and B. Vemuri, "Robust Point Set Registration Using Gaussian Mixture Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1633–1645, aug 2011.

[3] Y. Chen, T. T. Georgiou, and A. Tannenbaum, "Optimal transport for Gaussian mixture models," *arXiv preprint arXiv:1710.07876*, oct 2017.

[4] M. Dworzak, G. Fröschl, D. Printz, G. Mann, U. Pötschger, N. Mühlegger, G. Fritsch, and H. Gadner, "Prognostic significance and modalities of flow cytometric minimal residual disease detection in childhood acute lymphoblastic leukemia," *Blood*, vol. 99, no. 6, pp. 1952–1958, 2002.

[5] G. Gaipa, G. Cazzaniga, M. Valsecchi, R. Panzer-Grümayer, B. Buldini, D. Silvestri, L. Karawajew, O. Maglia, R. Ratei, A. Benetello, S. Sala, A. Schumich, A. Schrauder, T. Villa, M. Veltroni, W.-D. Ludwig, V. Conter, M. Schrappe, A. Biondi, M. Dworzak, and G. Basso, "Time point-dependent concordance of flow cytometry and real-time quantitative polymerase chain reaction for minimal residual disease detection in childhood acute lymphoblastic leukemia." *Haematologica*, vol. 97, no. 10, pp. 1582–93, October 2012.

[6] H. J. Kim, N. Adluru, M. Banerjee, B. C. Vemuri, and V. Singh, "Interpolation on the Manifold of K Component GMMs," in *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, dec 2015, pp. 2884–2892.

[7] R. Licandro, M. Reiter, M. Diem, M. Dworzak, A. Schumich, and M. Kampel, "Application of machine learning for automatic mrd assessment in paediatric acute myeloid leukaemia," in *Proceedings of the 7th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM,*, INSTICC. SciTePress, 2018, pp. 401–408.

[8] I. Naim, S. Datta, J. Rebhahn, J. Cavenaugh, T. Mosmann, and G. Sharma, "Swift - scalable clustering for automated identification of rare cell populations in large, high-dimensional flow cytometry datasets, part 1: Algorithm design," *Cytometry Part A*, vol. 85, no. 5, pp. 408–421, 2014.

[9] D. Y. Orlova, S. Meehan, D. Parks, W. A. Moore, C. Meehan, Q. Zhao, E. E. B. Ghosn, L. A. Herzenberg, and G. Walther, "QFMatch: multidimensional flow and mass cytometry samples alignment," *Scientific Reports*, vol. 8, no. 1, p. 3291, dec 2018.

[10] D. Y. Orlova, N. Zimmerman, S. Meehan, C. Meehan, J. Waters, E. E. B. Ghosn, A. Filatenkov, G. A. Kolyagin, Y. Gernez, S. Tsuda, W. Moore, R. B. Moss, L. A. Herzenberg, and G. Walther, "Earth movers distance (emd): A true metric for comparing biomarker expression levels in cell populations," *PLOS ONE*, vol. 11, no. 3, pp. 1–14, 03 2016.

[11] M. Reiter, P. Rota, F. Kleber, M. Diem, S. Groeneveld-Krentz, and M. Dworzak, "Clustering of cell populations in flow cytometry data using a combination of gaussian mixtures," *Pattern Recogn.*, vol. 60, no. C, pp. 1029–1040, Dec. 2016.

[12] P. Rota, M. Reiter, S. Groeneveld-Krentz, and M. Kampel, "The role of machine learning in medical data analysis. a case study: Flow cytometry," in *Proceedings of the Internaction Conference on Computer Vision Theory and Applications*, 2016.

Detecting Out-of-Distribution Traffic Signs

Madhav Iyengar[1], Michael Opitz[2] and Horst Bischof[2]

***Abstract*— This work addresses the problem of novel traffic sign detection, i.e. detecting new traffic sign classes during test-time, which were not seen by the classifier during training. This problem is especially relevant for the development of autonomous vehicles, as these vehicles operate in an open-ended environment. Due to which, the vehicle will always come across a traffic sign that it has never seen before. These new traffic signs need to be immediately identified so that they can be used later for re-training the vehicle. However, detecting these novel traffic signs becomes an extremely difficult task, as there is no mechanism to identify from the output of the classifier whether it has seen a given test sample before or not. To address this issue, we pose the novel traffic-sign detection problem as an out-of-distribution (OOD) detection problem. We apply several state-of-the-art OOD detection methods and novelty detection methods on the novel traffic-sign detection problem and also establish a benchmark using the German Traffic Sign Recognition Benchmark dataset(GTSRB). In our evaluation, we show that both out-of-distribution approaches and novelty detection approaches are suitable for OOD traffic sign detection.**

I. Introduction

The problem of detecting novel traffic signs is to detect whether a given traffic sign is from a class which was not seen by the classifier during training. This is crucial for autonomous vehicles, as they navigate within an open-ended environment and therefore, come across previously unseen traffic signs regularly. Consequently, these novel traffic signs have to be recognized, so that they can be labeled and added to the ever-increasing training data of the vehicle. However, the collection of this data is an extremely difficult task, since there is no way to ascertain from the output of the classifier whether a given test sample is similar to the training data or not. Even though modern neural networks manage to attain state-of-the-art performance in several complex tasks like image classification. [12], medical image diagnosis [1], speech recognition [10], natural language processing [19], etc., they are typically overconfident in their predictions. Several recent works substantiate this and show that neural networks give high predictions even on irrelevant, [11], [26], [20] and unrecognizable, [21] inputs. To overcome this problem of collecting a vast amount of traffic sign data, we propose to present the novel traffic sign detection problem as an out-of-distribution (OOD)/ novelty detection problem.

[1]This work was performed during an internship at the Institute of Computer Graphics and Vision, {thealmightylion.madhav}@gmail.com

[2]Institute of Computer Graphics and Vision, Graz University of Technology {michael.opitz,bischof}@icg.tugraz.at

The main objective of OOD detection is to detect whether a given test sample is from the in-distribution (i.e. same distribution as the data on which the network was trained on) or from the out-distribution (i.e. a distribution different from the in-distribution). This can also be interpreted as an additional binary classification task, where we want to predict true, if the given data is from the in-distribution and false, if it is not. However, it is crucial to add this binary classification task without affecting the performance of the original classifier.

A naïve solution to the OOD problem is to increase the size of the training data and explicitly add OOD examples to it. This enables us to teach the network to classify whether a test sample is in-distribution or OOD, by just using an additional label. However, collecting such a dataset is prohibitively expensive, as OOD samples, by definition, can be infinitely many. Furthermore, with the addition of these OOD samples more complex neural network architectures may need to be employed to correctly classify the training samples. This makes training of the network intractable and eventually makes this approach computationally expensive.

Thus, to solve this challenging real-world problem of novel traffic-sign detection, we adopt state-of-the-art OOD detection methods, i.e. [16], [14], and also apply non deep-learning based novelty detection methods like the One-Class Support Vector Machine (SVM) [23]. Further, we also evaluate a supervised linear SVM to this problem, in order to get an estimate of the upper bound accuracy of the methods. and linear SVM [3]. Except the supervised Linear SVM, none of these methods need OOD samples during training time. We illustrate how we use OOD detectors to ease the labeling task of traffic signs in Figure 1. To the best of our knowledge, we are the first to establish a benchmark on the detection of OOD traffic signs. We accomplish this task by using the German Traffic Sign Recognition Benchmark (GTSRB) [25] and also test the performance of our trained classifiers on a private dataset with extremely promising results. These methods manage to achieve a high AUROC score of 97.2% and a very low detection error of 5.7% on the challenging task of detecting novel classes in the GTSRB dataset.

II. Related Work

Out-of-distribution detection has received a lot of attention recently. Approaches can be mainly categorized into simple threshold based detectors, GAN based approaches and works that directly estimate confidence.

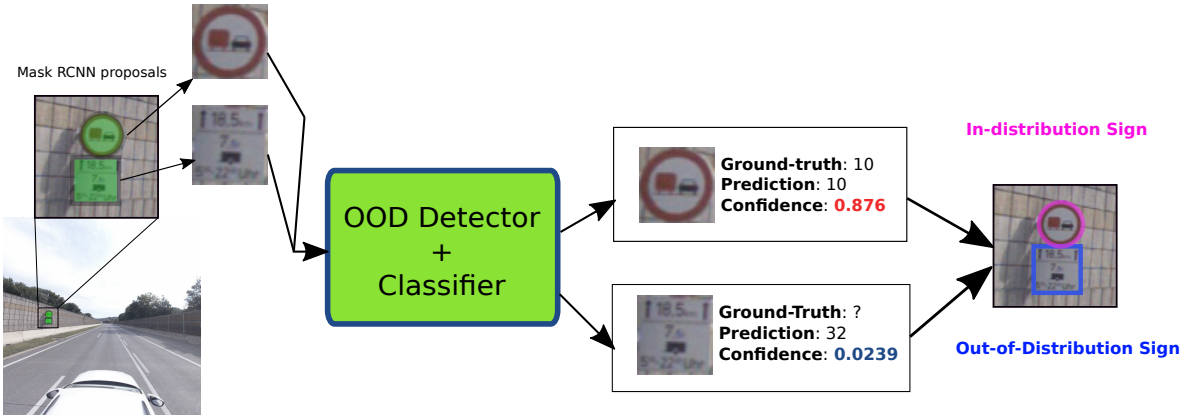


Fig. 1: Pipeline to ease labeling task of traffic-signs. The first step of our pipeline is to detect traffic-signs using Mask RCNN. The second step is to extract the traffic-sign crops from the predicted proposals. In the third step, our OOD detector provides a confidence score for each input sample. If the confidence score is greater than a selected threshold, i.e. in-distribution, the input sign does not require labeling. Whereas, if the confidence score is less than the threshold, i.e. OOD, the input sign should be marked for labeling.

Threshold Based Detectors : Hendrycks and Gimpel [11] propose a baseline method to detect OOD samples by using a simple threshold based detector mechanism, which requires no additional re-training of a network. Their method is based on the empirical observation that well-trained neural networks tend to assign higher softmax scores to samples which are from the in-distribution compared to samples from OOD. They use the predictions of a pre-trained classifier to compute a confidence score $c(x)$ on a test sample x. The detector then assigns the label 1 if the confidence score $c(x)$ is greater than some chosen threshold δ and 0 otherwise. This confidence score c is defined as the maximum value of the predictive distribution of the neural network.
Liang et al. [16] propose ODIN, which builds upon this confidence threshold based method and further enhances its performance by applying temperature scaling [9] and by adding small controlled perturbations [8] to the input data. Liang et al. [16] use these controlled perturbations to increase the softmax score assigned by the network on a given input. They show that adding these perturbations to the input combined with temperature scaling, helps in further enlarging the gap between the softmax scores assigned to in-distribution and OOD examples.
These techniques, although computationally cheap, heavily rely on the ability of the pre-trained classifier to separate the softmax scores on in-distribution and OOD samples.

Confidence Calibration using GAN : Lee et al. [14] propose a training mechanism which focuses on minimizing the Kullback-Leibler (KL) divergence loss between the softmax scores assigned by the network on OOD samples and the uniform distribution. Minimizing this loss forces the network to be uncertain (i.e. unable to assign a high softmax score) on examples which are not from the in-distribution. However, to minimize the KL divergence, OOD samples are required. To obtain the training data for this loss, the authors propose introducing a Generative Adversarial Network (GAN) [7] based loss which will be responsible to generate the OOD samples. However, unlike the original GAN, they modify the GAN loss so that it generates samples which are in the the low density region of the in-distribution. These generated samples are used as the OOD dataset to minimize the KL divergence. The model is then trained jointly with this KL divergence loss, GAN loss and the original classification loss.

Confidence Estimation : DeVries and Taylor [5] propose a training method which directly outputs the confidence of a network on a given sample. They achieve this by introducing an additional neuron from the last convolutional layer of the network which is solely responsible for confidence estimation. This neuron has a sigmoid activation function to keep the confidence scores between 0 and 1. They teach the network to output high confidence scores, by weighting the softmax values of the network with the confidence score. The confidence is then penalized by a log-loss, which forces the predicted confidence scores to be far from 0 for in-distribution (training) samples. However, this approach requires extremely high regularization to work as intended.

Some metric learning based approaches also try to perform the task of OOD detection. For example, Masana et al. [18] propose a metric learning based method which uses a contrastive loss on a Siamese network to learn feature embeddings. However, their method requires samples from an OOD dataset to train.

There are also approaches which use an ensemble of classifiers [13] and Bayesian probabilistic models [15], [17] to tackle the problem of OOD detection. However, these methods are computationally expensive and have higher inference times compared to the previously mentioned approaches.

III. Methodology

To detect novel traffic signs we use OOD methods. More specifically, we consider threshold based detectors *Section III − A* and GAN based approaches *Section III − B*. Further, we compare these approaches with novelty detection methods i.e One-Class SVM in *Section III − C*

A. Threshold Based Detectors

Threshold-based detector methods do not require any additional pre-training of the network and can be used out-of-the-box on any pre-trained classifier to detect OOD samples. This can be used as a baseline to compare the performance of detectors on the OOD detection task. They feed each input sample $\boldsymbol{x}$ into the neural network, and calculate its softmax score $S(\boldsymbol{x})$ is calculated. Next, they compare this score to a threshold δ. The input $\boldsymbol{x}$ is considered to be in-distribution if the softmax score is above the threshold and is considered OOD, otherwise. This simple OOD detector g is formulated as,

$$g(\boldsymbol{x};\delta) = \begin{cases} 1 & \text{if} \max S(\boldsymbol{x}) \leq \delta, \\ 0 & \text{if} \max S(\boldsymbol{x}) > \delta. \end{cases} \tag{1}$$

Liang et al. [16] further improve the ability of the OOD detector in Eq. (1) by introducing temperature scaling and a pre-processing technique based on adding perturbations to the input data.

During training time, they do not apply temperature scaling. During test time, the temperature scaling modifies the standard softmax function with a temperature scaling parameter $T \in \mathbb{R}^+$ such that,

$$S_i(\boldsymbol{x};T) = \frac{\exp\left(f_i(\boldsymbol{x})/T\right)}{\sum_{j=1}^{N} \exp\left(f_j(\boldsymbol{x})/T\right)}, \tag{2}$$

where $\mathbf{x}$ denotes the input sample, $\boldsymbol{f} = (f_1, ..., f_N)$ represents the logits, i.e. the output of the last layer of the neural network classifying N classes and S denotes the modified softmax function. The network predicts an output label y using this temperature scaled softmax function. The authors show that during test time, a favorable selection of the scaling parameter T can help push the softmax scores of the in and out-of distribution samples further apart from each other, thus making the OOD samples easier to differentiate.

Liang et al. [16] also propose a pre-processing technique, which involves adding small controlled perturbations to the input data as follows:

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} - \varepsilon \text{sign}(-\nabla_{\boldsymbol{x}} \log S_y(\boldsymbol{x};T)), \tag{3}$$

where ε is the perturbation magnitude. Adding controlled perturbations affects the in-distribution samples more than it does OOD samples, thus helping the classifier easily distinguish in-distribution from OOD samples. This enhanced detector is formulated similar to Eq. (1) as,

$$g(\boldsymbol{x};\delta,T,\varepsilon) = \begin{cases} 1 & \text{if} \max S(\tilde{\boldsymbol{x}};T) \leq \delta, \\ 0 & \text{if} \max S(\tilde{\boldsymbol{x}};T) > \delta. \end{cases}$$

B. Confidence Calibration using GAN

Lee et al. [14] propose a training technique to further improve the inherent property (as shown by Hendrycks and Gimpel [11]) of trained classifiers to assign higher softmax scores to in-distribution samples and lower softmax scores to OOD samples. They suggest that additionally minimizing the KL divergence loss between the softmax scores assigned by the classifier on OOD samples and the uniform distribution $\mathscr{U}(y)$, where y is the prediction of the network, should help the network learn to be less confident on OOD samples. To this end, they optimize

$$\begin{aligned} \min \; & E_{P_{\texttt{in}}(\widehat{\mathbf{x}},\widehat{y})}\left[-\log S(y=\widehat{y}|\widehat{\mathbf{x}}) \right] \\ & + \beta \cdot E_{P_{\texttt{out}}(\mathbf{x})}\left[KL(\mathscr{U}(y) \parallel S(y|\mathbf{x})) \right], \end{aligned} \tag{4}$$

where P_{in} represents the in-distribution, P_{out} represents the OOD and $\beta > 0$ is a penalty parameter. Thus, $P_{\texttt{out}}(\mathbf{x})$ refers to a sample taken from the out-distribution and $P_{\texttt{in}}(\widehat{\mathbf{x}},\widehat{y})$ refers to a sample and its corresponding ground truth taken from the in-distribution. The first term in the confidence loss corresponds to the standard label-based cross entropy loss used for the task of correctly classifying the categories of in-distribution samples. The second term of the confidence loss, i.e. the KL divergence term, forces the classifier to predict values closer to the uniform distribution for OOD samples. Therefore, the KL divergence term forces the classifier to be uncertain, i.e. it is unable to predict a high softmax value on OOD samples. Consequently, the classifier not only learns to perform well on its original classification task, but is also able to distinguish whether a given sample is from the OOD. However, to minimize the KL divergence term in Eq. (4), the authors need explicit samples from the OOD, to which they do not have access to during training. To tackle this problem the authors use a GAN to generate samples from the out-distribution. However, as the priori knowledge for OOD samples is not available, Lee et al. [14] propose to modify the original GAN loss so that it generates samples in the low-density region of the in-distribution. The original GAN loss [7] is formulated as,

$$\begin{aligned} \min_G \max_D \quad & E_{P_{\texttt{in}}(\mathbf{x})}\left[\log D(\mathbf{x}) \right] \\ & + E_{P_{\texttt{pri}}(\mathbf{z})}\left[\log\left(1 - D(G(\mathbf{z}))\right) \right], \end{aligned} \tag{5}$$

where G is the Generator, D is the Discriminator and $P_{\texttt{pri}}(\mathbf{z})$ represents a latent variable $\mathbf{z}$ sampled from a prior distribution which is used by the generator. Optimizing this min-max objective forces $P_G \approx P_{\texttt{in}}$, i.e. forces the generator to generate samples from the in-distribution.

However, as the objective is to generate samples from P_{out}, the authors add an additional KL divergence loss term to Eq. (5) similar to the one in Eq. (4). This modified loss is

formulated as,

$$\min_G \max_D \beta \cdot \underbrace{E_{P_G(\mathbf{x})}\big[KL(\mathscr{U}(y) \parallel S(y|\mathbf{x}))\big]}_{\text{(a)}} + \underbrace{E_{P_{\text{in}}(\mathbf{x})}\big[\log D(\mathbf{x})\big] + E_{P_G(\mathbf{x})}\big[\log(1-D(\mathbf{x}))\big]}_{\text{(b)}}, \quad (6)$$

The first term of this loss is similar to the KL divergence loss term in Eq. (4). However, the generator distribution P_G replaces the P_{out} in this KL loss i.e.,the KL loss minimizes the samples generated by the generator, instead of needing explicit OOD samples. The second term corresponds to the original GAN loss in Eq. (5) which tries to generate in-distribution samples. However, term (a) in Eq. (6) forces the generator to generate samples further away from the in-distribution, as in-distribution samples would increase the KL divergence loss. On the contrary, if the generator generates samples which are too far away from the in-distribution boundary, the term (b) will be very high, i.e. the GAN loss forces the generator to create samples which are not too far away from the in-distribution boundary. The authors combine these 2 equations, i.e. Eq. (6) + Eq. (4), to jointly train the GAN and the classifier. Thus the KL divergence loss term in both equations not only encourages the GAN to generate samples in the low-density area of the in-distribution, but also forces the classifier to be uncertain on OOD samples. This joint confidence loss is formulated as follows:

$$\min_G \max_D \min \underbrace{E_{P_{\text{in}}(\widehat{\mathbf{x}},\widehat{y})}\big[-\log S(y=\widehat{y}|\widehat{\mathbf{x}})\big]}_{\text{(c)}} + \beta \cdot \underbrace{E_{P_G(\mathbf{x})}\big[KL(\mathscr{U}(y) \parallel S(y|\mathbf{x}))\big]}_{\text{(d)}} + \underbrace{E_{P_{\text{in}}(\widehat{\mathbf{x}})}\big[\log D(\widehat{\mathbf{x}})\big] + E_{P_G(\mathbf{x})}\big[\log(1-D(\mathbf{x}))\big]}_{\text{(e)}}, \quad (7)$$

where (c) + (d) correspond to the classification loss and (d) + (e) corresponds to the GAN loss. Thus, this joint confidence loss minimizes the KL divergence on low-density in-distribution samples, effectively helping the classifier to create a tighter bound on the in-distribution, without affecting the performance of the original classification task.

C. One-Class SVM

One-class SVMs [23] try to separate all data points from the origin with a hyperplane in a projected feature space. Further, they maximize the distance from the hyperplane to the origin. To this end they solve a quadratic optimization problem i.e.,

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\sum_{ij}\alpha_i\alpha_j k(\mathbf{x}_i,\mathbf{x}_j) \text{ subject to } 0 \le \alpha_i \le \frac{1}{\nu\ell}, \sum_i \alpha_i = 1 \quad (8)$$

where x_i denotes the hidden layer features of our CNN, $k(\cdot,\cdot)$ is an RBF kernel function, $\alpha_{i,j}$ are support vectors, ℓ is the number of training samples and ν is a hyperparameter.

IV. Experiments

In this section we compare the performance of OOD detectors on the task of detecting OOD traffic signs. In Section IV-A we give an overview of our in-distribution and OOD datasets. In Section IV-B we give a detailed overview of the pre-processing and data-augmentation techniques applied by us on the in-distribution dataset. In Section IV-C we explain our training setup and provide choices of the different hyperparameters that give us the best results. In Section IV-D we summarize the results of the state-of-the-art methods used by us on the task of novel traffic-sign detection.

For evaluation, we use the metrics proposed by Liang et al. [16]. Specifically, we compare the confidence scores from both in-distribution and OOD samples in the test set. From these scores, we compare the area under the ROC curve (AUROC), the FPR at 95% TPR and minimal detection error metrics. Further, we compute the area under the precision-recall curve, where we consider the in-distribution as positive class (AUPR In) and the OOD as the positive class AUPR Out.

A. Datasets

In the following, we describe the in-distribution and OOD datasets that we use for all our experiments.

1) In-Distribution Dataset: We choose the first 35 of 43 classes, of the GTSRB as our in-distribution. This dataset originally consists of 30,621 training images and 11,070 test images, each of size 32×32. After applying our extensive pre-processing and data-augmentation techniques, we extend the first 35 classes of the GTSRB (GTSRB-35) dataset to 612,420 images (see in Figure 2).

2) Out-of-Distribution Datasets: At test time, we consider the test-images from our training dataset (GTSRB-35) as in-distribution samples. For OOD samples we test our detectors with several image datasets, as listed below.

(1) The **GTSRB-last8** consists of the last 8 classes of the GTSRB, i.e the classes of GTSRB which were not used during the training phase. This dataset comprises of 4,440 images, each of size 32×32. We show a test sample from each of the 8 classes in Figure 3.
(2) The **Private Dataset** consists of 3 classes, i.e. Bike and Pedestrians sign, No Stopping Zone sign and Unknown

Fig. 2: Samples of the 35 classes used for training our network.

Fig. 3: Samples from the last 8 classes of GTSRB, which are used by us for evaluation.

traffic-signs. The Unknown sign class consists of several traffic signs which are region specific and not too crucial for everyday usage. We obtain this dataset from a drive in an urban setting. This private dataset comprises of 1,293 images, each image has been resized to a size of 32×32. We also show some sample images from each of the 3 classes in Figure 4.

B. Pre-Processing

We apply some specialized pre-processing and data augmentation techniques to increase the size of the input data from GTSRB to improve robustness of our trained classifiers. We cannot use standard augmentation techniques such as random flipping and rotation as they can change the meaning of the sign.
To overcome this problem, we use different augmentations depending on the class. We always apply small geometric distortions (e.g., rotation, projective transforms). We flip signs horizontally or vertically only if they are symmetric (i.e., the class label does not change), or if the transformation yields an image with a different class label. For the latter images, we change the corresponding label after this transformation.(e.g., turn left - turn right)

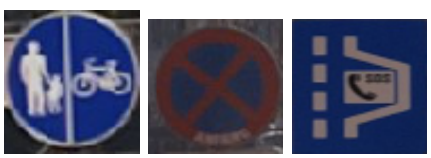


Fig. 4: Samples from our private dataset, which are used for evaluation. The first sign is from the Bike & Pedestrians class, the second from the No Stopping Zone class and the third a random sign from the Unknown class.

C. Training Setup

We perform all our experiments using a simple VGG-13 network [24]. We adopt the same CNN architecture and hyperparameters for our VGG-13 network as Lee et al. [14]. Our network is able to achieve a classification test accuracy of 98% on the first 35 classes of the GTSRB dataset (GTSRB-35).

D. Results

In this section, we show the results of our baseline approaches in Section IV-D.1, threshold based approaches (ODIN) in Section IV-D.2 and GAN-based approaches in Section IV-D.3. We compare the results of all our experiments with that of the Baseline detector [11], so as to get a qualitative understanding of the performance of each approach. We also compare the simple softmax based thresholding detector with novelty detection methods like the One-Class SVM [23] and also with a supervised method i.e. the linear SVM, so as to get an upper bound on the performance of the OOD methods.

1) Baseline Methods: In this section, we explain our three baseline approaches, i.e. simple softmax thresholding, one-class SVM and supervised linear SVM. For these approaches we first train a standard CNN classifier. The simple thresholding approach detects OOD images based on the maximum softmax scores.
For the one-class SVM we use an RBF kernel and sample 10,000 images from the training set. We extract 512 dimensional features from the last hidden layer of our CNN, which we use for training. For the one-class SVM we apply a grid search on the parameters nu and gamma, and find that setting ν to 0.0001 and γ to 3.2 and 2.3 works best for GTSRB-last8 and our private dataset, respectively.
For the supervised linear SVM baseline, we further use

In distribution dataset	Out-of-distribution dataset	Method	FPR (95% TPR) ↓	Detection Error ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
GTSRB-35	GTSRB-last8	**Baseline [11]**	100.0	37.4	26.5	57.3	94.7
		ODIN	28.1	10.4	95.1	96.2	92.6
		GAN-based approach	24.1	10.5	91.7	92.1	94.2
		One-class SVM	**12.1**	**5.7**	**97.2**	**98.1**	**95.1**
		Linear SVM*	0.1	1.7	99.7	99.7	99.5
	Private Dataset	**Baseline [11]**	100.0	8.7	83.7	85.9	98.9
		ODIN	1.0	2.7	99.4	99.5	99.2
		GAN-based approach	**0.7**	**2.5**	**99.5**	**99.6**	**99.4**
		One-class SVM	1.8	2.7	98.8	98.0	98.8
		Linear SVM*	0.8	1.7	99.3	99.0	99.5

TABLE I: Distinguishing in- and OOD test set data for traffic-sign classification. All values are percentages. ↑ indicates larger value is better, and ↓ indicates lower value is better.
* denotes supervised learning

samples from the train split of the GTSRB-last8 and our private dataset (which are not used during evaluation) for training. We also perform a grid search on the penalty parameter C and set it to 1 and 10 for GTSRB-last8 and our private dataset, respectively.

2) Threshold based Methods: In this section, we summarize the results of the ODIN detector proposed by Liang et al. [16] in Table I. We empirically set the temperature parameter to 1000 and the perturbation magnitude to 0.0034 for both our target datasets.

Fig. 5: Images generated by the GAN in the low-density region of the in-distribution

3) Confidence Calibration using GAN: In this section we summarize the results of the GAN based OOD method proposed by Lee et al. [14] in Table I. We choose the penalty parameter for the KL loss, i.e. β to be empirically 1.3.
We also show some of the traffic-sign images generated by the GAN to verify that the generated samples are indeed in the low-density region of the in-distribution (GTSRB-35) in Figure 5. Interestingly, as we observe from the first and second images generated by the GAN, the generated samples are hard to recognize even by humans. For example, the first image could be either interpreted as 50km/h or 60km/h sign. Similarly, the second image could be interpreted as either No Vehicles (Circular sign with red boundary) or Yield (Triangular Sign with red boundary). Thus, this suggests that the generator does indeed generate samples which look similar to the in-distribution. However, as these samples are extremely difficult to recognize even by humans, we can conclude that the generated samples are indeed from the low-density region of the in-distribution.

To summarize our results, we find that the baseline one-class SVM yields the best results on the GTSRB-last8 dataset and the GAN-based approach performs best on our private dataset. Interestingly, we find that the supervised linear SVM serves as a reasonable estimate of the upper-bound on the task of OOD traffic sign detection.

V. Conclusion

In this paper, we proposed a benchmark for the task of novel traffic sign detection by using current state-of-the-art OOD detection and novelty detection methods. We showed that the one-class SVM performs best on the challenging task of detecting the GTSRB-last8 dataset and the GAN-based approach performs best on detecting our private dataset as OOD. For future work, we intend to use this benchmark to help autonomous vehicles learn new traffic sign classes incrementally. We also plan to experiment with metric learning based approaches and a variety of divergence losses to further improve performance of OOD detectors.

VI. Acknowledgements

This work was supported by the Austrian Research Promotion Agency (FFG) under the project *DGT - Dynamic Ground Truth (860820)*. We also thank *Joanneum Research Digital* for providing us with anonymized imagery, which made our dataset evaluations possible.

References

[1] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission," in *Proceedings of ACM SIGKDD*. ACM, 2015.

[2] H. S. Christopher D Manning, "Foundations of statistical natural language processing," *MIT Press*, vol. 999, 1999.

[3] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, 1995.

[4] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves." in *ICML*, 2006.

[5] T. DeVries and G. W. Taylor, "Learning confidence for out-of-distribution detection in neural networks," *arXiv preprint arXiv:1802.04865*, 2018.

[6] T. Fawcett, "An introduction to roc analysis," *Pattern recognition letters*, 2006.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS*, 2014.

[8] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples. corr (2015)."

[9] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," *arXiv preprint arXiv:1706.04599*, 2017.

[10] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[11] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in *ICLR*, 2017.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.

[13] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *NIPS*, 2017.

[14] K. Lee, H. Lee, K. Lee, and J. Shin, "Training confidence-calibrated classifiers for detecting out-of-distribution samples," *arXiv preprint arXiv:1711.09325*, 2017.

[15] Y. Li and Y. Gal, "Dropout inference in Bayesian neural networks with alpha-divergences," in *ICML*, 2017.

[16] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," *ICLR*, 2018.

[17] C. Louizos and M. Welling, "Multiplicative normalizing flows for variational Bayesian neural networks," in *ICML*, 2017.

[18] M. Masana, I. Ruiz, J. Serrat, J. van de Weijer, and A. M. Lopez, "Metric learning for novelty and anomaly detection," *BMVC*, 2018.

[19] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[20] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[21] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[22] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," *PloS one*, vol. 10, no. 3, 2015.

[23] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *NIPS*, 2000.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[25] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The German Traffic Sign Recognition Benchmark: A multi-class classification competition," in *IEEE International Joint Conference on Neural Networks*, 2011.

[26] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *International Conference on Learning Representations*, 2014. [Online]. Available: http://arxiv.org/abs/1312.6199

The Quest for the Golden Activation Function*

Mina Basirat[1], Alexandra Jammer[2], and Peter M. Roth[1]

I. PROBLEM STATEMENT AND MOTIVATION

Deep Neural Networks have been shown to be beneficial for a variety of tasks, in particular allowing for end-to-end learning and reducing the requirement for manual design decisions. However, still many parameters have to be chosen in advance, also raising the need to optimize them. Moreover, since increasingly more complex and deeper networks are of interest, strategies are required to make neural network training efficient and stable. While initialization and normalization techniques are well studied, a relevant and important factor is often neglected: the selection of a proper activation function (AF). In [1], we tackled this problem and learned task-specific activation functions. For that purpose, we take two main observations into account. First, the positive and negative parts of activation functions have a different influence on information propagation. Second, the search space is very huge and hard to explore. Thus, motivated by evolution theory (e.g., [3], [4]) we introduced an approach to evolving piece-wise activation functions building on the ideas of Genetic Programming (e.g., [2]).

II. OVERVIEW OF THE APPROACH

The evolution typically starts from a population consisting of randomly selected candidate solutions, called *individuals*. These are described by a set of properties (*genes*: functions in our case), which can be altered by three breeding operations: (a) *Selection*, (b) *Crossover*, and (c) *Mutation*. In addition, we introduced two new operators, especially representing our problem: *Inheritance Crossover* and *Hybrid Crossover*. The first additionally allows for combining different positive and negative parts, whereas the second one introduces the possibility to combine functions using mathematical operators. Then, in an iterative process, where we refer to one iteration as a *generation*, each individual is evaluated and based on their fitness, we select a set of parent solutions for breeding. Subsequently, we apply breeding operators on pairs of individuals to generate new pairs of offsprings. This process is repeated until a pre-defined number of generations or a pre-defined optimality criterion is met. To this end, we are able to evolve even more complex activation functions. This is in particular remarkable as only very basic candidate solutions are provided (in contrast to, e.g., `Swish`). Moreover, our approach is adapting very well to different kinds of problems, also yielding different activation functions for different tasks.

*We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

[1]Graz University of Technology, {mina.basirat, pmroth}@icg.tugraz.at
[2]University of Graz, alexandra.jammer@uni-graz.at

III. ILLUSTRATIVE EXPERIMENTAL RESULTS

To demonstrate the benefits of our approach, we run experiments on two different classification benchmarks of different complexity, namely CIFAR-10 and CIFAR-100, and compared it to existing approaches. Illustrative results for ResNet-20 are shown in Table I (In addition, we carried out experiments for ResNet-56 and VGG-16.).

TABLE I: Results for CIFAR-10 using ResNet-20.

Activation Function	Accuracy
Ours (best)	79.24%
`Swish`	78.51%
`ELU`	73.00%
`ReLU`	71.98%
`SeLU`	65.79%
Random Search (best)	76.03%

In addition, Figs. 1 and 2 show the best-performing activation functions for CIFAR-10 and CIFAR-100, respectively. It cannot only be seen that for the different tasks different activation functions have been evolved, but also that the shapes of the top-performing functions are similar. For more details, we would like to refer to [1].

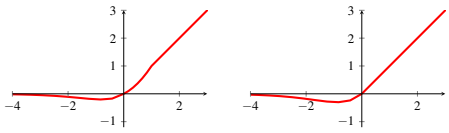


Fig. 1: Top 2 evolved AFs for CIFAR-10.

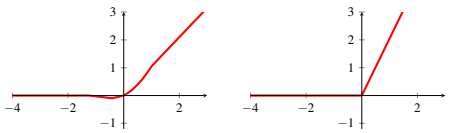


Fig. 2: Top 2 evolved AFs for CIFAR-100.

REFERENCES

[1] M. Basirat and P. M. Roth. Learning task-specific activation functions using genetic programming. In *Proc. Int'l Conf. on Computer Vision Theory and Applications*, 2019.
[2] K. A. De Jong. *Evolutionary Computation: A Unified Approach*. MIT Press, 2006.
[3] D. J. Futuyma and M. Kirkpatrick. *Evolution*. Oxford University Press, 4th edition, 2017.
[4] J. B. Losos, D. A. Baum, D. J. Futuyma, H. E. Hoekstra, R. E. Lenski, A. J. Moore, C. L. Peichel, D. Schluter, and M. J. Whitlock, editors. *The Princeton Guide to Evolution*. Princeton University Press, 2014.

Combining Deep Learning and Variational Level Sets for Segmentation of Buildings

Muntaha Sakeena[1] and Matthias Zeppelzauer[2]

***Abstract*— The larger context behind this work is the automated visual assessment of building characteristics (e.g. building age and condition) for the estimation of real estate prices from outdoor pictures. A basic requirement to this end is the automated segmentation of buildings from photos, which is the focus of this work. We propose a combined deep-learned and variational segmentation method for the extraction of the building area from real estate images and our experimental results with dice similarity of approx. 92% demonstrate its capabilities on a novel dataset for building segmentation.**

I. INTRODUCTION

Deep learning-based approaches proved to be effective in many computer vision applications including the image-based appraisal of real estate [6]. The estimation of certain building characteristics, however, requires a different perspective, i.e. the street view perspective. The segmentation of buildings in street view perspective has hardly been performed and is more complex than aerial image segmentation because perspective and appearance varies much stronger. Furthermore, occluding foreground objects, such as trees and plants impede segmentation.

For the robust assessment of building parameters there is a need for a precise segmentation of buildings to remove unnecessary and potentially misleading information (see Fig. 1). Today, segmentation is used for the tasks like autonomous driving [1] and urban scene understanding [3]. In this work, we present a robust method for the segmentation of unconstrained building views to lay the foundations for extended real estate image analysis (REIA) which is an emerging and challenging computer vision problem [4], [7]. To mitigate shortcomings of existing segmentation networks, we propose a combination with Variational Level sets (VLS) [5] to improve the segmentation quality. We evaluate the approach on a set of pixel-wise annotated building images from real estate websites and show (i) that transfer learning (fine tuning) is essential for obtaining satisfactory results and (ii) that VLS improve the boundary, which seems to be difficult to learn for the network, especially when training data is limited.

II. APPROACH

A major challenge is that buildings can be captured from different perspectives and with different scales. To capture this variety, we propose a two-step approach, illustrated in Fig. 2. In the first step, we employ SegNet [1], which is a

Fig. 1. The task of building segmentation

powerful and flexible encoder-decoder network for semantic segmentation to initially detect the building in the image. Each encoder stores the feature indices extracted from each layer and uses them at the decoder side. The SegNet input is usually an RGB image with a pixel-wise label image as ground truth. Due to the pixel-wise nature of SegNet and in the presence of a limited amount of training data, there is a risk of obtaining noisy, irregular or over-smooth boundaries for the segmented objects as well as outliers (small false positive areas). Therefore, we add a second processing step to refine the initial output mask. To get more exact and accurate boundaries, we integrate Variational Level Sets (VLS) into the approach which are well-suited for the detection of smooth and regular boundaries around objects having different shapes and topology [5].

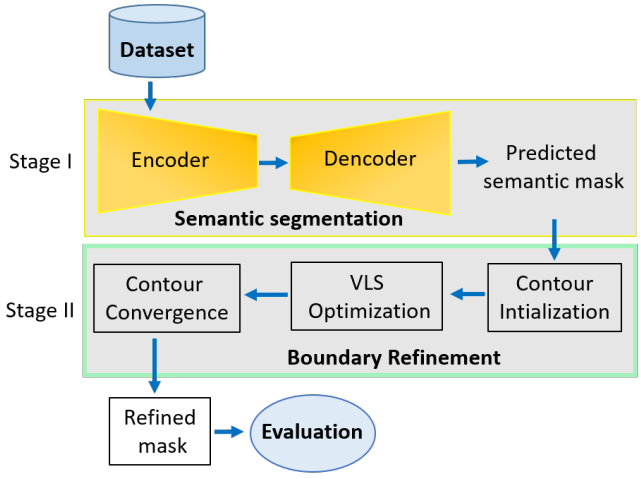


Fig. 2. Proposed overall approach

The idea of VLS is based on the minimization of energy E which integrates the geometric properties of the level set with image features. As input, we consider an image I composed of two regions: "building" and "no-building", which are

*This work was supported by FFG projects 855784, 856333 and 865973
Sankt Pölten University of Applied Sciences, Austria
[1] muntaha.sakeena@fhstp.ac.at
[2] matthias.zeppelzauer@fhstp.ac.at

mutually exclusive and separated by a contour ϕ. Gradient descent is then applied for optimizing and converging ϕ around the boundaries. Contour convergence VLS further includes a regularization term $R(\phi) = \frac{1}{2}\int_I (\nabla\phi - 1)^2 dI$ to limit the contour to a signed distance function. The curve evolves until separate regions can be defined by minimizing E as $(\partial\phi)/(\partial t) = -(\partial E)/(\partial\phi)$ which represents the gradient flow that minimizes the energy E.

III. Experiments & Results

Dataset: We collected the dataset from real estate websites to train and evaluate our approach. The dataset is composed of 975 images in which each building was manually labeled at the pixel level. The dataset is divided into three subsets: training (50%), validation (20%), and testing (30%).

Performance Measures: We employ the Dice similarity coefficient (DSC) to assess the performance of our approach which is specifically designed to evaluate segmentation tasks: $DSC = 2(|S \cap G|)/(|S| + |G|)$, where S is the segmented area in the result image and G is the ground truth mask.

Experimental setup: Initially, we employ SegNet pre-trained on the CamVid dataset [2] which contains a building class (aside from other classes like road and car) to evaluate its generalization ability. In a next step, we fine-tune SegNet for 50 epochs for domain adaption. Input images were resized proportionally to the size of the input layer (360×480). We trained SegNet with a learning rate of $1e^{-3}$, a momentum of 0.9 and a batch size of 4. For VLS, we directly used the output of SegNet to initialize ϕ. This reduces the computation time for contour initialization significantly and mitigates the instabilities resulting from the sensitivity of VLS to initial contour selection. We keep the contour evolving for 110 iterations to assure convergence.

Results: We evaluate our approach on the independent test set. For a fair comparison, we evaluate (i) SegNet pre-trained on the CamVid [2] dataset, (ii) SegNet fine-tuned on our training set and (iii) the latter net with boundary refinement by VLS, see Table I for the results. Our experiments show that the pre-trained SegNet generalized poorly to buildings of different types and from different perspectives than seen before (in the CamVid dataset), see column 2 of Fig. 3. Fine-tuning the network strongly boosts performance from 60% to approx. 89%, showing that domain adaption is of high importance. Still, we observe that the output contours are noisy. By applying VLS refinement to the output of the fine-tuned network, the boundaries further improve which is also reflected in the segmentation results in Table I and column 3 of Fig. 3. Further results are depicted in Fig. 4.

TABLE I

Achieved segmentation performance

Methods	DSC %
Random Baseline	51.17
Pre-trained SegNet	60.23
Fine-tuned SegNet	89.27
Boundary refinement by VLS	**91.70**

Fig. 3. Test images with predicted contours/masks: Original image (first column), Pre-trained SegNet (second column), predicted mask by proposed method with boundary refinement (third column)

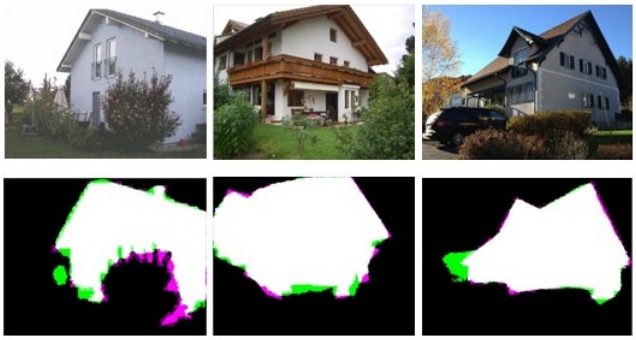


Fig. 4. Test results of the proposed method: original image (first row), predicted masks (second row): true positives (white), false positives (green), true negatives (black), false negatives (pink).

IV. Conclusions

We have presented an approach for the segmentation of buildings by combining a semantic segmentation network with VLS. Results are promising and sufficiently accurate for future visual extraction of higher-level building parameters for real estate appraisal. In future, we plan to compare our method with Conditional random field (CRF) methods.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *CoRR*, vol. abs/1511.00561, 2015.

[2] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *ECCV (1)*, 2008, pp. 44–57.

[3] C.-A. Brust, S. Sickert, M. Simon, E. Rodner, and J. Denzler, "Efficient convolutional patch networks for scene understanding," in *CVPR Workshop on Scene Understanding (CVPR-WS)*, 2015.

[4] D. Koch, M. Despotovic, M. Sakeena, M. Döller, and M. Zeppelzauer, "Visual estimation of building condition with patch-level convnets," in *Proc. ACM Wsp. on Multimedia for Real Estate Tech*, 2018, pp. 12–17.

[5] C. Li, C. Xu, C. Gui, and M. D. Fox, "Distance regularized level set evolution and its application to image segmentation," *Image Processing, IEEE Transactions on*, vol. 19, no. 12, pp. 3243–3254, 2010.

[6] Q. You, R. Pang, and J. Luo, "Image based appraisal of real estate properties," *CoRR*, vol. abs/1611.09180, 2016.

[7] M. Zeppelzauer, M. Despotovic, M. Sakeena, D. Koch, and M. Döller, "Automatic prediction of building age from photographs," in *Proc. of ACM ICMR*, 2018, pp. 126–134.

Automatic Intrinsics and Extrinsics Projector Calibration with Embedded Light Sensors

Thomas Pönitz[1], Christoph Heindl[1], Andreas Pichler[1], Martin Kampel[2]

***Abstract*— We propose a novel projector calibration method based on embedded light sensors. Our method can be used to determine intrinsics and extrinsics of one or multiple projectors without relying on an additional camera. We show that our method is highly accurate and more than 17 times faster than state of the art methods. This renders our method suitable for spatial augmented reality applications in the industrial domain.**

I. INTRODUCTION

Augmented reality has become an important topic for industrial assembly. Workers are provided with localized and contextualized information to support them during difficult tasks. However in industrial environments special conditions prevail. Heat, dirt, gloves etc. impede the use of touch-screens. Wearables become cumbersome during long working hours and battery life is a constant concern.

As a solution to these problems spatial augmented reality (SAR) has gained traction. In SAR there are no monitors, head-mounted displays or hand-held devices. Instead physical objects become the canvas themselves. Information is either projected directly on the object of interest or objects next to it, e.g. a table surface, are used as screens. SAR does not obstruct workers in any kind Obstruction can be solved with multiple projectors targeting the same workstation.

In industrial settings high spatial precision is of special importance for SAR setups, therefore it is critical to achieve a highly accurate calibration of the SAR system's projectors.

II. STATE OF THE ART

To describe the intrinsics of a projector the same parameters as for cameras are used – focal lengths f_x, f_y, principal point (c_x, c_y) and radial and tangential distortion parameters k_i, p_j [1]. Determined by their intended use projectors exhibit very low distortion and it is typically sufficient to consider only k_1 and k_2.

In [3] a printed chessboard is observed by an uncalibrated camera. The projector projects a graycode based pattern as a series of images. The camera captures each of these projections. From these captured images it is possible to generate a homography mapping from camera pixel coordinates to projector pixel coordinates. Because both the camera and the projector exhibit distortions, a local homography is calculated around each chessboard corner. The camera pixel coordinates of all chessboard corners are determined with subpixel accuracy and mapped into projector pixel coordinates by the local homographies. At this stage the projector can be calibrated with standard methods for camera calibration, e.g. [4]. The drawback with this method is that an additional device, the camera, is needed. Projector and camera need to be synchronized which limits acquisition speed. The chessboard corner detection introduces additional complexity, processing time and error sources. The resolution of the pattern is limited by the camera as it has to resolve the fine pattern details.

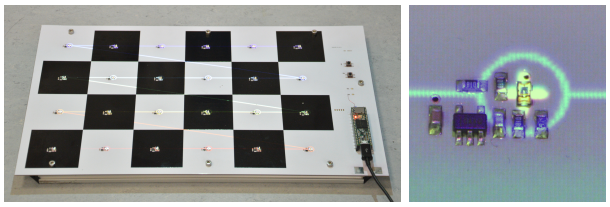


Fig. 1: Active calibration object with projected detections and detailed picture of the embedded light sensor.

[5] proposes a virtual pattern consisting of white and black circles. The black circles are printed as a physical pattern. The white circles are rendered in 3d using the projector's extrinsics and intrinsics and projected. A calibrated camera depicts the projected white circles and printed black circles onto an undistorted image. The centers of all circles are detected with subpixel accuracy.

The extrinsic and intrinsic parameters of the projector are iteratively adapted until black circle centers and white circle centers match exactly the virtual pattern. Similar drawbacks as described above apply. Additionally the camera needs to be calibrated beforehand and a starting solution has to be provided.

III. METHOD

To overcome the disadvantages of state of the art methods described in section II we propose a method without an additional camera.

Our method is based on [2], where embedded light sensors are placed on the corners of rectangular surfaces. We use sensors that have a radiant sensitive area of 0.23 square millimeter to observe a projected graycode pattern as in [5]. From the observed pattern the projector pixel coordinates can be determined and a homography is calculated. This allows to project onto the rectangle and let it act as a display. Furthermore a three dimensional object is equipped with sensors and the projector to object pose estimated, which enables projecting a texture onto the object.

However no intrinsics model is calculated, which is necessary for industrial high precision SAR, because even small distortions interfere with high-precision projections

[1]Profactor GmbH, 4407 Steyr-Gleink, Austria
[2]TU Wien, 1040 Vienna, Austria

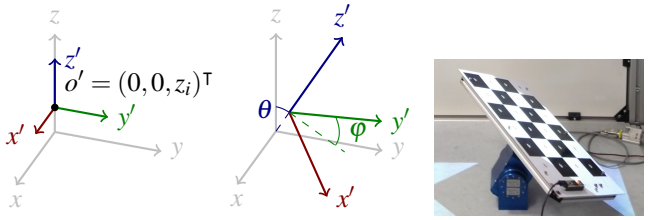
and the intrinsics of Ultra Short Throw (UST) projectors can not simply be inferred from the projectors field of view specification. Therefore we propose the **active calibration object**, a three dimensional object equipped with embedded light sensors at well defined positions. Figure 1 depicts an early flat, rectangular prototype.

The hardware is capable of capturing 60 frames per second, as is currently the standard for projectors. For WUXGA resolution (1920 × 1200 pixel) the theoretical number of patterns is $\lceil \log_2(1920) \rceil + \lceil \log_2(1200) \rceil = 22$. To increase robustness against external lighting and resolve cases where pixel edges fall directly onto the sensors, we project each image followed by its inversion. Additionally we add two images at start and end which allows us to detect the beginning and ending of the pattern solely from the observed data. This results in a pattern length of $2+22\cdot 2+2=48$ and a total acquisition time of $\frac{48}{60}=0.8$ seconds, limited currently only by the frames per second of the projector.

In comparison [5] would need 46 pattern images for the same resolution (a white and black frame to gauge brightness levels instead of the two start and stop pattern images). Considering the time needed for the camera to acquire the image and projector latency, an optimistic estimate is 0.3 seconds per image, resulting in 13.8 seconds, which is $\frac{13.8}{0.8} = 17.25$ times slower.

IV. EXPERIMENTS

To evaluate our method we used two setups. First, our calibration object is mounted on a linear axis, with movement orthogonal to the object's plane and second, mounted on a wrist joint with two rotational axis, see figure 2. In the



(a) (b) (c)

Fig. 2: Experimental setups: (a) variable height z_i and (b) wrist joint with two radial axis (θ rotates around x, φ around z'), illustrated in (c).

first setup we move the calibration object to 20 positions, equally spaced 1 cm apart, and acquire measurements, see figure 2a. We calibrate with these measurements stacked together, resulting in 12 degrees of freedom (6 extrinsics, 6 intrinsics f_x, f_y, c_x, c_y, k_1, k_2). Table I compares results of 15 calibrations from distinct measurements. In the second setup we rotate the calibration object around two axis, see figure 2b. θ assumes values $-35°,-30°,\ldots,35°$ and φ values $-135°,-130°,\ldots,45°$ – resulting in 555 positions total. Utilizing the projector intrinsics obtained from the prior setup we solve the PNP problem for each position resulting in calibration object poses and therefore a position $\mathbf{s}_{ijk} \in \mathbb{R}^3$ for every embedded light sensor (i$^{\text{th}}$ sensor, j$^{\text{th}}$ angle θ and k$^{\text{th}}$ angle φ). For each pose we compare the observed φ against the reported angle of the wrist joint, see table II.

TABLE I: Mean and standard deviation over 15 calibrations of the root mean square reprojection-error (RMSE), focal lengths f_x, f_y, principal point $(c_x, c_y)^\intercal$ and radial distortion coefficients k_1, k_2. ($\alpha^\beta \triangleq \alpha \cdot 10^\beta$)

	RMSE	f_x	f_y	c_x	c_y	k_1	k_2
μ	7.6^{-1}	2,965.9	2,971.6	1,008	−76.7	-4.4^{-2}	6.6^{-2}
σ	1.9^{-2}	2.7	3.1	1.4	4	8.9^{-3}	2.7^{-2}

Additionally for each angle $\theta_{j^\star}$ we fit a plane $P_{j^\star}$ through all sensor positions $\mathbf{s}_{ij^\star k}$. The distance of sensor position $\mathbf{s}_{ij^\star k}$ to plane $P_{j^\star}$ is $z_{ij^\star k}$. Sensor positions are projected onto their respective plane. When φ is varied, all plane points rotate around the same center point and all points belonging to the same sensor lie on the same circle. We obtain the center point and all circles, respectively their radii, by minimizing the square of the circle-point distances. The deviation of a point from its circle is $\mathbf{c}_{ij^\star k} = (x_{ij^\star k}, y_{ij^\star k})^\intercal$. We now form error vectors $\mathbf{e}_{ij^\star k} = (x_{ij^\star k}, y_{ij^\star k}, z_{ij^\star k})^\intercal$ and transform them into the projector coordinate system and compare them in table II as root mean square errors (RMSE) over angles θ_i.

Summarizing, we showed the high accuracy of our proposed system and its practicability for calibrating projectors in industrial SAR applications.

TABLE II: Root mean square error (RMSE) of angle φ and sensor positions $(x,y,z)^\intercal$ in projector coordinates, over angles θ. ($\alpha^\beta \triangleq \alpha \cdot 10^\beta$)

θ [°]	RMSE_φ [°]	RMSE_x [mm]	RMSE_y [mm]	RMSE_z [mm]
−35	2.8^{-2}	2.4^{-1}	1^{-1}	3.2^{-1}
−30	2.4^{-2}	2.2^{-1}	9.4^{-2}	3.4^{-1}
−25	2.4^{-2}	1.9^{-1}	6.6^{-2}	3.4^{-1}
−20	2.8^{-2}	1.5^{-1}	5.4^{-2}	3.7^{-1}
−15	2.3^{-2}	1.2^{-1}	5.6^{-2}	3.9^{-1}
−10	2^{-2}	6.2^{-2}	3.1^{-2}	3.1^{-1}
−5	1.9^{-2}	3.3^{-2}	2.1^{-2}	2.8^{-1}
0	1.7^{-2}	2.5^{-2}	2.3^{-2}	3.7^{-1}
5	1.9^{-2}	4.2^{-2}	3.2^{-2}	3.6^{-1}
10	2.3^{-2}	6.7^{-2}	5^{-2}	3.2^{-1}
15	2.3^{-2}	1.1^{-1}	6.6^{-2}	3.3^{-1}
20	2.3^{-2}	1.4^{-1}	7.5^{-2}	3.1^{-1}
25	2.5^{-2}	2.2^{-1}	1.2^{-1}	3.9^{-1}
30	3^{-2}	2^{-1}	1^{-1}	3.1^{-1}
35	2.7^{-2}	2.7^{-1}	1.5^{-1}	3.5^{-1}

REFERENCES

[1] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2003.

[2] J. C. Lee, P. H. Dietz, D. Maynes-Aminzade, R. Raskar, and S. E. Hudson, "Automatic projector calibration with embedded light sensors," in *Proceedings of the Symposium on User Interface Software and Technology*. ACM, 2004, pp. 123–126.

[3] D. Moreno and G. Taubin, "Simple, accurate, and robust projector-camera calibration," in *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*. IEEE, 2012, pp. 464–471.

[4] R. Y. Tsai, "An efficient and accurate camera calibration technique for 3d machine vision," 01 1986.

[5] L. Yang, J.-M. Normand, and G. Moreau, "Practical and precise projector-camera calibration," in *International Symposium on Mixed and Augmented Reality*. IEEE, 2016, pp. 63–70.

Acknowledgments The project leading to this application has received funding from the Fuel Cells and Hydrogen 2 Joint Undertaking under grant agreement No 735367. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and Hydrogen Europe and N.ERGHY.

The Coarse-to-Fine Contour-based Multimodal Image Registration

Maliheh Assadpour Tehrani[1] and Robert Sablatnig[1]

***Abstract*— Image registration brings two images into alignment despite any initial misalignment. Several approaches to image registration make extensive use of local image information extracted in interest points, known as local image descriptors. State-of-the-art methods perform a statistical analysis of the gradient information around the interest points. However, one of the challenges in image registration by using these local image descriptors arises for multimodal images taken from different imaging devices and/or modalities. In many applications such as medical image registration, the relation between the gray values of multimodal images is complex and a functional dependency is generally missing. This paper focuses on registering Mass spectrometry images to microscopic images based on contour features. To achieve more accurate multimodal image registration performance, we proposed a coarse-to-fine image registration framework. The pre-registration process is performed by using contour-based corners and curvature similarity between corners. Image blocking and DEPAC descriptors are used in the fine registration process. A local adaptive matching is performed for the final registration step.**

I. INTRODUCTION

The term image registration describe the procedure of aligning two images that have been acquired by different imaging conditions. The overall aim of image registration approaches is to determine a transformation between the source and target images. There have been multiple approaches during the last decades in order to find such a transformatation[19][17]. According to[19] the process of image registration involves the following steps: Feature detection and matching, transform model estimation and image transformation. Two or more images are called multimodal if different sensors or different imaging devices obtain them. They can provide different information of the same scene but the multimodal registration task is difficult since the images obtained from different modalities can have extreme intensity mapping dissimilarity.

A. Mass Spectrometry Imaging

Mass Spectrometry Imaging (MSI) technologies are powerful tools to investigate the molecular information from biological tissue samples and visualize their complex spatial distributions [4]. MSI allows untargeted analysis of hundreds of molecular species directly from a tissue sample, providing direct spatial correlation between their abundances and histological features. Beside mass-to-charge (m/z) values and their respective intensities, MSI is also recording the spatial position where signals are desorbed and ionized from surface [14].

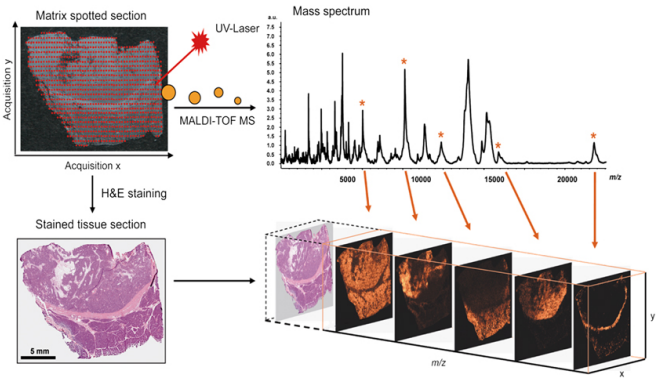


Fig. 1. An example of MSI and Histology Imaging.

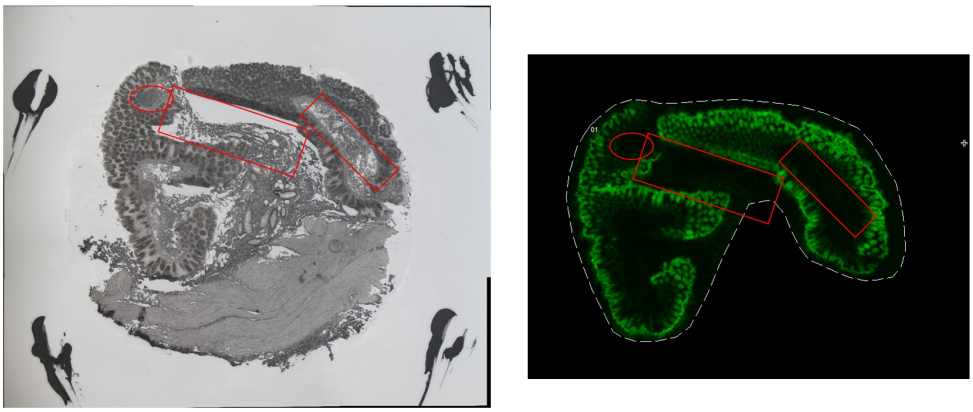
A schematic representation of a MSI is shown in Fig. 1. The sample (tissue section) is covered with matrix in order to allow desorption and ionization of analytes. The resulting ions are transferred to the mass spectrometer, and a mass spectrum is acquired. Then, the next position by a defined distance is analysing. MSI of a selected analyte peak are generated after the measurement by extracting the signal intensity within a certain m/z window. The intensities are plotted as gray scale values for each pixel in a grid representing the corresponding positions on the sample.

B. Challenging multimodal images

MSI can generate biomolecular profiles that describe the spatial distribution of specific biomolecules including metabolites, lipids, peptids and proteins. However, it can not provide histoanatomical and molecular depth information [4]. Therefore, the combination of information from different modalities has proven to be a powerful approach for obtaining molecular signatures from specific cells/tissues of interest. For example, MSI is combined with imaging modalities that have high spatial resolution (Optical Microscopy Imaging-OMI) or tissue structural information (Histology Images). An example of multimodal images is given in Fig.2, an OMI and a MSI. Obviously, there are very large content differences between these two images. It can be seen that each of the regions in the microscopy images contains a large amount of information. In contrast, there is much less information in each of corresponding regions in the MSI. Moreover, Medical microscopic images contain many objects which are visually very similar. These issues are affecting the accuracy of registration results.

*This work was not supported by any organization

[1]Maliheh Assadpour Tehrani and Robert Sablatnig are with Institute of Visual Computing and Human-Centered Technology, Faculty of Informatics, TU Wien, Austria {tehrani,sab}@cvl.tuwien.ac.at



(a)Optical Microscopy (b)Mass Spectrometry

Fig. 2. An example OMI(a) and MSI(b). The selected regions by red lines show very large content difference between images.

C. Image Registration

The term image registration describes the procedure of aligning two images that have been acquired by different imaging conditions. The overall aim of image registration approaches is to determine a transformation between the source and target images. There have been multiple approaches during the last decades in order to find such a transformatation [17][19]. According to [19] the process of image registration involves the following steps: Feature detection and matching, transform model estimation and image transformation.

The extraction of effective features is a crucial step for the application of image registration. Over the last decades, numerous hand-engineered features such as SIFT[12] and Gabor filters[11] are used in the image registration procedures. However, they suffer from limitations for the multimodal image registration because the statistics on the local intensity distribution are insufficient to describe the complex relationship between modalities with different underlying imaging physics[3].

Two or more images are called multimodal if they are obtained by different sensors or different imaging devices. They can provide different information of the same scene but the multimodal registration task is difficult since the images obtained from different modalities can have extreme intensity mapping dissimilarity. Variation in intensities has two possible consequences as 'Gradient Reversal' and 'Region Reversal' which are described in the following part.

The gradient direction of corresponding parts in the multimodal images are changing by exactly 180°[15]. This is called 'Gradient Reversal' which is one of the main reasons that causes SIFT to fail with multimodal images. It also may draw on rotation normalization of regions fault. In a consequence of gradient reversal, the direction of the dominant orientation will reverse which may cause two similar regions remain totally out of phase. We call this property as 'Region Reversal'.

Among local features methods, multimodal variants of SIFT are particularly popular in multimodal image registration, including SIFT-GM (GM: Gradient Mirroring)[10], Symmetric SIFT [7], IS-SIFT (IS:Improved Symmetric) [9], GO-IS-SIFT [16], PIIFD (Partial Intensity Invariant Feature Descriptor) [6]. These variants of SIFT only consider 'Gradient Reversal' and 'Region Reversal' problem of multimodal images, however the real situation may be more complex, such as registering the two images shown in Fig. 2.

In [13], a multimodal image registration method based on a contour-based corner technique (COREG) which is independent of intensity and gradient changes is proposed to overcome these challenging images. COREG algorithm has shown satisfactory registration performance when registering images without large scale difference. Although, when the scale difference increase, the COREG performance is decreasing.

D. Contribution of the Paper

The problem that we try to overcome in this work is caused by large content and scale differences between MSI and OMI. First, the multimodal image registration technique based on a countour-based corner technique that is independent of intensity and gradient changes is investigated [13]. Then, we proposed a method by taking the advantage of COREG algorithm and adding Image blocking and local adaptive matching to the registration procedure for multimodal medical image registration.

The rest of this paper is organized as follow. In Section II, we present our proposed method. Coarse registration is described in II-A, followed by fine registration steps consist of image blocking in II-B, DEPAC descriptor extraction in II-C and local adaptive matching in II-D. The dataset and results are presented in Section III and this paper is concluded in Section IV.

II. THE PROPOSED TECHNIQUE

In order to register these challenging multimodal images, a fully automatic image registration approach is proposed in this paper. Our proposed method is designed based on

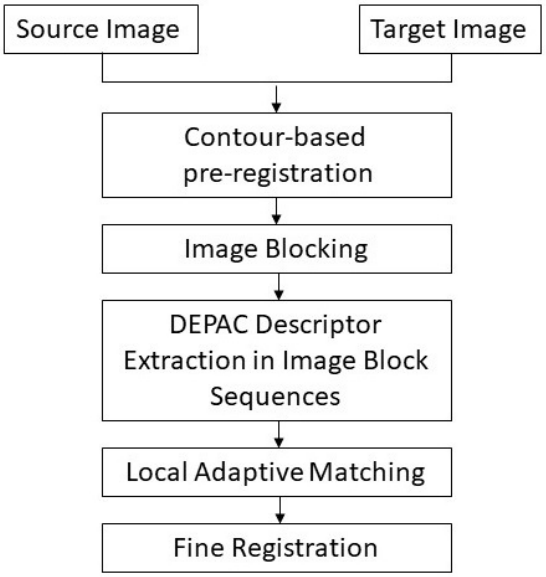


Fig. 3. The proposed algorithm flowchart.

registration process in COREG using the following processes as shown in Fig.3.

In this work, the image registration process is implemented in two main steps. First, the pre-registration process is initialized by detecting corners using Fast-CPDA and then the matching of corners are obtained according to the curvature similarities of all the corners. For each pair of corner triplets, the transformation is computed. By using this transformation, the target edge image is transformed to the reference edge image and the Number of Overlapped Pixels (NOP) is estimated. The transformation related to the maximum NOP is selected for coarse registration.

Second, the algorithm performs uniform image blocking of reference image and transformed target image obtained from pre-registration step. In this step, the DEPAC descriptors for all the corners are computed in each image blocks of reference and transformed target image. The second round matching process is carried out by local adaptive matching strategy.

A. Pre-registration

In this paper, the fast contour-based multi-scale corner detector based on the chord-to-point distance accumulation (Fast-CPDA) technique by [8] has proposed to detect the corners. This technique has proved its robustness over many other single- and multi-scale detectors [8][2][1].The CPDA detector [8] first extracts planar curves from the edge image detected by the Canny edge detector [5].

After corner detection, the curvature similarity of all detected corners in the source and the target image are determined as following.

We have two sets of corners in the reference and target images as

$$C_r = \left\{C_r^1, C_r^2, \ldots C_r^{N_r}\right\}, \tag{1}$$

and

$$C_t = \left\{C_t^1, C_t^2, \ldots C_t^{N_t}\right\}, \tag{2}$$

where N_r and N_t denote the number of corners in the reference and target images respectively. The curvatures of these corners are:

$$K_r = \left\{K_r^1, K_r^2, \ldots K_r^{N_r}\right\}, \tag{3}$$

and

$$K_t = \left\{K_t^1, K_t^2, \ldots K_t^{N_t}\right\}. \tag{4}$$

The curvature similarity of two corners is determined as

$$s^{ij} = \frac{\left|K_r^i - K_t^j\right|}{K_r^i}, \tag{5}$$

where $1 \le i \le N_r$ and $1 \le j \le N_t$.

Then all the target corners are ranked based on their curvature similarities to the each source corner. With these matches, the corner triplets are generated. After matching corner triplet pairs, the transformation is computed for each matching pair. This transformation is used to transform the target edge image to the reference edge image and the Number of Overlapped Pixels (NOP) is estimated. The transformation obtained from the pair of corner triplets with the maximum NOP is selected as a coarse transformation.

B. Image Blocking

The large content and scale difference between MSI and OMI are affecting the accuracy of the registration. COREG algorithm has shown satisfactory registration performance when registering images without large scale difference. Although, when the scale difference increase, the COREG performance is decreasing. Moreover, the space of geometric transformations becomes larger which means costing more time to compare corner triplets. Therefore, we employed the image blocking method which dividing each source and transformed target image (warp image) into the image blocks. After pre-registration step is performed, it is assumed that the source image size is $M \times N$ and the warp image size is $P \times Q$. We are dividing them into the $m \times n$ blocks that means the size of source and warp image blocks are $(M/m) \times (N/n)$ and $(P/m) \times (Q/n)$ accordingly. All these blocks are labeled and the similar blocks in the two images will have the same number. Hence, this approach decomposes a matching problem of a whole image into numerous matching problems of image blocks. For each block, the DEPAC method is used to build a DEPAC descriptor for each corner. The DEPAC descriptor is presented in Section II-C.

C. DEPAC Descriptor Extraction

G.Lv. et.al [13] presented the novel corner descriptor which is based on the curvature of a corner in order to capture important edge information in the neighborhood of a corner. The proposed corner descriptor is called Distribution of Edge Pixels Along Contour (DEPAC). The corners and their contours are presented as C_r^i, C_t^j, $\Gamma\left(C_r^i\right)$ and $\Gamma\left(C_t^j\right)$ in

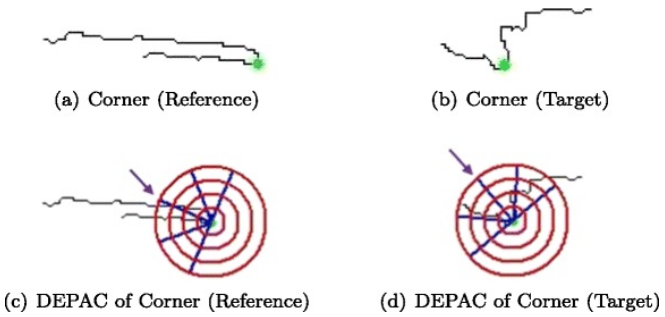


Fig. 4. DEPAC corner descriptor [13].

Fig.4 a and b. The DEPAC corner descriptor is built using C_r^i and $\Gamma\left(C_r^i\right)$. The main steps are as follows.

i. Each corner is used as a center to compute concentric circles are plotted, as shown in Fig. 4. Where R is denoted the radius of the internal circle. The radius of a concentric circle is incremented by R, from inside to outside.

ii. The average orientations of two tangants is considered as the main orientation of the corner, O_m. Arrows in Fig. 4 c,d, show the main orientation.

iii. At each side of main orientation, the orientation bins are defined. Therefore, the four quantized orientations are estimated as $O_1 = O_m - 90^\circ$, $O_2 = O_m - 45^\circ$,$O_3 = O_m$ and $O_4 = O_m + 45^\circ$ in an anticlockwise direction. Finally, we have 16 sub-regions which are defined in the neighborhood of the corner and each sub-region is denoted as (c,o), where $1 \le c \le 4$ and $1 \le o \le 4$.

iv. The number of edge pixels is increased if an edge pixel, P_e, along the contour falls into this sub-region, i.e.

$$(c-1) \times R < d\left(P_e, C_r^i\right) \le c \times R, \tag{6}$$

and

$$O_o \le \overrightarrow{C_r^i P_e} < O_{o+1}, \tag{7}$$

where $d\left(P_e, C_r^i\right)$ is the Euclidean distance between P_e and C_r^i.

v. Final step is the normalizing the number of edge pixels in each sub-region, NEPc,o, into [0,1] by

$$NEP_{c,o} = \frac{NEP_{c,o}}{max\{NEP_{c,o}\}}. \tag{8}$$

D. Local Matching Scheme

After DEPAC descriptor extraction is performed in each image sub-block, we are assuming that $D_{A_i} = \{f_j\}_i$ and $D_{B_i} = \{g_j\}_i$ are a set of DEPAC descriptors where A_i and B_i are image sub-blocks with number i in the source and the target image respectively. Number of extracted DEPAC descriptors in image sub-blocks is shown by j. The Euclidean distance is calculated between each DEPAC descriptor in image sub-block A_i and each DEPAC descriptor in image sub-block B_i.

By ranking the DEPAC descriptor distances, the nearest and the next nearest distance are selected to calculate the ratio distance as following:

$$R_j = d_{j1}/d_{j2}. \tag{9}$$

Where R_j is the ratio distance. The nearest and next nearest distance are denoted by d_{j1} and d_{j2}, respectively.

For each image sub-block, the ratio distance calculation is performed. All R_j are ranked from small to big. The matching corners to the smallest R are selected and merged for fine-registration step. Similar to the final step in coarse-registration, the matching of corner triplets is carried out based on selected corners. These corner triplets are used to estimate transformation which correspond to the higher NOP. Fig. 5 shows the fine-registration process consist of image blocking, DEPAC descriptor extraction and local adaptive matching.

III. EXPERIMENTS

In this section we present comparative study of presented methods for our specific multimodal images to investigate how they perform on registering MSI to Optical microscopy image. Then, we will evaluate our proposed method.

A. Test Data

Our dataset consists of 10 reference microscopy images from tissues and for each reference image, 4 Mass spectrometry images are available, leading to a total of 40 image pairs. The ground truth for all of our test images are processed using FlexAnalysis (version 3.0, Bruker Daltonics). By this software, the MS images are manually registered to the optical images.

B. Evaluation Metric

In order to quantitatively comparing the performance of the presented registration methods, we used the Average Registration Error(ARE) [18]. After aligning the reference and target images with the estimated transformation, ARE is used to measure the overlap error as defined by

$$ARE = \frac{1}{H \times W} \sum_{x=1}^{W} \sum_{y=1}^{H} \|T_e(x,y) - T_g(x,y)\|, \tag{10}$$

where H and W are the height and width of the reference image, T_g is the ground-truth transformation and T_e is the estimated transformation. The smaller the ARE value is, the better the registration performance will be.

C. Performance Comparisons

Fig. 6 compares our presented method with COREG in terms of ARE when registering image pairs of MSI and OMI. From the experimental results, it can be summarized that our proposed algorithm shows clear improvement over COREG. The scale difference between two images in image pairs 24 to 40 (Image Pair ID 5-8) is approximately 1X:10X. ARE values obtained by COREG registration process for these image pairs are obviously bigger compared to pairs 1 to 24. On the other hand, ARE values achieved by our method remain relatively stable in comparison with COREG when increasing scale differences.

In this paper we have proposed coarse-to-fine registration method to overcome the problem of big scale difference

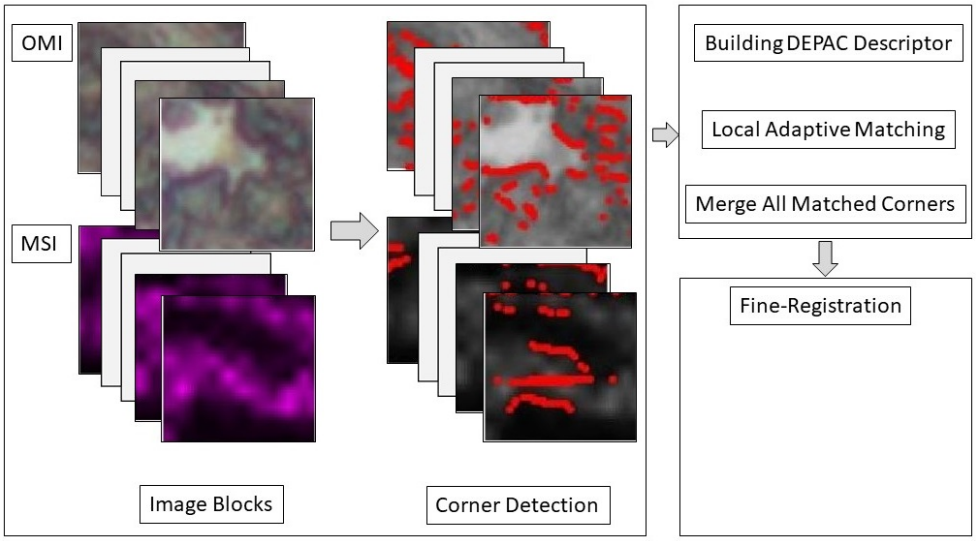


Fig. 5. Fine-registration Process.

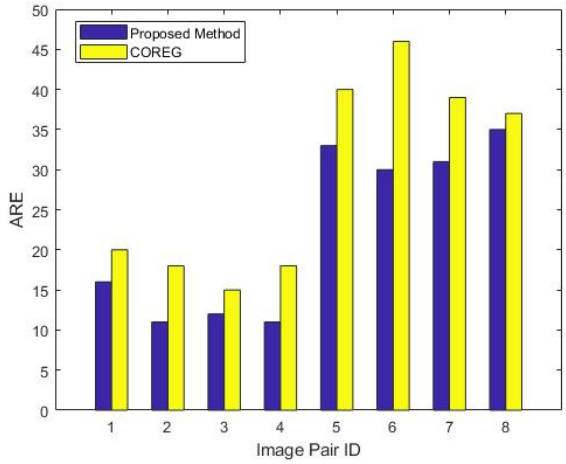


Fig. 6. ARE comparisons between our proposed method and COREG. In this figure image pairs 1-3 have scale difference of 1X:1X and image pairs 4-5 have scale difference of 1X:5X. Image pairs 6-8 have a large scale difference of 10X.

when registering MSI and OMI. The impact of scale difference in terms of ARE is presented in Table I. This proposed method has shown better results with average ARE of 80.45 in comparison of COREG with average ARE of 115.65.

TABLE I
IMPACT OF SCALE DIFFERENCE IN TERMS OF ARE.

Scale Difference	Proposed Method	COREG
$1X : 1X$	13	43
$1X : 5X$	27.5	38
$1X : 10X$	72.67	97.33

IV. CONCLUSION

We have presented a new multimodal image registration method based on contours and DEPAC descriptor. In order to address the large content and large scale difference in MSI and OMI, we have proposed coarse-to-fine registration framework. Our coarse registration is contour-based procedure. The proposed approach utilizes various techniques including the contour matching algorithm, curvature similarity, DEPAC descriptors, image blocking and local adaptive matching as a robust matching mechanism. The algorithm can complete fully automatic registration without any manual intervention. At the same time, all the matching corners extracted in the matching process are distributed uniformly, which can consequently improve the accuracy of registration.

ACKNOWLEDGMENT

Data used in preparation of this paper were obtained from the Institute of Chemical Technologies and Analytics, Vienna University of Technology, Vienna, Austria.

REFERENCES

[1] M. Awrangjeb and G. Lu, "Robust image corner detection based on the chord-to-point distance accumulation technique," *IEEE Transactions on Multimedia*, vol. 10, no. 6, pp. 1059–1072, oct 2008. [Online]. Available: http://ieeexplore.ieee.org/document/4657455/

[2] M. Awrangjeb, G. Lu, C. S. Fraser, and M. Ravanbakhsh, "A fast corner detector based on the chord-to-point distance accumulation technique," in *DICTA 2009 - Digital Image Computing: Techniques and Applications*. IEEE, 2009, pp. 519–525. [Online]. Available: http://ieeexplore.ieee.org/document/5384897/

[3] W. Bingjian, L. Quan, L. Yapeng, L. Fan, B. Liping, L. Gang, and L. Rui, "Image registration method for multimodal images," *Appl. Opt.*, vol. 50, no. 13, pp. 1861–1867, may 2011. [Online]. Available: http://ao.osa.org/abstract.cfm?URI=ao-50-13-1861

[4] A. R. Buchberger, K. DeLaney, J. Johnson, and L. Li, "Mass Spectrometry Imaging: A Review of Emerging Advancements and Future Insights," *Analytical Chemistry*, vol. 90, no. 1, pp. 240–265, jan 2018. [Online]. Available: http://pubs.acs.org/doi/10.1021/acs.analchem.7b04733

[5] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, nov 1986. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4767851

[6] J. Chen, J. Tian, N. Lee, J. Zheng, R. T. Smith, and A. F. Laine, "A partial intensity invariant feature descriptor for multimodal retinal image registration," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 7, pp. 1707–1718, July 2010.

[7] J. Chen and J. Tian, "Real-time multi-modal rigid registration based on a novel symmetric-SIFT descriptor," *Progress in Natural Science*, vol. 19, no. 5, pp. 643–651, may 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1002007109000240

[8] J. H. Han and T. Poston, "Chord-to-point distance accumulation and planar curvature: A new approach to discrete curvature," *Pattern Recognition Letters*, no. 10, pp. 1133–1144, aug.

[9] M. T. Hossain, G. Lv, S. W. Teng, G. Lu, and M. Lackmann, "Improved symmetric-sift for multi-modal image registration," in *2011 International Conference on Digital Image Computing: Techniques and Applications*, Dec 2011, pp. 197–202.

[10] A. Kelman, M. Sofka, and C. V. Stewart, "Keypoint descriptors for matching across multiple image modalities and non-linear intensity variations," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–7.

[11] R. Kolkila and T. P, "Image Registration Based on Fast Fourier Transform Using Gabor Filter," *International Journal of Computer Science and Electronics Engineering (IJCSEE).*, vol. 2, pp. 31–37., 2014.

[12] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, nov 2004. [Online]. Available: http://link.springer.com/10.1023/B:VISI.0000029664.99615.94

[13] G. Lv, S. W. Teng, and G. Lu, "COREG: a corner based registration technique for multimodal images," *Multimedia Tools and Applications*, vol. 77, no. 10, pp. 12 607–12 634, 2018.

[14] L. A. McDonnell and R. M. Heeren, "Imaging mass spectrometry," *Mass Spectrometry Reviews*, vol. 26, no. 4, pp. 606–643, jul 2007. [Online]. Available: http://doi.wiley.com/10.1002/mas.20124

[15] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever, "Image Registration by Maximization of Combined Mutual Information and Gradient Information," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2000*, S. L. Delp, A. M. DiGoia, and B. Jaramaz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 452–461.

[16] G. L. Shyh Wei Teng, Md. Tanvir Hossain, "Multimodal image registration technique based on improved local feature descriptors," *Journal of Electronic Imaging*, vol. 24, no. 1, pp. 1 – 17 – 17, 2015. [Online]. Available: https://doi.org/10.1117/1.JEI.24.1.013013

[17] A. Sotiras, C. Davatzikos, and N. Paragios, "Deformable medical image registration: A survey," *IEEE Transactions on Medical Imaging*, vol. 32, no. 7, pp. 1153–1190, jul 2013. [Online]. Available: http://ieeexplore.ieee.org/document/6522524/

[18] M. Xia and B. Liu, "Image registration by "super-curves"," *IEEE Transactions on Image Processing*, vol. 13, no. 5, pp. 720–732, may 2004. [Online]. Available: http://ieeexplore.ieee.org/document/1288197/

[19] B. Zitová and J. Flusser, "Image registration methods: a survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977–1000, oct 2003. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0262885603001379

Evaluation Study on Semantic Object Labelling in Street Scenes

Andreas Wittmann[1], Margrit Gelautz[2], Florian Seitner[3]

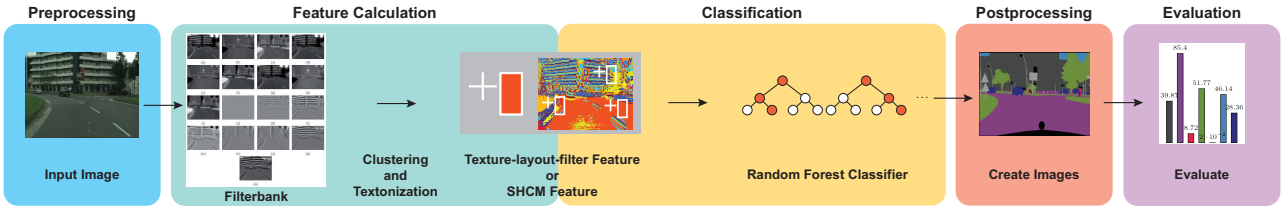


Fig. 1: Overview of our semantic segmentation system.

***Abstract*— We present a processing pipeline for semantic scene labelling that was developed in view of autonomous driving applications. Our study focuses on two different methods for feature selection - Texture-layout-filter (TLF) and Single Histogram Class Models (SHCM) - whose influence on the performance of a random forest classifier is investigated. In tests on the Cityscapes dataset, we assess the effects of parameter variation and observe an improvement of the Intersection over Union score by 44 percent when substituting the TLF by the computationally more demanding SHCM feature.**

I. INTRODUCTION

Advanced driving assistance systems and fully autonomous vehicles require detailed knowledge about vehicle surrounding and real-time interpretation of street scenes. Deep learning approaches are computationally demanding and need powerful and expensive hardware. On the other hand, conventional machine learning methods can be used for real-time semantic scene labelling at a fraction of the computational complexity and on cost-efficient hardware. In view of a subsequent transfer to an embedded platform, we implement a semantic scene labelling pipeline based on a random forest classifier and evaluate its performance regarding the overlap of the predicted semantic segmentation results and the ground-truth labelling by using the Intersection over Union score.

Thoma [8] gives a comprehensive survey of different techniques and features used for pixel-wise semantic segmentation. Several semantic scene labelling approaches rely on contextural features [9], [1], [4], [2], [7]. We employ two features that make use of textural and contextural properties proposed in the literature [7], [6] and analyze and compare their results on the publicly available Cityscapes dataset [3].

*This work was supported by the project CarVisionLight (Project No. 861251), which is funded by the Austrian Research Promotion Agency (FFG) under the programme "ICT of the Future", an initiative of the Federal Ministry of Transport, Innovation and Technology (BMVIT).

[1]Andreas Wittmann is with emotion3D, 1040 Vienna, Austria awi@emotion3d.ai

[2]Margrit Gelautz is with the Institute of Visual Computing and Human-Centered Technology, TU Vienna, 1040 Vienna, Austria margrit.gelautz@tuwien.ac.at

[3]Florian Seitner is with emotion3D, 1040 Vienna, Austria fse@emotion3d.ai

II. SEMANTIC SCENE LABELLING PIPELINE

Figure 1 visualizes the semantic segmentation pipeline realized by our system. Our segmentation pipeline consists of five modules. The *preprocessing module* serves to prepare incoming images and datasets. The *feature module* prepares the extraction of the features by providing a feature factory that can create variations of the feature within the predefined parameter ranges. The *classification module* instantiates features by using the *feature module* and calculates the feature response in real-time. It uses the feature responses in a random forest classifier. The *postprocessing module* generates prediction label images from the output of the random forest. Finally, the *evaluation module* calculates the performance of our algorithm in terms of the per-class prediction accuracy.

The feature module contains the computation of two features: the Texture-layout-filter (TLF) feature and the Single Histogram Class Models (SHCM) feature. The TLF feature combines textural information represented by textons with semantic, spatial and scale context by using information of neighbouring pixels and regions. The computation of the TLF feature is explained in detail by Shotton et al. [7]. The SHCM feature is based on the semantic segmentation approach of Schroff et al. [6]. The authors compare a histogram of a selected region with precalculated SHCM [5] in a random forest classifier. The SHCM feature can be seen as an extension of the TLF feature. While the TLF feature counts the occurrences of one single texton id within a region, the SHCM feature calculates the distribution of all texton ids within a region. Thus, the SHCM feature module extracts a more dense form of information for a specific region.

III. EVALUATION AND RESULTS

The aim of our evaluation study is to gain insight into how well a computationally cheap semantic segmentation system can detect regions and objects in street scenes. Furthermore, we want to explore how much information of a neighbouring region of a pixel is needed to semantically classify the pixel. We use the Intersection over Union score to measure the performance of our system.

We evaluate the performance of our system on the Cityscapes per-category dataset [3]. The data of the dataset is captured in different seasons, under different weather conditions and contains crowded street scenes with a high variety of vegetation, which makes the dataset challenging.

By using the Intersection over Union performance score and by applying our system on the Cityscapes dataset, we make it possible to compare the performance of our system with other approaches published in the Cityscapes benchmark.

The final prediction result of our semantic scene labelling system is dependent on several parameters whose influence we investigate in more detail. Both features use k-means clustering with κ clusters in their textonization process, a window size ω and a window offset o. The window size and offset to the candidate pixel define the size and position of the region that is used to extract information of neighbouring pixels. We evaluate the performance of our system with parameter values within the following ranges: $\kappa = [50, 8000]$, $\omega = [4, 49]$ and $o = [0, 100]$.

In the first set of experiments, we evaluate the performance of the TLF by varying the number of clusters, and the ranges for the window size and the window offset. We achieved best results with an Intersection over Union score of 26.50 by using the TLF feature with $\kappa = 100$, $\omega = [4, 24]$ and $o = [0, 23]$.

By using the SHCM feature in the second part of our experiments, we could increase the Intersection over Union score to 38.29 with $\kappa = 8000$, $\omega = [4, 17]$ and $o = [0, 23]$.

By looking at example segmentation results in Figure 3, we can observe that the TLF feature (column 3) is not able to distinguish between all object classes. By comparing the results of the TLF feature in Column 3 with the results of the SHCM feature in Column 4 of Figure 3, one can recognize a clear improvement of the semantic segmentation. Especially region borders are refined by using the SHCM feature. However, one can see that the classifier has difficulties to distinguish between the class *vehicle* and the class *human* in Column 4. These missclassifications cannot be observed in Column 3, because the classifier hardly detects any pixels that belong to the class *human* by using the TLF feature.

In particular, we found that the system has a bias towards classes representing big object regions such as the classes *flat (road)*, *nature* and *sky*, which can be detected to a certain extent. We assume that a reason for this behaviour is the context-driven classification approach of both features, which tends to interpret small objects in large neighbouring regions as noisy pixels. This bias gets smaller by using the SHCM feature, because the SHCM feature extracts more detailed information from neighbouring regions.

IV. CONCLUSIONS

Semantic, spatial and scale context as well as textural properties play an important role for semantic scene labelling. The TLF feature is a computationally cheap way to extract textural properties and context of data points. In our experimental evaluation, using the SHCM feature instead of the TLF feature increases the accuracy by around 44%. However, the SHCM feature is computationally more complex than the TLF feature. Both features have a bias towards classes representing big objects when used in a random forest classifier and applied on the Cityscapes dataset.

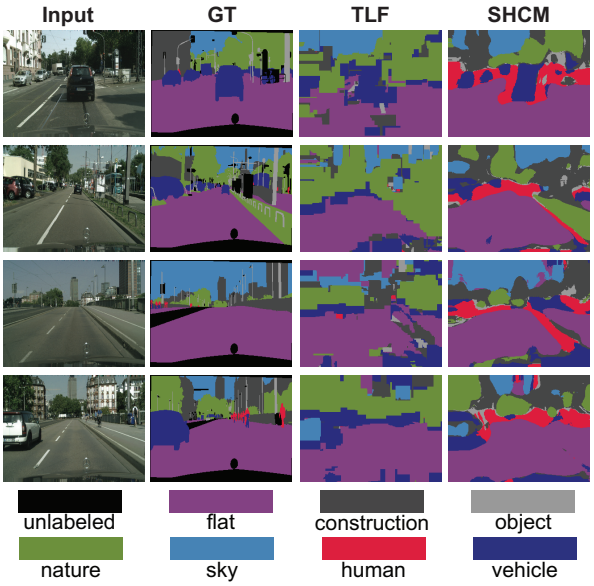


Fig. 3: Example segmentation results. 1st column: input image; 2nd column: ground-truth; 3rd column: results by using the TLF feature; 4th column: results by using the SHCM feature.

REFERENCES

[1] A. Arnab, S. Zheng, S. Jayasumana, B. Romera-Paredes, M. Larsson, A. Kirillov, B. Savchynskyy, C. Rother, F. Kahl, and P. H. Torr, "Conditional Random Fields Meet Deep Neural Networks for Semantic Segmentation: Combining Probabilistic Graphical Models with Deep Learning for Structured Prediction," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 37–52, Jan. 2018.

[2] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[3] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[4] Z. Liu, X. Li, P. Luo, C. C. Loy, and X. Tang, "Deep Learning Markov Random Field for Semantic Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 8, pp. 1814–1828, Aug. 2018.

[5] F. Schroff, A. Criminisi, and A. Zisserman, "Single-Histogram Class Models for Image Segmentation," in *Computer Vision, Graphics and Image Processing*. Springer Berlin Heidelberg, 2006, pp. 82–93.

[6] ——, "Object Class Segmentation using Random Forests," in *Conference on British Machine Vision (BMVC)*. British Machine Vision Association, 2008, pp. 54.1–54.10.

[7] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "TextonBoost for Image Understanding: Multi-Class Object Recognition and Segmentation by Jointly Modeling Texture, Layout, and Context," *International Journal of Computer Vision (IJCV)*, vol. 81, no. 1, pp. 2–23, Jan. 2009.

[8] M. Thoma, "A Survey of Semantic Segmentation," *arXiv:1602.06541*, 2016.

[9] J. Xie, L. Yu, L. Zhu, and X. Chen, "Semantic Image Segmentation Method with Multiple Adjacency Trees and Multiscale Features," *Cognitive Computation*, vol. 9, no. 2, pp. 168–179, Apr. 2017.

Towards Object Detection and Pose Estimation in Clutter using only Synthetic Depth Data for Training

Stefan Thalhammer, Timothy Patten and Markus Vincze

***Abstract*— Object pose estimation is an important problem in robotics because it supports scene understanding and enables subsequent grasping and manipulation. Many methods, including modern deep learning approaches, exploit known object models, however, in industry these are difficult and expensive to obtain. 3D CAD models, on the other hand, are often readily available. Consequently, training a deep architecture for pose estimation exclusively from CAD models leads to a considerable decrease of the data creation effort. While this has been shown to work well for feature- and template-based approaches, real-world data is still required for pose estimation in clutter using deep learning. We use synthetically created depth data with domain-relevant background and randomized augmentation to train an end-to-end, multi-task network to detect and estimate poses of texture-less objects in cluttered real-world depth images of an arbitrary amount of objects. We present experiments and ablation studies on the architectural design choices and data representation with the LineMOD dataset.**

I. Introduction

Assembly systems in manufacturing are subject to increasing number of variants, smaller lot sizes and shorter life cycles. As such, the application of assistance or robotic systems is expected to reduce error rate and increase capacity [6]. Typically, the task of assistance systems in an industrial context is robust object detection as well as pose estimation. However, developing methods that deliver accurate estimates, especially for texture-less objects, is still an open research problem.

Recently deep learning advanced the state of the art for computer vision tasks, however, the advent of deep networks for 3D pose estimation has yet to be fully realized [9]. While deep networks achieve superior performance, they require a huge amount of training data [12]. Capturing and annotating these data is time and labour consuming, often requiring physical instances, which is problematic in fast paced manufacturing environments. Industrial applications typically have CAD data readily available, therefore, we propose to take advantage of this by directly training models for pose estimation of texture-less objects using only synthetic depth images for training.

Accurate pose estimation systems consist of multiple steps, firstly creating initial pose candidates and subsequently refining these using one or more refinement and verification steps. In this work we address the task of creating an initial pose estimate for further refinement.

[1]All authors are with the Faculty of Electrical Engineering and Information Technology, TU Wien, 1040 Vienna, Austria {sthalham, patten, vincze}@acin.tuwien.ac.at

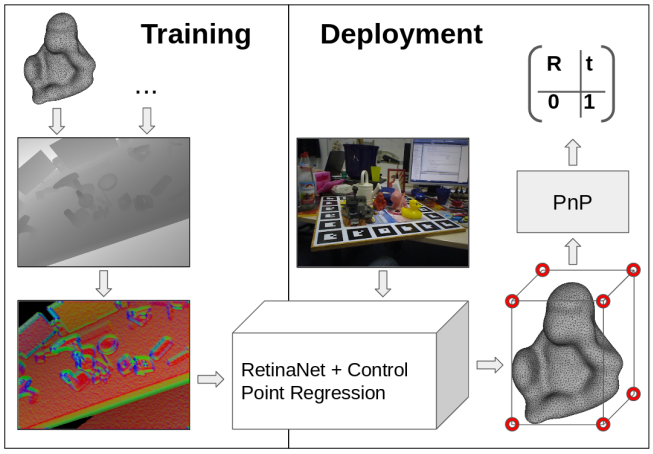


Fig. 1. Control point regression and pose computation in real-world images, trained using rendered and augmented data.

Feature- and template-based approaches for pose estimation employ meshes or point clouds to create templates or hash tables in order to detect objects and estimate their pose at runtime [2], [4], [10]. Consequently, these methods require no real-world data for training. Current deep learning approaches do not close the domain gap, i.e. traversing from synthetic to real-world data without a decrease in performance, and therefore need real-world data during training time. We address the task of training deep pose estimators only from synthetic depth data by rendering and augmenting these data in terms of background information and sensor noise through random shape perturbations.

Pose estimation is a non-trivial task for learning-based approaches, consequently strong approaches tend to train separate models for the detection and the subsequent pose estimation. Often the models for pose estimation are also trained separately for distinct classes [19], [22], [24]. However, end-to-end learning, i.e. training and deploying multiple stages of a vision pipeline at once, is desired to reach a high frame rate. Additionally, when deep architectures are trained on multiple objectives, i.e. in a multi-task fashion, the learned features are stronger, which has been shown to be beneficial for each individual task [5]. Especially when employing pre-trained models in a domain different from RGB, e.g. on depth data, retraining the backbone with additional guidance is desired to create stronger features. Our multi-task, end-to-end models for pose estimation are consequently trained with the capacity to estimate the poses of different classes simultaneously.

In summary, we propose a method for texture-less ob-

ject pose estimation in real-world depth images using only synthetic data for training. Figure 1 outlines our proposed approach.

The contributions are the following:

- We present an approach for simultaneous object detection, classification and pose estimation in a multi-task, end-to-end manner, of an arbitrary number of texture-less objects in real-world depth images. For training we only need meshes of the desired objects.
- We present our findings by evaluating on a standard dataset, the LineMOD [8].

The remainder of the paper is structured as follows. Section 2 summarizes related work. The approach is described in section 3. Section 4 presents the results and evaluation. Section 5 concludes with a discussion.

II. Related Work

The attention of pose estimation research has recently shifted to texture-poor or texture-less objects. Currently, the domain is dominated by template or hand-crafted feature-based approaches [9]. However, estimating object poses using deep architectures is gaining popularity due to the state-of-the-art performance for other computer vision tasks.

While there are manifold approaches to estimate the poses using colored images, depth data is usually only used for refinement [27]. Only very few employ depth data only [17], [16], [22]. However, depth data already gives strong cues about the shape and consequently also about the pose of the object. A major advantage of using only depth data to train networks for pose estimation is the possibility to exclusively train models using CAD data, independent of the color variations of the manufactured object.

A. Classical Approaches

Point-Pair features create a strong basis for pose estimation pipelines. Point pairs are matched between the test scene and the provided models then stored in a hash table. Votes are accumulated to create hypotheses, subsequently refined using ICP and non-maxima are suppressed. Hypotheses are favoured when the detected 3D edges match the model contours. These approaches do not use RGB data and have multiples stages, subsequently removing or refining pose hypotheses [4], [26].

Template matching methods can also exhibit strong pose estimation results. Hodan *et al.* [10] use a sliding window with cascading evaluation. Pre-filtering differentiates between the object and background. Hypotheses are generated for every window by hashing. Hypotheses verification consists of verifying size, normals, gradients, depth template and color. Object pose refinement is initialized from the verified hypotheses using particle swarm optimization.

While these approaches usually yield strong pose estimates, they are slow compared to end-to-end learning based approaches and lack high detection performance.

B. Learning-based Approaches

Learning-based approaches yield strong results for some pose estimation tasks, but they are currently not on par with classical approaches.

Random forest approaches can be used to sample pose hypotheses, which are used to choose and iteratively refine promising pose estimates [1], [23].

A common practice for pose estimation using deep learning is to treat the translation and the rotation part of the pose separately [11], [14], [19], [22], [27]. While the center of the detected bounding boxes already results in feasible translation estimates in image space [11], translation regression is desired when dealing with occlusions [22], [27]. When estimating the rotational part of the pose separately either regression [27] or classification [11], [22] can be employed. While regression of the rotation is computationally more efficient and natural due to the smooth representation space, classification yields better results in practice [11], [22].

For pose estimation using depth only, pixel-wise segmentation can be employed to create masks and then to be matched against previously computed templates [16] resulting in similar performance as classical approaches.

One of the strongest approaches for object pose estimation using deep learning is the regression of virtual control points [3], [19], [24], i.e. regressing a 3D bounding box projected into image space and alike. The regressed control points are used to solve Perspective-n-Points (PnP) in order to obtain a pose estimation. This approach is used for RGB images and models are trained separately for each object or even decoupled from detection. Considering the task of pose estimation as a translation regression problem is promising because CNNs exhibit translation equivariance between image and feature space.

We employ one model for detection, classification and pose estimation, independent of the amount of objects of interest. Our approaches modifies RetinaNet [13] to include pose estimation in their one-staged architecture. We use RetinaNet due to the very strong object detection performance on diverse datasets and its fast computation, running with approximately ten frames per second (fps). We consequently regard pose estimation as a multi-task, end-to-end learning approach, using only translation regression and subsequent PnP for pose calculation. Compared to other pose estimation approaches using depth images, our method is one-staged, uses no refinement and deals with diverse objects simultaneously at approximately five fps.

III. 6D Pose Estimation from Synthetic Data

We render synthetic depth data in Blender from a virtual scene resembling the area of deployment of our model. These data are subsequently augmented and annotated using a randomized noise model and are used for supervised training. We base our architecture on RetinaNet [13] and add an additional branch in order to enable multi-task, end-to-end 6D object pose estimation in complete scenes. The additional branch takes the features concatenated by the feature pyramid network as inputs and outputs n virtual control points as

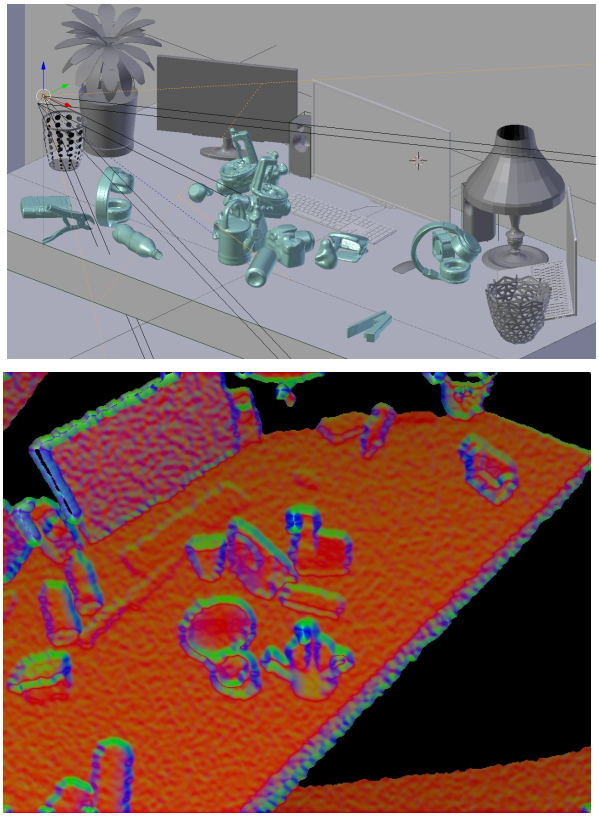


Fig. 2. Virtual scene to render synthetic training data from (top). Augmented synthetic depth image used for training (bottom).

defined in [3]. RetinaNet is currently considered one of the strongest object detectors and additionally exhibits tight bounding box estimates, thus ideally fitted for control point regression.

Since CNNs yield translation equivariance between image and feature space it is reasonable to regard pose estimation as regression tasks, in the context of deep learning. The authors of [24] showed that a similarly simple approach for RGB-data can achieve state-of-the-art results without limiting general applicability.

A. Training Dataset Creation

We use only synthetically created training data and deploy our model on real-world depth scans. We create synthetic depth data with a diverse scene setup and various background information and additionally apply noise heuristics in order to produce training data with high variation regarding views and occlusion patterns [25]. This has been shown to generate high quality data to train deep architectures for object detection and classification. An example for a virtual scene can be seen in the top image of Figure 2.

1) Data Rendering: We render 15,000 training images of virtual scenes exhibiting the expected variations of the area of deployment. For each image, we randomly place five to eight objects of interest with repetition. The objects are annotated with a bounding box, 6D pose and pixel-level class correspondences. The camera pose is sampled similar to the expected poses in the test set.

The output of the synthetic data creation step is a depth image, a binary mask indicating visible image regions and a mask indicating pixel-level class correspondences. The binary mask provides information about image regions with invalid depth values depending on the imaging geometry of infrared depth sensors.

2) Dataset Creation: The synthetic dataset used for training is created by combining the outputs of the rendering step.

The binary mask is applied to the synthetic depth images using randomized morphological operations. This results in missing image regions similar to real-world depth scans. Blur is added to minimize the discrepancy between depth gradients in the real-world and synthetic images. The synthetic depth values are rounded to the nearest quantization value based on the hypothesized sensor's depth resolution. This operation reduces the domain shift between synthetic and real depth images. Additional noise is added to the quantized depth values using an offset chosen randomly from a Gaussian distribution, assuming non-linearly increasing noise. Further randomness of the appearance of occluded scene parts, depth and lateral noise is added by warping the depth images through the application of pixel offsets using the Perlin noise technique [18], which was shown to significantly improve the performance of trained models [25]. The augmentation process is sampled twice per rendered image to create a dataset of approximately 30,000 images.

B. Network Architecture

We use RetinaNet[1] [13] with ResNet-50 [5] backbone and pretrained on ImageNet [20] as feature extractor and detector. We add an additional network branch for control point regression, parallel to the classification and detection branches.

1) Data Representation for Pose Estimation: We regress eight control points to exactly encapsulate the object's dimensions in 3D. In general, an arbitrary number of control points can be chosen and regressed. Those points are virtual, i.e. they do not represent actual object parts, thus can be chosen arbitrarily in the objects' coordinate frame. Using the camera intrinsics and the calculated corresponding object pose these points are projected into image space.

The design of the additional branch is based on RetinaNet's bounding box regression branch. We slightly modify it by adding *l2* regularization of the weights of every convolution layer with the hyperparameter set to 0.001. We perform experiments using other penalties, dropout and batch normalization but that resulted in decreased performance. The 16 values representing the x and y components of the eight control points are regressed for every object class (n) separately. The architecture of our control point estimation branch is shown in Figure 3.

2) Loss: The overall loss to minimize is defined as

$$L = L_{box} + L_{cls} + L_{box3D} \quad (1)$$

[1]https://github.com/fizyr/keras-retinanet

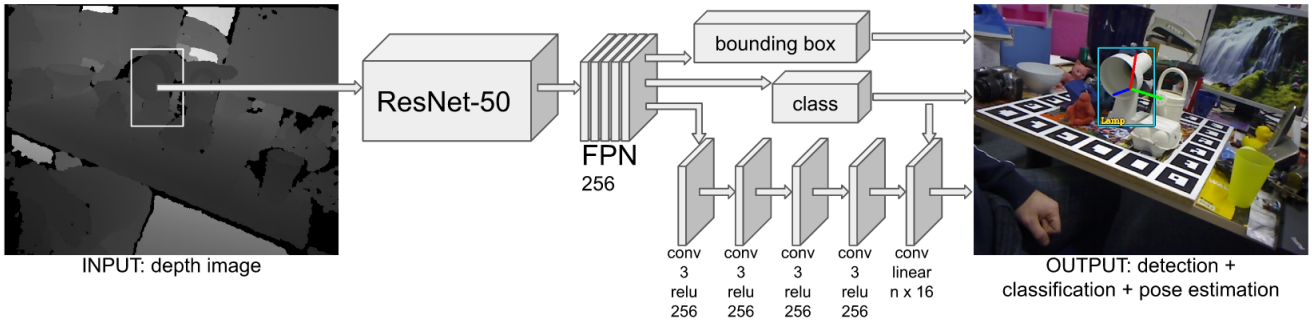


Fig. 3. Our multi-task, end-to-end network architecture.

where L_{box}, L_{cls} and L_{box3D} are the losses for bounding box regression, object classification and control point regression, respectively. We use smoothed *l1* loss for bounding box regression and focal loss for classification. The control point regression loss is formally, per image, defined as

$$L_P = \frac{1}{m} \sum_{o \in gt} \left|\left| Proj_T(cp) - Proj_{\hat{T}}(cp) \right|\right|^k \qquad (2)$$

where m is the total number of object instances o in the ground truth gt of an image, $Proj_T$ and $Proj_{\hat{T}}$ are the projection and the estimation projection of the control points cp onto the image plane and k the desired norm. As norm we use smooth *l1*.

We weight the contribution of each of the loss parts differently. Our experiments showed that weighting L_{box3D} such that its magnitude is twice the magnitude of L_{box} and four times the magnitude of L_{cls}, results in good recall and precision regarding detections and reasonable pose estimates.

The estimated control points during test time are reprojected into 3D space and the object's pose is simultaneously estimated using PnP. For our purpose we use the iterative RANSAC based algorithm.

3) Data Augmentation: In order to prevent the network from overfitting to the limited amount of training data we apply extensive data augmentation of the training images. Every input image is randomly augmented online using a superposition of translation and scaling up to 20 percent each.

IV. Experiments

All the experiments are conducted on the LineMOD dataset [8]. LineMOD contains approximately 1,100 test images for each of the 13 dataset objects. Each object is placed in a heavily cluttered scene and annotated with bounding box, class and 6DoF pose. We provide ablation studies, specific for the task at hand. Comparing image preprocessing, regularization strategies and possible loss functions for the control point regression branch. In order to provide a reasonable comparison against the state of the art we compare against [4], [7] and [23].

A. Experimental Setup

For testing we use only the depth images of the LineMOD dataset that are captured using a Microsoft Kinect V1. Images are converted to three channel RGB images, coloured based on the depth gradient using the approach of Nakagawa *et al.* [15]. The ablation study shows the benefit of this. Image regions with missing depth values are inpainted using OpenCV[2] and depth cuts are applied to image regions farther than two meters.

Our networks are trained using the Adam optimizer with adaptive learning rate. Ablation studies are trained for 20 epochs using 10,000 images. Comparison against state of the art is trained for 100 epochs with 30,000 images. We use a batch size of one and an initial learning rate of 10^{-5}. We choose the best performing model after the above mention amount of epochs to provide comparisons. All networks are trained on a Nvidia GeForce GTX 1080.

B. Ablation Studies

We perform three studies to adapt RetinaNet to our needs. Firstly we compare different dataset augmentations and depth image representations, secondly we evaluate which loss to use for bounding box regression and thirdly we show an ablation study regarding regularization applied by the control point regression branch. All studies are performed on a validation set of 2400 images, taken uniformly from all classes of the LineMOD dataset.

1) Image Representation: We compare different options for of the augmentations applied to the depth images, as well as possibilities for depth to three channel image conversion.

Table I provides results in terms of recall and precision of detections with an Intersection over Union (IoU) higher than 0.5, and percentage of rotation estimates below a five degree deviation from the ground truth. *Depth* refers to repeating the depth images three times and converting it to eight bit, *rgb* refers to color coding the depth images based on the normal direction [15]. Options for augmentation are either *perlin*, which refers to only augmenting the synthetic training images by removing occluded image regions due to the imaging geometry and warping pixel locations using Perlin noise, and *full*, which refers to additionally adding blur and depth noise and quantizing depth values as described in section III-A.

Using color coded depth images with full augmentation applied shows best detection and rotation estimation results.

[2]https://opencv.org/

TABLE I

EXPERIMENTS REGARDING IMAGE DATA REPRESENTATION AND AUGMENTATION APPLIED.

Representation	*depth*		*rgb*	
Augmentation	*perlin*	*full*	*perlin*	*full*
Recall	83.61	82.72	86.67	**89.77**
Precision	**94.65**	89.54	93.91	93.59
Rotation $< 5°$	6.93	5.07	6.76	**7.33**

TABLE II

LOSS FUNCTIONS USED FOR CONTROL POINT REGRESSION AND THEIR INFLUENCE ON THE DERIVED ROTATION ESTIMATES.

Loss	*mse*	*l1*	*smooth l1*
Rotation $< 5°$	6.03	4.78	**7.33**

2) Loss Function: Using a pose regression branch similar to the bounding box regression branch suggests to use a similar loss function.

Table II presents results of different loss functions used for regressing the virtual control points. Care was taken to weight the individual loss parts in a way to preserve the above mentioned ratio between bounding box, classification and pose losses. The percentage of rotation estimates below a five degree deviation from the ground truth is provided for mean squared error (*mse*), absolute error (*l1*) and Huber loss (*smooth l1*). *Smooth l1* provides the strongest control point estimates evaluated using rotation estimation.

3) Regularization: Regularization reduces the generalization error, thus reducing the model's performance discrepancy between the training and the validation/test set. Since our data domains for source and target are very different it is not straight forward to decide which regularization strategy to apply. Table III provides information about regularization applied and their influence on the networks performance. All regularization strategies applied here are only applied to the control point regression branch. Batch normalization (*bn*) is applied to each convolution layer's output except from the last, weight decay (*wd*) is applied to all the weights of the convolution kernels and dropout (*do*) tested here is applied to the inputs of the last convolution layer with a probability of 20 percent.

The metric *5cm 5°* refers to the metric defined in [21], *6D pose* refers to the metric defined by [8], where we evaluate on ten percent of the mesh-model diameter, and *proj. 2D* refers to the reprojection of the object mesh to the image using the estimated pose. The pose is considered as true if the average pixel difference is smaller than a threshold. For this we use five pixels. Results show that only *l2* weight decay with a hyperparameter of 0.001 improves results on the validation set.

C. Comparison Against the State of the Art

For evaluation against the state of the art we use the metric defined in [8] as well as the F1-score from the harmonic mean of the precision and recall as in [23]. Unlike [23] we train only one model and not separate models for every object. While [23] and [7] use RGB and depth, [4] only uses depth data. Consequently, we consider [4] as the most relevant method to compare against. Table IV provides a comparison of our approach against the state of the art.

TABLE III

REGULARIZATION APPLIED BY THE NETWORK, TESTED ON DIFFERENT METRICS.

Metric	*bn*	*wd(0.01)*	*wd(0.001)*	*wd(0.0001)*	*do(0.2)*
Rot. $< 5°$	0.0	6.84	**7.85**	7.03	1.83
5cm 5	0.0	3.36	**5.02**	3.76	0.0
6D pose	0.0	4.82	**6.62**	6.17	0.35
proj. 2D	0.0	18.06	22.33	**23.48**	0.5

TABLE IV

F1-SCORE COMPARISON OF OUR METHOD AGAINST COMMON STATE-OF-THE-ART METHODS.

Method	LINEMOD[7]	Drost[4]	Tejani[23]	ours
Ape	53.3	62.8	85.5	34.0
Benchvise	84.6	23.7	96.1	52.2
Driller	69.1	59.7	90.5	31.6
Cam	64.0	51.3	71.8	52.4
Can	51.2	51.0	70.9	51.2
Iron	68.3	40.5	73.5	46.5
Lamp	67.5	77.6	92.1	26.0
Phone	56.3	47.1	72.8	66.2
Cat	65.6	56.6	88.8	60.6
Holepuncher	51.6	50.0	87.5	46.6
Duck	58.0	31.3	90.7	44.6
Eggbox	86.0	82.6	74.0	54.0
Glue	43.8	38.2	67.8	30.5
Average	63.0	51.7	81.7	46.8

Our method exhibits comparable results to Drost *et al.* [4] when taking the false-positive rate into account. When considering detections above 0.5 IoU as true our method exhibits a recall and precision of 96.71 and 94.43 percent respectively on the LineMOD dataset.

Figure 4 shows control point estimation of the object *Glue* on the left and the corresponding ground truth on the right. A severely distorted 3D bounding box estimation of the object is visible, the box appears to vanish in one dimension. This happens often for the object *Glue*, leading to the conclusion that for objects with a small size along one dimension the control points have to be chosen significantly higher than the corresponding dimension.

Figure 5 shows the 3D bounding box of the object *Lamp*, defined by the estimated control points, on the left and again the corresponding ground truth on the right. A detection with good alignment of the estimated 3D box and pose, with respect to the ground truth, is visible.

V. CONCLUSION

In this paper we presented a deep learning architecture for multi-task, end-to-end 6D object pose estimation for an arbitrary number of objects from only depth images. The architecture was trained entirely from synthetic data that is generated to resemble real-world data. Experiments

Fig. 4. Warped 3D box detection of the object *glue*

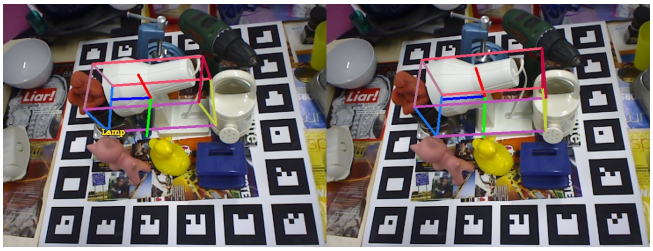


Fig. 5. 3D box detection of the object *lamp*

with the LineMOD dataset showed promising results. Our ablation studies provide valuable information for detection, classification and pose estimation of texture-less objects in clutter.

Future work will tackle the improvement of the 3D bounding box regression results. Experiments with other data modalities will also be conducted. We will furthermore investigate the benefits of enforcing orthogonality on the boxes. Additional architecture modifications will be tested to disentangle the control point estimation per object further in order to enhance pose estimation. Other directions for future work include addressing object symmetries and tuning the parameters for the generation of the synthetic training data to randomize the applied noise more specifically to the desired sensor.

REFERENCES

[1] E. Brachmann, F. Michel, A. Krull, M. Ying Yang, S. Gumhold, *et al.*, "Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3364–3372.

[2] A. G. Buch, L. Kiforenko, and D. Kraft, "Rotational subgroup voting and pose clustering for robust 3d object recognition," 10 2017, pp. 4137–4145.

[3] A. Crivellaro, M. Rad, Y. Verdie, K. M. Yi, P. Fua, and V. Lepetit, "A novel representation of parts for accurate 3d object detection and tracking in monocular images," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 4391–4399.

[4] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: Efficient and robust 3d object recognition," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 998–1005.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[6] S. Hinrichsen, D. Riediger, and A. Unrau, "Assistance systems in manual assembly," in *Proceedings 6th International Conference on Production Engineering and Management*, 2016, pp. 3–13.

[7] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, and V. Lepetit, "Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes," in *2011 international conference on computer vision*. IEEE, 2011, pp. 858–865.

[8] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Asian conference on computer vision*. Springer, 2012, pp. 548–562.

[9] T. Hodaň, F. Michel, E. Brachmann, W. Kehl, A. G. Buch, D. Kraft, B. Drost, J. Vidal, S. Ihrke, X. Zabulis, C. Sahin, F. Manhardt, F. Tombari, T.-K. Kim, J. Matas, and C. Rother, "Bop: Benchmark for 6d object pose estimation," in *Computer Vision – ECCV*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 19–35.

[10] T. Hoda, X. Zabulis, M. Lourakis, . Obdrlek, and J. Matas, "Detection and fine 3d pose estimation of texture-less objects in rgb-d images," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sep. 2015, pp. 4421–4428.

[11] W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab, "Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1521–1529.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[13] T.-Y. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *2017 IEEE International Conference on Computer Vision*, pp. 2999–3007, 2017.

[14] F. Manhardt, W. Kehl, N. Navab, and F. Tombari, "Deep model-based 6d pose refinement in rgb," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 800–815.

[15] Y. Nakagawa, H. Uchiyama, H. Nagahara, and R.-I. Taniguchi, "Estimating surface normals with depth image gradients for fast and accurate registration," in *3D Vision, International Conference on*. IEEE, 2015, pp. 640–647.

[16] K. Park, T. Patten, J. Prankl, and M. Vincze, "Multi-task template matching for object detection, segmentation and pose estimation using depth images," in *International Conference on Robotics and Automation*, 2019.

[17] K. Park, J. Prankl, M. Zillich, and M. Vincze, "Pose estimation of similar shape objects using convolutional neural network trained by synthetic data," in *Proceedings of the OAGM-ARW Joint Workshop*, 5 2017, pp. 87–91.

[18] K. Perlin, "Improving noise," in *ACM Transactions on Graphics*, vol. 21. ACM, 2002, pp. 681–682.

[19] M. Rad and V. Lepetit, "Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth," in *The IEEE International Conference on Computer Vision*, Oct 2017.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, pp. 211–252, 2015.

[21] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in rgb-d images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2930–2937.

[22] J. Sock, K. Kim, C. Sahin, and T. Kim, "Multi-task deep networks for depth-based 6d object pose and joint registration in crowd scenarios," in *Proceedings of British Machine Vision Conference*, 7 2018.

[23] A. Tejani, R. Kouskouridas, A. Doumanoglou, D. Tang, and T. Kim, "Latent-class hough forests for 6 dof object pose estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 1, pp. 119–132, Jan 2018.

[24] B. Tekin, S. Sinha, and P. Fua, "Real-time seamless single shot 6d object pose prediction," 06 2018, pp. 292–301.

[25] S. Thalhammer, K. Park, T. Patten, M. Vincze, and W. Kropatsch, "Sydd: Synthetic depth data randomization for object detection using domain-relevant background." TUGraz OPEN Library, 2019, pp. 14–22.

[26] J. Vidal, C.-Y. Lin, and R. Marti, "6d pose estimation using an improved method based on point pair features," 04 2018, pp. 405–409.

[27] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes," 2018.

A Two-Stage Classifier for Collagen in Electron Tomography Images using a Convolutional Neural Network and TV Segmentation

Verena Horak[1,2] and Kristian Bredies[1,2]

***Abstract*— We present an easily realizable practical strategy for the segmentation of tissue types in microscopy images of biological tissue. The strategy bases on a convolutional neural network (CNN) classifier that requires a low amount of manually-labeled data. Spatial regularity of the segmented images is enforced by a total variation (TV) regularization approach. The proposed strategy is applied to and tested on collagen segmentation in electron tomography image stacks.**

I. INTRODUCTION

In cell biology, one often seeks segmentation of microscopy images according to some biological tissue types or structures. Since manual labeling is very tedious, time consuming, and error-prone, automated classification algorithms are desired. In this respect, many segmentation methods have been proposed and studied, usually focusing on cell segmentation [10] where cell boundaries are clearly defined. In contrast to that, more complex structures such as collagen, for instance, are often only vaguely visible and subject to noise, in particular, if the images were reconstructed from imperfect data, which is often the case in electron tomography. Machine-learning approaches have the potential to overcome this problem [6], [7] but require a sufficient amount of manually-labeled training data which is often not available. One approach is to generate more training data is to consider all possible patches of a specified size within the training images and to train a patch-based classifier. Such a classifier is, however, unaware of the spatial structure of the patches and usually produces binary images with irregular labeled regions. A shape regularization is thus necessary.

In this work, we report on the application of a binary two-stage classifier to automated collagen segmentation that, similar to [8], employs, on the one hand, a patch-based machine-learning approach and, on the other hand, provides regular label regions via total variation (TV) regularization. The method requires only a single manually labeled image stack as training data as well as minimal human interaction.

II. THE TWO-STAGE CLASSIFIER

The implemented classifier uses the same ideas as [8] for a different application. We assume that a training set of images to label as well as a manually-generated labeling is available, where is latter might the affected by human error. We further assume that only local information is necessary in order to determine whether an image pixel has to be labeled or not.

[1]Institute of Mathematics and Scientific Computing, University of Graz, Austria, Email: {verena.horak, kristian.bredies}@uni-graz.at

[2]BioTechMed-Graz, Austria, https://biotechmedgraz.at/en/

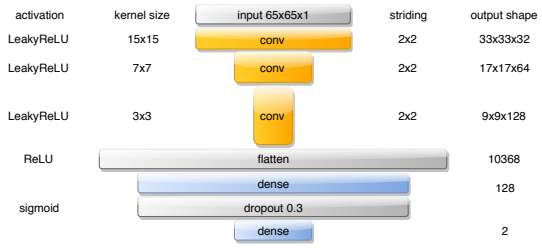


Fig. 1. Example of a weak classifier for patches of size 65×65.

For this reason, we pursue a patch-based approach which leads to all possible patches and single 0/1 values constituting the training set pairs. This might, however, lead to artifacts and irregular contours. Hence, we propose to learn the basic distribution of collagen based on a patch surrounding the pixel under investigation, and then using a regularization to obtain artifact-free, regular regions.

A. Stage 1: A weak patch-based machine-learning classifier

Our approach is to predict each pixel in the label image from a so-called patch, that is a sub-image of a specified window size with the corresponding pixel to label as its center. With r, c, w indicating the number of rows, columns of the image, and the window size, respectively, we obtain from every image $(r-w+1)\cdot(c-w+1)$ of such patches of size $w \times w$, provided that $r \geq w$ and $c \geq w$. Using a certain subset of these patches as training data, a convolutional neural network (CNN) is trained to weakly predict the label corresponding to the center pixel of a given patch, i.e., the outcome is not a binary 0/1-result but a value in $[0,1]$ that can be interpreted as a probability. An example for such a predictor can be found in Fig. 1.

B. Stage 2: A binary TV-regularized classifier

Due to the patch-based prediction, spatial information between neighboring pixels is not taken into account. This typically yields irregular contours and artifacts. To get rid of these disruptive factors and to obtain the final binary result, a classifier based on a total variation (TV) regularizer is used [2]. More specifically, if l_0 is image of predicted labels from Stage 1 with values in $[0,1]$, we solve the problem

$$\min_{l} \int \lambda (b - l_0) \cdot l \, \mathrm{d}x + \mathrm{TV}(l) \quad \text{subject to} \quad 0 \leq l \leq 1, \quad (1)$$

where $b \in [0,1]$ is a bias parameter corresponding to a threshold and $\lambda > 0$ a regularization parameter controlling the regularity of the contours. Afterwards, the minimizer l^* is thresholded (for instance, at 0.5), in order to obtain a

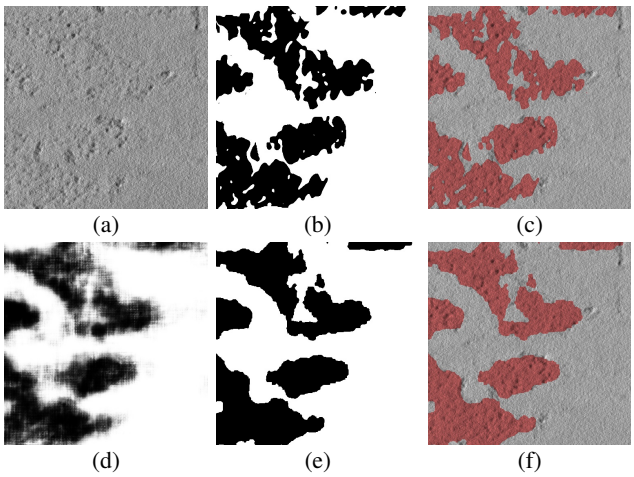


Fig. 2. Visualization of the two-stage classifier for a subregion of a test image. (a) the microscopy image to label, (b) the manually-generated label image, (c) the manual labeling laid over the microscopy image, (d) the weak prediction of Stage 1, (e) the regularized binary prediction of Stage 2, (f) the predicted label image (e) laid over the input image.

TABLE I

QUANTITATIVE EVALUATION OF BOTH STAGES OF THE CLASSIFIER. $P_i = v$ REPRESENTS A PREDICTION OF A PIXEL VALUE v IN STAGE i, WHEREAS $L = v$ MEANS A PIXEL LABEL OF v.

	$P_1 = 1$	$P_1 = 0$	$P_2 = 1$	$P_2 = 0$
$L = 1$	20.63%	3.42%	21.10%	2.94%
$L = 0$	3.45%	72.50%	3.61%	72.45%

binary image as final result. This yields a global solution of the corresponding shape optimization problem where the constraints in (1) are replaced by $l \in \{0,1\}$ in each pixel [4]. A primal-dual algorithm [3] is used for the solution of (1).

III. NUMERICAL EXPERIMENTS

Our numerical studies were carried out on an image stack of 100 images of size 2048×2048. These images were obtained by tomographic reconstruction from a tilt series of transmission electron microscopy (TEM) images of human aortic tissue. The reconstruction was computed with the IMOD software package [9] and the collagen in this data set was labeled manually, see Figure 2 for a section.

A. Training the patch-based classifier

For the patch building process, a window size of 65×65 was chosen, yielding $(2048 - 65 + 1)^2 = 3936256$ patches for each image. For Stage 1, the patches of 80 images were subjected to a random permutation and taken as training data for the CNN described in Fig. 1 with a binary cross entropy loss function. The network was realized by Keras 2.2.4 [5] and Tensorflow 1.12.0 [1], and the computations were performed on a NVIDIA Tesla K40c GPU. The training was stopped after 90 hours of computation.

B. Classification results

We present quantitative and visual results for both stages of the classifier. Table I, left shows the performance of Stage 1 if a threshold of 0.43 is taken for binary classification. This value was manually chosen to maximize the accuracy and leads to an accuracy of 93.13%, a precision of 85.68%, and a phi coefficient of 81.2%. For Stage 2, the performance is displayed in Table I, right. An accuracy of 93.45%, a precision of 85.41%, and a phi coefficient of 82.25% are obtained by choosing $b = 0.4$ and $\lambda = 4.5$. These values were manually found to be optimal by analyzing the accuracy as well as by visual inspection. The effect of each stage on a region of a test image is shown in Fig. 2. One can see that the TV-regularization step in Stage 2 is indeed beneficial for the visual appearance of the predicted label image. The quantitative results in Table I moreover underline that Stage 2 does not deteriorate the performance of the learned classifier in Stage 1, and even seems to improve it.

IV. CONCLUSIONS

This study shows the effectiveness of the presented method for labeling problems that, e.g., researchers working with microscopy images commonly face. It can easily be adapted to other types of tissues beyond collagen by providing suitable manually-segmented data. The method could, for instance, enable to significantly increase the throughput for image-based analysis of biological tissue samples.

ACKNOWLEDGMENTS

The training and test image stack as well as the manual segmentation are courtesy of the Institute of Biomechanics, Graz University of Technology, and the Gottfried Schatz Research Center, Medical University of Graz. The authors thank Anna Pukaluk and Gerd Leitinger for providing the data sets and valuable feedback on the algorithms.

REFERENCES

[1] M. Abadi *et al.*, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," https://www.tensorflow.org/, 2015.
[2] V. Caselles, A. Chambolle, and M. Novaga, *Total Variation in Imaging*. Springer New York, 2011, pp. 1016–1057.
[3] A. Chambolle and T. Pock, "A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging," *J. Math. Imaging Vis.*, vol. 40, no. 1, pp. 120–145, 2011.
[4] T. Chan, S. Esedoglu, and M. Nikolova, "Algorithms for Finding Global Minimizers of Image Segmentation and Denoising Models," *SIAM J. Appl. Math.*, vol. 66, no. 5, pp. 1632–1648, 2006.
[5] F. Chollet *et al.*, "Keras," https://keras.io/, 2015.
[6] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images," in *Proceedings of NIPS 2012*, pp. 2843–2851.
[7] A. Fakhry, T. Zeng, and S. Ji, "Residual Deconvolutional Networks for Brain Electron Microscopy Image Segmentation," *IEEE Transactions on Medical Imaging*, vol. 36, no. 2, pp. 447–456, 2017.
[8] P. Kainz, M. Pfeiffer, and M. Urschler, "Segmentation and Classification of Colon Glands with Deep Convolutional Neural Networks and Total Variation Regularization," *PeerJ*, vol. 5, p. e3874, 2017. [Online]. Available: https://doi.org/10.7717/peerj.3874
[9] J. R. Kremer, D. N. Mastronarde, and J. McIntosh, "Computer Visualization of Three-Dimensional Image Data Using IMOD," *Journal of Structural Biology*, vol. 116, no. 1, pp. 71 – 76, 1996.
[10] E. Meijering, "Cell Segmentation: 50 Years Down the Road," *IEEE Signal Proc Mag*, vol. 29, 2012.

Semantic Image Segmentation using Convolutional Neural Nets for Lawn Mower Robots

Armin Pointinger[1] and Gerald Zauner[2]

Abstract— **Robots are becoming more and more part of our daily lives. They take on different tasks to make our everyday life easier. In order to be able to fulfill these tasks expediently, high demands are placed on the robots with regard to their abilities. Accordingly, lawn mower robots are also expected to achieve a perfect mowing result and ease of handling. To do this, the robot must be able to find its way around and be able to react appropriately.**

I. INTRODUCTION

Within a few years, semantic image segmentation has become a key task in image processing. This rapid progress already allows a paradigm shift in many areas with regard to the solution approach of many problems. Thus, it is obvious to use semantic image segmentation for autonomous lawnmowers. The resulting benefits are good orientation abilities in previously unseen environment, optimal path planning and the reduction of danger to humans and animals. Compared with conventional lawnmower robots, whose navigation usually relies on a perimeter wire, this could be an alternative in future.

This master's thesis deals with the comparison of different network architectures for semantic segmentation with respect to their suitability for use in autonomous lawn mowers. Sufficient segmentation accuracy and real-time capability are used as criteria for this.

II. APPROACH

By using different network architectures their advantages and disadvantages are evaluated. TensorFlow was used as framework and the implementation of the models is available on the following link on GitHub. [2] Extensive data material is essential for Deep Learning. No data set was available for this task and had to be created first. In order to keep the effort manageable, a two-class problem was assumed. One class is represented by the lawn and one class by the environment. In order to achieve good results, different hyper parameters are optimized manually and automatically during training. Using these results, performance tests were done on different hardware platforms.

A. Dataset

A data set was created which was used for this problem. The images were taken with an RGB camera from the perspective of a lawn mower robot at a distance of 26 cm from the ground. The original resolution of 4032 x 3024 pixels was reduced to 1280 x 704 pixels. Mainly private gardens represent the 25 different locations, which were also selected with regard to different lawn conditions. For each of these individually acquired images the ground truth was created as a label and a data set was generated, which divided the image data as follows.

- 860 Training images
- 89 Validation images
- 91 Test images

B. Training

Among other things, the models used have been selected in a way of providing a comparison of new, older, more complex and simpler architectures. In order to achieve good training results, the hyper parameters need to be adjusted. Therefore empirical attempts were made. The respective models require different hyper parameters for good results. The training process was done with a NVIDIA GeForce GTX 1080. No algorithms for data augmentation were used.

III. EXPERIMENTAL RESULTS

ResNet-101 was used as base model so that the test results could be compared as closely as possible. Table I shows used models and the related per class mean IoU (intersection over union). The result shows that all models provide very similar results in terms of accuracy. The average run time refers to one frame and was calculated with a NVIDIA GeForce GTX1080.

TABLE I
TEST RESULTS

Model	Mean IoU	Average Run Time
DeepLabV3+ [3]	0.966	0.073 sec
BiSeNet [6]	0.955	0.088 sec
DenseASPP [5]	0.952	0.037 sec
GCN [4]	0.959	0.098 sec

The aim is to obtain a system with real-time capability. This requires appropriate computing power. Table II lists three different hardware systems with their achieved inference speed. This measurement was performed with the DeepLabV3+ model. In order to make inference also work with the Jetson TX2, the image resolution was reduced to 640 x 352 pixels. This resolution was used for all three systems. It was observed that the mean IoU (intersection over union) is slightly lower than with the resolution of 1280 x 704 pixels, which was used for training. Since only temporal aspects are

This work was supported by the FH Oberösterreich Forschungs & Entwicklungs GmbH and the Ginzinger Electronic Systems GmbH.

[1]Armin Pointinger, University of Applied Sciences Upper Austria, 4600 Wels, Austria armin.pointinger@gmx.at

[2]Gerald Zauner, University of Applied Sciences Upper Austria, 4600 Wels, Austria gerald.zauner@fh-wels.at

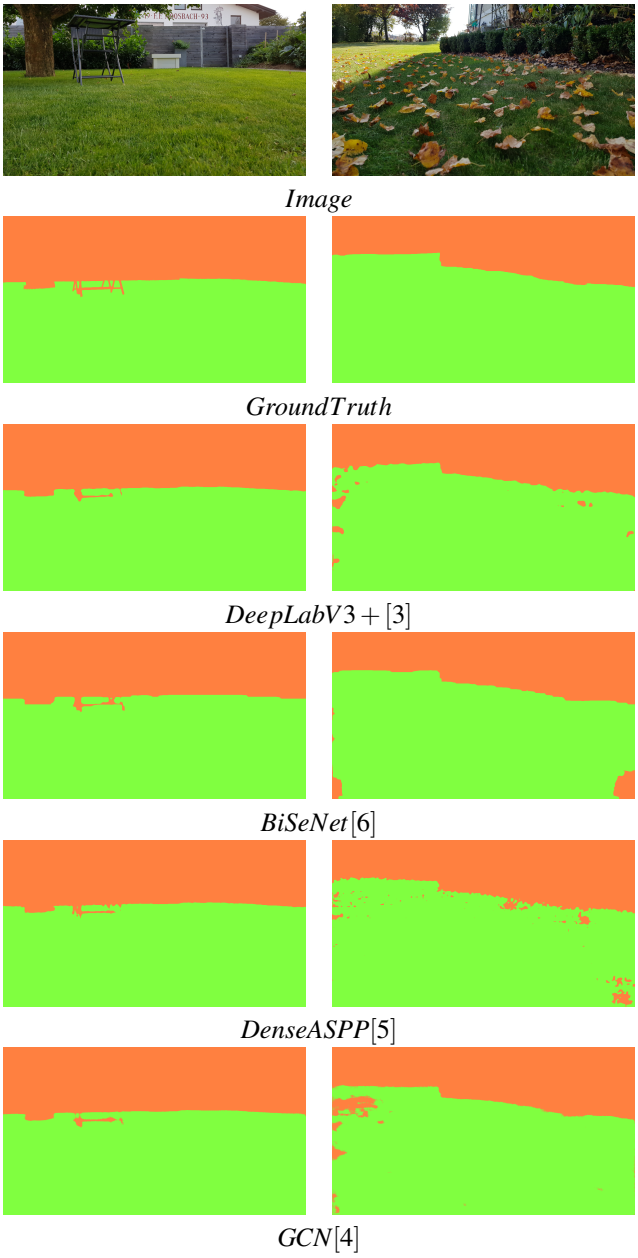
listed in table II, it should be noted that the labels generated by the Jetson TX2 did not contain useful information.

TABLE II
RUN TIME

Hardware	Average Run Time	Frames per Second
GeForce GTX1080	0.030 sec	33.3
Quadro P2000	0.085 sec	11.8
Jetson TX2	0.403 sec	2.48

Figure 1 shows two different test images, their ground truth and the predicted labels of different models. This Link can be used to watch a video about inference. [1]

Fig. 1. Test Images



IV. CONCLUSIONS

Good results were achieved with all tested models. In addition, the data set used is very small and should be extended for better results. In order to guarantee real-time capability, high resolution images require high computing power. The performance of a Jetson TX2 module is too low for this task, and the price for more powerful hardware is currently too high to compete with conventional consumer lawn mower robots. Nevertheless, semantic image segmentation provides lawn mower robots a good basis for terrain orientation and lawn recognition.

ACKNOWLEDGMENT

This work was supported by the FH Oberösterreich Forschungs & Entwicklungs GmbH and the Ginzinger Electronic Systems GmbH.

REFERENCES

[1] (2019) Inference video. [Online]. Available: https://youtu.be/GPCfSAO0TYc
[2] (2019) Semantic segmentation suite. [Online]. Available: https://github.com/GeorgeSeif/Semantic-Segmentation-Suite
[3] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," *ECCV*, 2018.
[4] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large kernel matters - improve semantic segmentation by global convolutional network," 2017.
[5] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "Denseaspp for semantic segmentation in street scenes," *CVPR*, 2018.
[6] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," *ECCV*, 2018.