Proceedings of the Joint Austrian Computer Vision and Robotics Workshop 2020 DOI: 10.3217/978-3-85125-752-6-04

Autonomous Grasping of Known Objects Using Depth Data and the PCA

Dominik Steigl, Mohamed Aburaia, Wilfried Wöber
UAS Technikum Wien
{mr18m007,mohamed.aburaia,wilfried.woeber}@technikum-wien.at

Abstract. *Two main goals for automated object manipulation processes are cost reduction and flexibility. Time-consuming, costly object-specific fixtures can be replaced by vision systems, whereby the manipulators are extended with cameras so that multiple objects in the environment can be precisely identified. To be able to manipulate an object, it must be recognized first in the world, and then the pose must be calculated. Neural network approaches recognize and estimate the pose of an object in a single step and yield superior results, but rely on vast amounts of training data. This work describes an approach for estimating the pose of identified objects without pre-trained pose data. Template matching is used to recognize objects in depth images, and the pose is estimated through principal component analysis (PCA). The input to the algorithm is reduced to the template. Pre-existing knowledge about the object further improves accuracy. A maximum deviation of 0.2 cm from the ground truth has been achieved, which suffices for the industrial grasping task. The system was evaluated with real measurements taken with an RGB-D camera. This work resembles a first step to estimate an object's pose with linear statistical methods.*

1. INTRODUCTION

Industrial robots are efficient at picking up objects in a predefined, structured environment [10]. When mobile manipulators are deployed in a factory setting and costly fixtures have to be avoided, robots need the ability to identify and locate objects for manipulation. To overcome this problem, a vision system can be used. One way to give robot vision is to use two-dimensional images with depth information, also known as 2.5D images or RGB-D images. RGB-D images can be used to find and localize objects by analyzing the environment. Building on top of the recognized and classified object, pose estimation tries to estimate the six degrees of freedom (DOF) pose of an object in an image. For mobile manipulation of objects this information is needed to accurately grasp objects with a manipulator in the correct position and orientation.

The current state of the art approaches towards object recognition and pose estimation are based on deep neural networks [15]. They usually outperform human crafted features [19], but unfortunately they rely on huge amounts of training data for classification and pose estimation and are difficult to adapt [9]. This is why, in this work, a more traditional approach was chosen. The target object is recognized using template matching in a 3D space. Pose estimation is implemented using the principal component analysis (PCA) to place an orthogonal basis in the center of the grabbing area. Using PCA to estimate the pose of the object, the needed input to the algorithm can be reduced to only the template. This work resembles a first step to estimate an object's pose with linear statistical methods.

In the following chapters the related work is summarized, the used methods are explained and the results are being discussed.

2. RELATED WORK

Object recognition describes the task of localizing known objects in images. Due to changes in the viewpoint or lighting, the task of mapping the huge amount of input pixels to a small output space is still complex [16]. To mitigate the influence of lighting conditions, approaches which rely on 3D information were researched [8]. The data used in these approaches is usually made up of a three channel 8-bit RBG image or an additional fourth channel which represents the 3D distance of the object to the image sensor, where each image is described using

$$I \in \mathbb{R}^{rows\ x\ cols\ x\ channels} \quad (1)$$

While research improved object recognition and classification with deep neural nets [15] , parallel efforts focused on template matching for object recognition [2][5]. Template matching uses extracted example images to find objects in new images. This method often involves sliding-window based algorithms [7], which find the template in a rectangular subpart of the image. Template matching works well for frontal images, but fails if the viewpoint differs from the actual template [4]. The simplicity of this technique still inspired new research, which is why its performance has improved significantly over the last 10 years [6][11].

2.1. Pose Estimation

Building on top of the recognized object, it is possible to estimate the pose of the object relative to the camera. This process is called pose estimation and it consists of three general categories. In the first category, the object's pose is stored alongside its feature vectors. Consequently, each different observed orientation represents a separate detection, which results in automatically knowing the objects pose if the object is matched with a previously trained one.

The second category uses statistical techniques to align two given RGB-D images with each other. For this, Iterative Closest Point [3], or ICP, is the most commonly used algorithm and many variants exist for different applications [12][14].

The third category tries to combine the pose estimation step with the recognition process itself. This makes sense, since, as stated earlier, a different viewpoint can change the appearance of an object entirely. This category has been covered by recent research due to the emerging field of machine learning [18].

Unfortunately, all of the before shown methods need either vast amounts of training data or an accurate model of the object that has to be detected. In this work, a different approach is taken. The principal component analysis (PCA) [1] is used for estimating the pose of a known object. PCA's intended purpose is to extract principal components and reduce dimensionality between the input and the output space. Using PCA to estimate the pose of the object, the needed input to the algorithm can be reduced to only the template. The proposed process of pose estimation with PCA is shown in the next chapter.

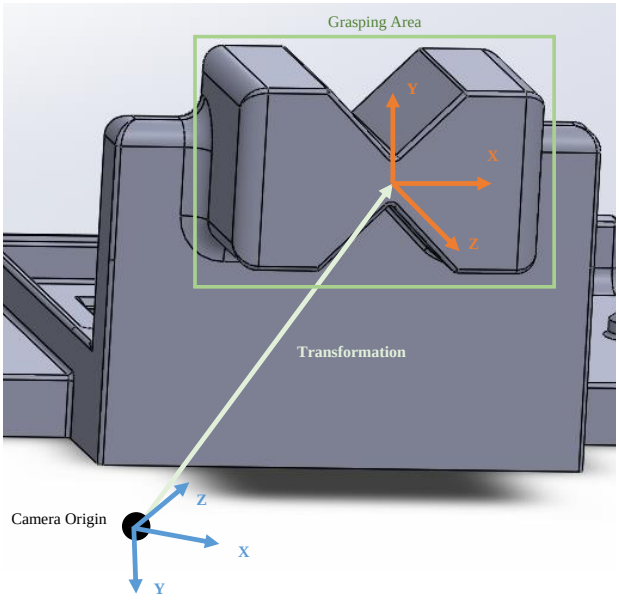

Figure 1. Visualization of the grasping point. The figure shows the object that has to be grasped. The orange coordinate system shows the center of the grasping area.

3. METHODS

The objective of the proposed approach is the estimation of the pose of a known object. Before the pose of an object can be calculated, it first has to be located in an image. For this task, template matching was chosen due to its ease of implementation and use. After the object has been recognized in the depth image, principal component analysis is used to determine the orientation of the found subpart of the image in 3D space.

Figure 1 shows the target object of this work. The pose of the shape in the "grabbing area" has to be calculated so that it can be successfully grasped. For this, the normal vector of the surface facing the camera has to be found. Through orientation of the vectors the rotational components of the 6D pose can be determined. This task can be solved by computing the PCA for the points in the grabbing area. In this case, the principal component analysis yields 3 eigenvectors with their respective eigenvalues for the given 3D points. As can be seen by studying Figure 1, 2 of the 3 dimensions of the shape in the grabbing area differ from the other. The span of values in the X and Y direction are comparatively large in respect to the depth dimension Z. This also applies to the respective variances. Using prior knowledge, the normal vector of the plane parallel to the camera origin (i.e. corresponding to the surface of the marked grabbing area) can be estimated using the eigenvec-

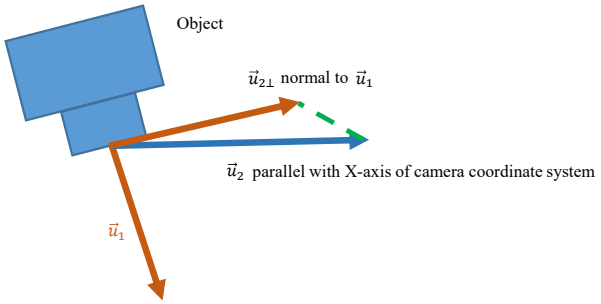

Figure 2. Visualization of the correct alignment of vector $\vec{u_2}$

tor with the smallest eigenvalue (eg. variance).

As the form of the shape is symmetrical, the mean of the points in the grabbing area estimates the origin of the coordinate system shown in Figure 1. Therefore, the mean of the PCA can be used as the translational component of the transformation matrix.

$$\vec{t} = (\vec{\mu_x},\ \vec{\mu_y},\ \vec{\mu_z})^T \tag{2}$$

The rotation matrix has to be assembled from three orthonormal vectors. The first vector has already been found, which is the smallest eigenvector of the PCA, which forms the Z vector pictured in Figure 1. The second vector can be obtained by leveraging knowledge about the environment of the industrial grasping use case. As the target object is located at a target location that is parallel to the ground, the rotation around the Z axis can be neglected. That is why the second vector can be aligned with the Y axis of the camera coordinate system. But since the first vector found with the PCA could be rotated around the Y axis of the object coordinate system, the second vector has to be projected orthogonally to the first. This is done with Equation (3) and the process is visualized in Figure 2.

$$\vec{u_{2||}} = (\vec{u_2^T} \cdot \vec{u_1}) \cdot \vec{u_1} \tag{3}$$
$$\vec{u_{2\perp}} = \vec{u_2} - \vec{u_{2||}}$$

The third vector can then be calculated using the cross product of $\vec{u_1}$ and $\vec{u_2}$. The resulting rotation matrix is constructed using Equation (4).

$$\mathbf{R} = \begin{pmatrix} u_{3,x} & u_{3,y} & u_{3,z} \\ u_{2,x} & u_{2,y} & u_{2,z} \\ u_{1,x} & u_{1,y} & u_{1,z} \end{pmatrix} \tag{4}$$

After calculating the rotation and translation components of the object, a transformation matrix can be formulated using Equation (5).

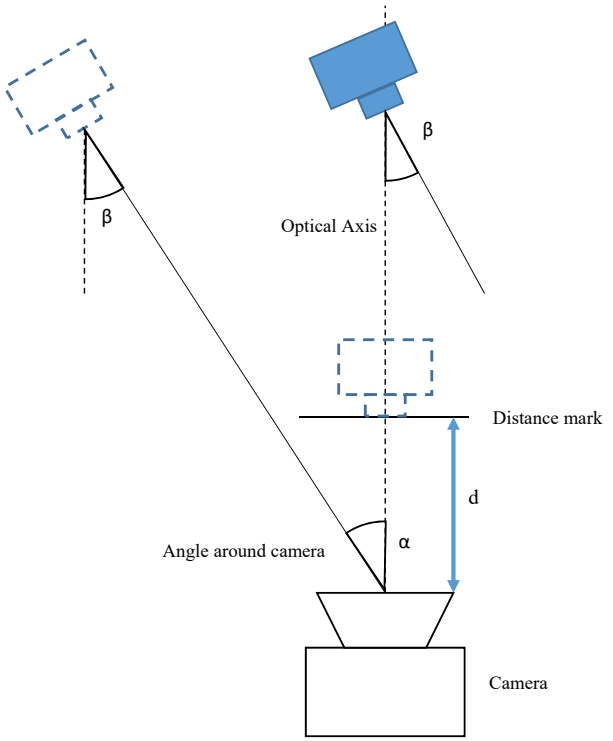

Figure 3. Visualization of the test setup

$$T_{obj}^{camera} = [\ \mathbf{R} \quad \vec{t}\ ] \tag{5}$$

The transformation matrix can then be used to express the grasping point in the world coordinate system, which is used for motion planning of the robot arm. Having calculated the transformation between the camera coordinate system and the objects coordinate system one can calculate the objects world position as follows

$$T_{obj} = T_{camera}^{world} \cdot T_{obj}^{camera} \tag{6}$$

In the following chapter, the performance of the proposed approach is discussed.

4. Results

The test setup consisted of the 3D printed model of the target object shown in Figure 1 and an Intel RealSense D435[1]. The RGB-D camera has been set up at a defined location on a table and the 3D printed model has been placed in front of it as can be seen in Figure 3.

To measure the error of the PCA-based approach, a metric had to be defined. For this, the Euclidean distance between the ground truth vector and the estimated plane normal vector of the PCA is used. Usu-

[1]https://www.intelrealsense.com/depth-camera-d435/

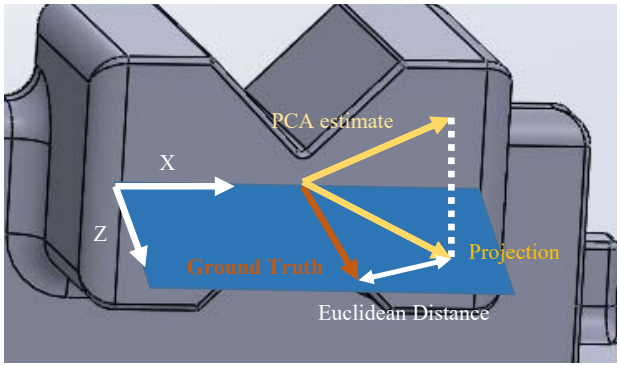

Figure 4. Visualization of calculation of the Euclidean distance between the ground truth vector and the vector estimated by the PCA

ally, with machine learning approaches, the error in the 2D projection of the 3D bounding box is measured [13]. Since the estimation error can be measured directly in this case, the Euclidean distance is used as a metric instead. To ease the calculation of the ground truth vector, environment knowledge has been used to eliminate one dimension out of the 3D vector. Since the target 3D model is guaranteed to always be parallel to the ground, as is the camera, the rotation around the Z-axis defined in Figure 1 can be ignored. Furthermore, as this approach is being used in an industrial grasping use case where the industrial robot has to grab the target object perpendicular to the estimated plane, the Y-component of the estimated PCA vector can be ignored and therefore set to 0. In order to get two vectors of the same length for further correct calculation, both, the ground truth vector and the vector estimated by the PCA have to be normalized. This results in an Euclidean distance being calculated between two vectors in the X-Z plane. This process is shown in Figure 4.

Equation (7) shows the calculation of the ground truth vector, where the $\vec{gt}$ vector is the ground truth. The X and Z components of the ground truth vector can be obtained by calculating the direction of the ground truth vector rotated by β depicted in Figure 3.

$$\vec{gt} = \begin{pmatrix} \sin(\beta) \\ 0 \\ \cos(\beta) \end{pmatrix} \tag{7}$$

Equation (8) shows the calculation of the error in form of the Euclidean distance.

$$r = \sqrt{(x_1 - x_2)^2 + (z_1 - z_2)^2} \tag{8}$$

x_1 and z_1 denote the respective components of the ground truth vector. x_2 and z_2 denote the respective components of the calculated normal vector by the PCA, that has been projected onto the X-Z plane.

Table 1. List of positions that were used for the experiments.

Angle [°]	Distance [cm]
+/- 0	30, 35, 40, 45, 50, 75
+/- 10	
+/- 20	
+/- 40	30, 35, 40, 45, 50
+/- 10 around camera	30, 35, 40, 45, 50, 75
+/- 20 around camera	
+/- 30 around camera	

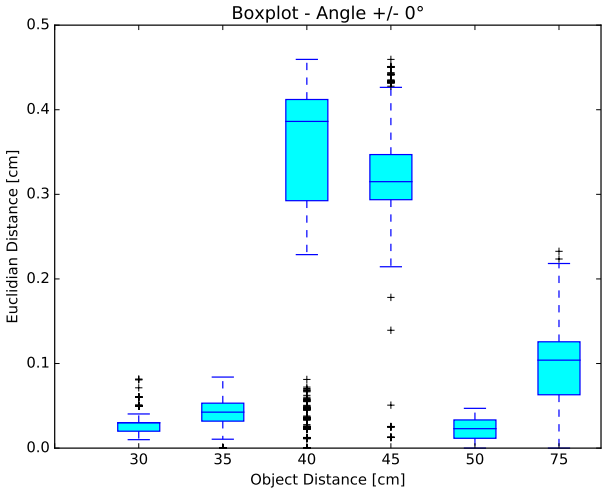

Figure 5. Visualization of the results with the object being placed on the optical axis

The measurements were taken in distances and orientations that relate to the industrial grasping use case. The target object has been moved to several fixed positions in front of the camera. Table 1 lists the positions that were used for the measurements.

Figure 5 shows the results for measurements taken with the object being placed on the optical axis.

Both of the anomalies at 40 and 45cm can be explained due to poorly selected templates. This can be mitigated by using advanced approaches for template matching [5][17]. Those rely on scaling of the template to get a more accurate match and also address the rotational limitations.

Figure 7 shows an example disparity image of the object viewed by the Intel RealSense camera. It can be argumented that the anomalies are induced because of the dark areas in the disparity image, which can be mostly traced back to occlusions of the stereo vision system. This has an even larger effect when

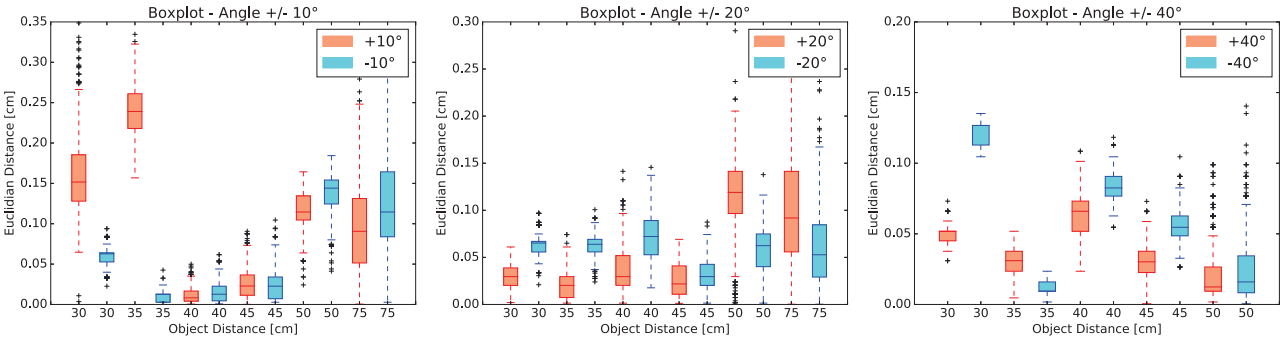

Figure 6. Visualization of the results with the object being placed at differing angles and distances on the optical axis

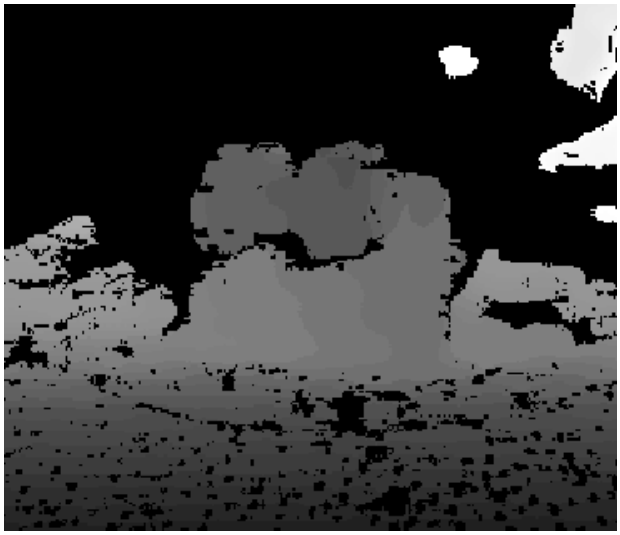

Figure 7. Disparity image of the object viewed by the Intel RealSense camera. Occlusions induced by the stereo vision system make it difficult to accurately locate the grasping area depicted in Figure 1.

the object is being rotated. Figure 7 also shows that it is difficult to depict the grabbing area of the object for creating a fitting template from the disparity image. Having a poorly chosen template leads to points being incorporated into the PCA estimation that are not actually part of the grabbing area and therefore lead to unexpected results. Nevertheless, it can be concluded that the anomalies are not induced by means of the method used for estimating the pose.

Figure 6 shows the results for measurements taken with the object being placed at different angles on the optical axis. Refer to Figure 3 for a visual presentation of this process. The angle depicted in Figure 6 corresponds to angle α shown in Figure 3. The distance mark relates to the distances shown on the X-axis labels of the graphs in Figure 6. The anomalies again can be explained by the problems mentioned before. The right graph in Figure 6 clearly shows the limits of the proposed approach, as the structure of the box plots over the graph changes in respect to the other two graphs.

Table 2. Regions in which the algorithm yields results sufficient enough for the industrial grasping use case at hand.

Angle [°]	Distance [cm]
+/- 0	30 - 75
+/- 10	
+/- 20	
+/- 10 around camera	30 - 75
+/- 20 around camera	
+/- 30 around camera	

Figure 8 shows the results for measurements taken with the object being placed at different angles around the camera. The angle depicted in Figure 8 corresponds to angle β shown in Figure 3. The anomalies again can be explained by the problems mentioned before.

The results show that the usable region for this algorithm can be summarized with Table 2. Argumenting, that the anomalies can be eliminated by using the enhancements already listed. Angles depicted with the "around camera" suffix correspond to the object being rotated around the camera with angle α, as depicted in Figure 3.

5. Conclusion

This work presented an approach to estimate the pose of a known object by using the principal component analysis. This resembles a first step to estimate an object's pose with linear statistical methods. The results showed that the approach is sufficient for the industrial use case at hand, since a maximum deviation of 0.2 cm compared to the ground truth is achieved, when anomalies are ignored. The results also show the limitations of this approach. Anoma-

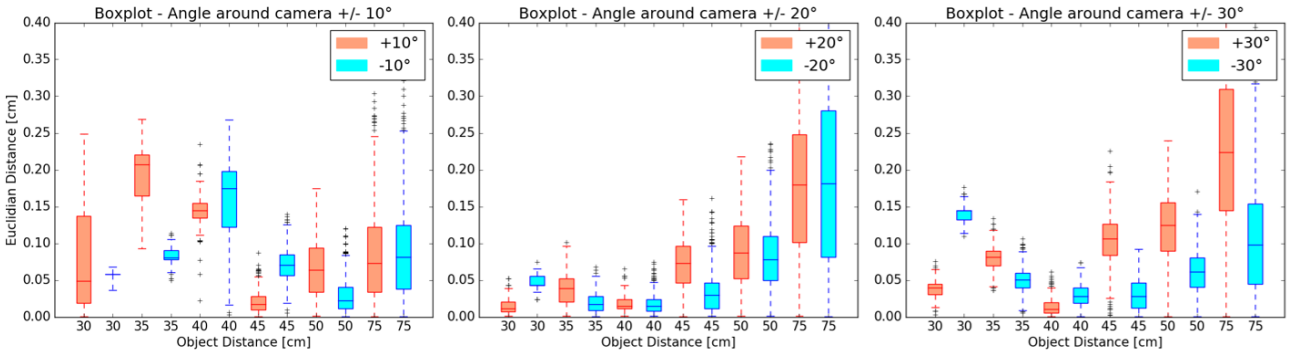

Figure 8. Visualization of the results with the object being placed at different angles and distances around the camera

lies shown in the data can be explained through poorly chosen templates. The problems faced could be solved in future work by using the recommendations given in this work.

References

[1] H. Abdi and L. J. Williams. Principal component analysis. *WIREs Computational Statistics*, 2(4):433–459, 2010.

[2] D. I. Barnea and H. F. Silverman. A Class of Algorithms for Fast Digital Image Registration. *IEEE Transactions on Computers*, C-21(2):179–186, Feb. 1972.

[3] P. Besl and N. D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, Feb. 1992.

[4] L. Cole, D. Austin, and L. Cole. Visual Object Recognition using Template Matching. In *Proceedings of Australian Conference on Robotics and Automation*, 2004.

[5] R. M. Dufour, E. L. Miller, and N. P. Galatsanos. Template matching based object recognition with unknown geometric parameters. *IEEE Transactions on Image Processing*, 11(12):1385–1396, 2002.

[6] S. Korman, D. Reichman, G. Tsur, and S. Avidan. Fast-Match: Fast Affine Template Matching. *International Journal of Computer Vision*, 121(1):111–125, Jan. 2017.

[7] K. Lai, L. Bo, X. Ren, and D. Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *2011 IEEE international conference on robotics and automation*, pages 1817–1824. IEEE, 2011.

[8] Y. Lu and D. Song. Robustness to lighting variations: An RGB-D indoor visual odometry using line segments. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 688–694. IEEE, 2015.

[9] G. Marcus. Deep learning: A critical appraisal. *arXiv preprint arXiv:1801.00631*, 2018.

[10] V. Nabat, M. de la O RODRIGUEZ, O. Company, S. Krut, and F. Pierrot. Par4: very high speed parallel robot for pick-and-place. In *2005 IEEE/RSJ International conference on intelligent robots and systems*, pages 553–558. IEEE, 2005.

[11] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi. A model-based 3d template matching technique for pose acquisition of an uncooperative space object. *Sensors*, 15(3):6360–6382, 2015.

[12] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat. Comparing ICP variants on real-world data sets. *Autonomous Robots*, 34(3):133–148, Apr. 2013.

[13] J. N. Rauer. Semi-Automatic Generation of Training Data for Neural Networks for 6d Pose Estimation and Robotic Grasping. Master's thesis, Fachhochschule Technikum Wien, Höchstädtplatz 5, 1200 Wien, 2019.

[14] S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. In *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pages 145–152, May 2001. ISSN: null.

[15] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In *Advances in neural information processing systems*, pages 2553–2561, 2013.

[16] B. S. Tjan and G. E. Legge. The viewpoint complexity of an object-recognition task. *Vision Research*, 38(15):2335–2350, Aug. 1998.

[17] F. Ullah and S. Kaneko. Using orientation codes for rotation-invariant template matching. *Pattern recognition*, 37(2):201–209, 2004.

[18] P. Wohlhart and V. Lepetit. Learning Descriptors for Object Recognition and 3d Pose Estimation. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Feb. 2015.

[19] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. PoseCNN: A Convolutional Neural Network for 6d Object Pose Estimation in Cluttered Scenes. *arXiv:1711.00199 [cs]*, May 2018. arXiv: 1711.00199.