 DOI: 10.3217/978-3-85125-603-1-19

Reliably Decoding Autoencoders' Latent Spaces for One-Class Learning Image Inspection Scenarios

Daniel Soukup[1] daniel.soukup@ait.ac.at Thomas Pinetz[1] thomas.pinetz@ait.ac.at

Abstract— **In industrial quality inspection, it is often the case that a lot of data of desired product appearance can be provided at training time, while very little erroneous examples are available. Thus, in order to train an inspection system, the target appearance has to be learned independently from the availability of defect samples. Defects have to be identified as anomalies w.r.t. the trained data distributions in the online inspection phase. In deep learning, autoencoders are a well known choice to realize anomaly detection scenarios, where significantly larger reconstruction errors of objects' images indicate defects. However, as the latent code contains enough information to reliably reconstruct good example images, the question arises if a decision about the validity of an input image can already be drawn in that latent space during online inspection. This would speed up the system by more than a factor of 2 by sparing the processing of the autoencoder's decoder part. Variational Autoencoders (VAE) are a modern variant of the classical autoencoder architecture, which could facilitate this purpose, because of its imposed regularization term, that forces the latent codes to be standard normally distributed.**

I. INTRODUCTION

In quality inspection of highly optimized industrial production processes, e.g. textile industries, a low rate of flaws is usually observed. This results in small and unrepresentative samples of defects. In such cases, inspection systems have to learn the valid product appearances only by means of valid product samples. Only later during actual inspection, some defects occur occasionally, which then have to be identified as anomalies or novelties w.r.t. the trained data distributions. Such a setting is referred to as *one-class learning*, *anomaly detection*, or *novelty detection* [8]. For image processing tasks, this kind of inspection is difficult for patterns that are on the one hand regular and contain repetitive structures on different scales, while on the other hand local variations and distortions are possible, which let each object region appear slightly different than its neighboring regions. An example of such a product type are textiles (Fig.1).

The majority of object regions are valid but vary slightly w.r.t. a trained area. An appropriate representation of the object structure under inspection should reflect a distinct deviation of defective areas, but at the same time it should be robust w.r.t. allowed distortions occurring due to inherent perturbations in the production processes.

A number of algorithms have been proposed to handle one-class learning problems. Very popular are one-class SVM [11], which separate the training data from the origin of the feature space using a hyperplane with maximum margin. SVMs can implicitly be applied to nonlinear and high-dimensional feature spaces. Sparse coding or convolutional sparse coding (CSC), respectively, was proposed to tackle the problem of novelty detection in images of nanofibrous material production [3] [4] by learning dictionaries to yield accurate and sparse representations.

We aim to process images of product parts, where the relevant, representative features have to be implicitly determined by the method itself in the course of the training procedure on multiple scales. Thus nowadays, Convolutional Neural Networks (CNN) are a reasonable choice. In deep learning, autoencoders [9] are a well known tool to perform unsupervised learning of object representations. They were extensively investigated and used for unsupervised pretraining, representation learning, data compression, etc. (e.g. [13]). Autoencoders are trained to reconstruct the input data as exactly as possible through a bottleneck layer of neurons, spanning the so called *latent space*. As a consequence, the autoencoder has to come up with internal representations of the trained patterns (e.g. images) that allow it to reconstruct input data only from those internal compressed vector codes, the so called *latent variables* or *latent codes*. They can be seen as a non-linear version of Principle Component Analysis (PCA) [2], because data are projected to an appropriate subspace in a non-linear manner, e.g. CNN layers, whereas the re-projection error is minimized. Alain and Bengio showed that autoencoders are capable of implicit recovery of the data generating density [1].

Consequently, autoencoders are a well-suited means to handle the image one-class learning task at hand. Cascades of convolutional layers (encoder) enable the identification of relevant pattern features on multiple scales, so that a characteristic, compressed latent representation can be obtained for trained good example images, that enables another cascade of transposed convolutional layers to decode that latent code into a reconstructed image. Good examples, similar to the trained patterns can be significantly better reconstructed than images comprising a deviation w.r.t. to the trained images, i.e. a defect. The autoencoder's decoders are capable of reconstructing input reliably only from the latent codes. Thus the entire structural information about an input image must already be coded in that latent variable. This raises the question, if the decoder part could be fully omitted in the online inspection phase after training is fully accomplished. The final decision about input validity could be solely drawn by evaluating the latent codes, which would spare more than half of the processing effort.

We investigate the opportunities of one-class learning with

[1]AIT Austrian Institute of Technology GmbH, Center for Vision, Automation & Control, Vienna, Austria

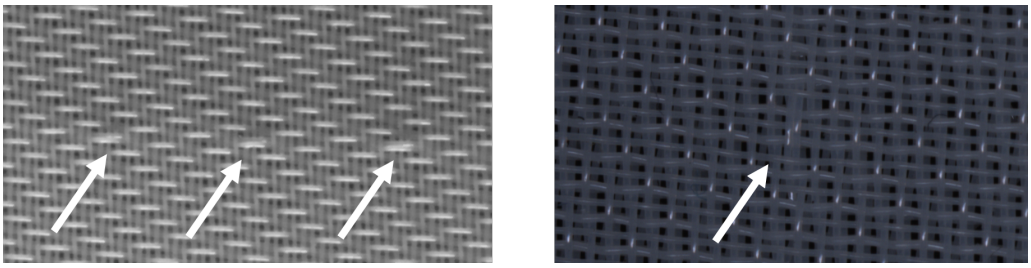

Fig. 1. Two cutouts from the investigated example web patterns with examples of weaving flaws indicated by white arrows. "Pattern 1" (left) and "Pattern 2" (right).

autoencoders, more specifically the reliability of drawing decisions already in the latent space for online applications of autoencoders in one-class learning for image anomaly detection tasks. In Section II, we describe autoencoders and the usage of the latent space in more detail. Results of experiments on two textile examples comprising weaving flaws are presented in Section III. We summarize and conclude in Section IV.

II. One-class learning with (Variational) Autoencoders

Autoencoders are an age-old concept in the area of neural networks (e.g. Rumelhart et al [9]). An autoencoder is a function $f : \mathbb{R}^n \to \mathbb{R}^n$, implemented as a neural network, that is trained to optimally reconstruct input data - images of good product appearance in our case - by minimizing the so-called *reconstruction loss function*

$$L_{rec} = \sum_i ||x_i - f(x_i)||_2^2. \tag{1}$$

f consists of an *encoder* part $c : \mathbb{R}^n \to \mathbb{R}^l$, where $l < n$, and a *decoder* part $d : \mathbb{R}^l \to \mathbb{R}^n$, such that $f = c \circ d$. As $l < n$, the autoencoder's weights have to adjust in such a way during the optimization process, that the l-dimensional latent codes contain all the information of the trained input patterns in order that the decoder d can reconstruct input images x_i of the training data distribution. Additionally, we make use of convolutional layers' expressive capabilities to extract representative image features. Thus we implement the encoder and the decoder as deep CNNs or tranposed CNNs, respectively.

In one-class learning, training data only contain examples of valid object appearance. When an image comprising a defect, e.g. weaving flaw, is presented to the fully trained autoencoder, the corresponding reconstruction error will be significantly larger than reconstruction errors of trained valid examples, by which the defect is detectable in the online phase.

However, as the total information about the training patterns and naturally deviations of which must already be mapped in the corresponding latent codes, there should be a way to already draw the decision from those latent codes. Such an early decision would speed up inspection processing in the online phase. Schlegl et al [10] managed to exactly implement this idea by means of Generative Adversarial Networks (GAN), which had been trained to generate artificial samples of the valid image data. Defects could be detected by the discriminator network, which was trained to detect anomalies w.r.t. to the training distribution. Makhazani et al [7] presented the Adversarial Autoencoder, a variant of a more classical autoencoder, where an adversarial network was trained to match the distribution of the latent codes with a predefined appropriate data distribution, e.g. standard normal distribution, by means of an added regularizing loss term to the reconstruction loss. Such a simple-shaped latent distribution would simplify the evaluation of latent codes in the one-class learning setting, because deviations from it can easily be detected.

The so-called Variational AutoEncoder (VAE) was introduced by Kingma et al [6]. Similarly to the Adversarial Autoencoder, the VAE is realized by adding a regularizing loss term to the reconstruction loss. This *latent loss* measures the dissimilarity of the latent codes' distribution to a predefined well-shaped target distribution, i.e. standard normal distribution, by means of the *Kullback-Leibler* divergence (KL):

$$\begin{aligned} L &= L_{rec} + \lambda \cdot L_{lat}, \quad with \\ L_{lat} &= KL(Q(z|X), \mathcal{N}(0,I)), \end{aligned} \tag{2}$$

where $Q(z|X)$ is the PDF of the distribution of latent codes $z_i = c(x_i) \in Z \subset \mathbb{R}^l$ given training examples $x_i \in X$. Usually, $Q(z|X) := \mathcal{N}(\mu(X;\theta), \Sigma(X;\theta))$, where μ and Σ are estimated by a neural network, in our case the encoder c. Although a regularization parameter λ is generally not required for VAE, in our application it was necessary in order to decrease the influence of the latent loss L_{lat}. Otherwise, L_{lat} dominates L_{rec} in the training process and the latent codes collapse to zero mean. That violates the main objective of optimal reconstruction, as those flattened latent codes do not have the expressive power to code pattern structures for reconstruction anymore. Doersch [5] provides a good tutorial over VAE, where he also discusses the requirement of regularization parameters for VAE.

In the online phase of the inspection with VAE, the decision, if an image contains a defect or is similar to the trained valid image distribution, could be made by means of the latent loss L_{lat}, which is at least in the average significantly larger for anomalies than valid data, just like for the reconstruction loss with classical autoencoders. We

analyze the applicability and reliability of decision making in one-class learning on the basis of latent VAE loss rather than the reconstruction loss in the next section on the basis of two textile examples.

III. Experiments

We present experiment results for two different web patterns (Fig. 1). The webs comprise regular structures, but also swirling local variations, which are typical for textiles. The autoencoders have to capture the distribution of allowed pattern variation from a set of sampled patches, where no weaving defects occur (*training set*). From another region of valid product appearance, patches were sampled which are not used in training, but only for validation (*validation set*). Around defect regions, i.e. weaving flaws, we extracted randomly distributed patches containing those defect patterns (*defect set*). In the setting of one-class learning, those were naturally also not used for training, as they are assumed to be not available in sufficient amounts for training in real scenarios. The size of all patches was fixed to 64×64 pixels, a field of view (FOV) where regular structures and disruptions of which are apparent.

According to the size of image patches, the input size of the autoencoder has a FOV of 64×64 as well. The autoencoder architecture is a U-shaped CNN with a bottleneck in the middle, yielding the latent codes $c(X)$, which is inspired by architectural elements from VGG[12] architectures, where only small filters are used, e.g. 3×3 to stick with the VGG scheme. In the encoder part c, the resolution of feature maps is decreased at every convolutional layer by strides of 2, while the number of feature maps is increased by a factor of 2. The decoder d, which generates the reconstruction of the input images from the latent codes, is structured analogously, only in a transposed manner, i.e. the number of feature maps is reduced and resolution of feature maps is increased. For all convolutional layers and all except the last transposed convolutional layers, the ReLU non-linearity was chosen. The last transposed convolutional layer is complemented by a *tanh* non-linearity.

The search for an optimal regularization parameter λ, balancing the influences of reconstruction and latent losses L_{rec} and L_{lat}, respectively, was conducted by repeating the training process with different values of λ and choosing the one, for which the reconstruction loss and the latent loss deviate most significantly between valid and the few available defective patches. For both experiment patterns, $\lambda = 10^{-5}$ was optimal. For a real scenario, where absolutely no defective examples are available in the training phase, this λ search is not applicable.

The appropriate learning rate was 10^{-3} and all autoenoders were trained for 10000 iterations. The training curves appeared to be very smooth and precisely reproducible between different runs with varying random training sets.

In Figs. 2 and 3, we visualize the distributions of reconstruction losses and latent losses of individual training (blue), validation (green), and defect (red) patches after the corresponding autoencoders were fully trained. We have augmented those histograms with Gaussian approximations in order to emphasize the gross distribution structures. Moreover, we computed the cross-entropy distances $H(p,q)$ between the training, validation, and defect distributions, respectively, according to

$$H(p,q) = -\sum_i p_i \cdot log_2(q_i), \tag{3}$$

which measures the dissimilarity between two distributions p and q (Tab. I).

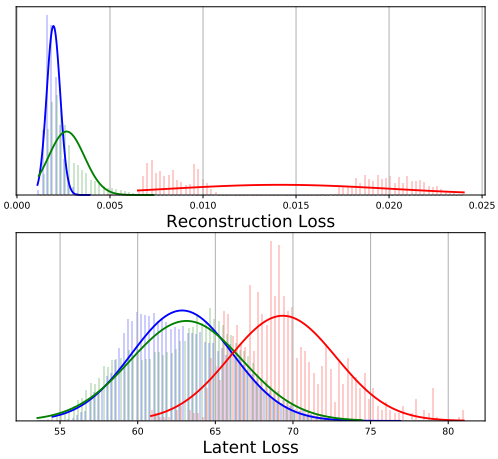

Fig. 2. Distributions of reconstruction loss (top) and latent loss (bottom) of individual patches sampled from Pattern 1. Training patches (blue), validation patches (green), defect patches (red). Histogram distributions of loss values augmented with Gaussian approximations.

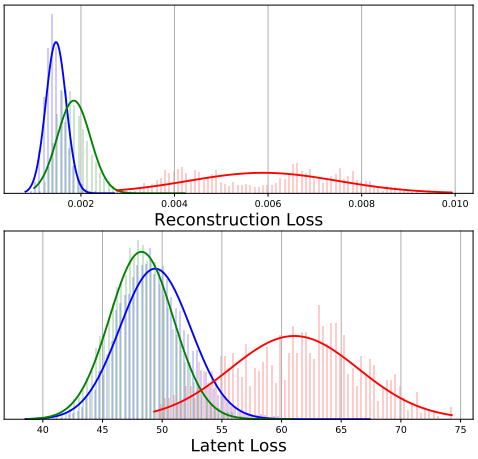

Fig. 3. Distributions of reconstruction loss (top) and latent loss (bottom) of individual patches sampled from Pattern 2. Training patches (blue), validation patches (green), defect patches (red). Histogram distributions of loss values augmented with Gaussian approximations.

For both patterns, from Figs. 2 and 3 as well as from Tab. I, it is apparent that the training and validation histograms are

TABLE I

CROSS-ENTROPY DISTANCES (EQU. 3) BETWEEN DEPICTED LOSS DISTRIBUTIONS OF TRAINING (T), VALIDATION (V), AND DEFECT (D) DATA SETS (FIGS. 2, 3) INDICATING THE DISSIMILARITY OF LOSS VALUE DISTRIBUTIONS BETWEEN DIFFERENT DATA SETS T, V, AND D, RESPECTIVELY.

Cross entropy $H(p,q)$	H(T,V)	H(T,D)	H(V,D)
Pattern 1 - Reconstruction Loss	0.67	20.35	20.30
Pattern 1 - Latent Loss	0.10	7.64	6.95
Pattern 2 - Reconstruction Loss	1.15	19.70	19.27
Pattern 2 - Latent Loss	0.17	6.98	9.38

more similar for the latent loss. Reconstruction losses of validation patches tend to be slightly larger than those of the training patches. However, that is not so disconcerting, as the autoencoders are explicitly optimized to tightly fit the training distributions, mainly by optimizing the reconstruction error. Thus a minor increase of reconstruction errors for not trained valid samples is to be expected and acceptable. More important is that the defect distributions are distinctly deviating from both the training and validation distributions in order to make defect detection feasible at all. While the defect distributions according to the latent loss are overlapping with both valid distributions, defects obviously comprise consistently, significantly larger reconstruction losses than the valid examples. This makes defect detection on the basis of reconstruction loss more reliable. Both observations are also confirmed by the cross-entropy distances in Tab. I.

IV. CONCLUSIONS

In industrial image inspection tasks, one-class learning is a common scenario, where target product appearances have to be learned solely on the basis of valid product examples, because examples of defects are not available in sufficient amounts for training. Autoencoders are a well investigated means in deep learning for learning data distributions in an unsupervised manner. Thus they are appropriate methods for one-class learning, where they are trained to reconstruct input training images through a bottleneck layer of neurons as precisely as possible. In online inspection, input defects result in measurably larger reconstruction errors than valid examples by which they are identifiable. We investigated the opportunities to speed up that process by drawing the inspection decision already from the outputs of that bottleneck layer, the latent codes. In order to simplify the structure of distributions of valid latent codes and therefore the decision making procedure, we applied VAE regularization, where the latent codes are forced to possibly follow a standard normal distribution by means of an added regularization term.

Our experiments with two textile examples show, that in the average, defective images actually comprise larger latent errors by which they could be identified over valid patches. However, an analysis of the distributions of reconstruction errors and latent errors over individual patches, the results indicated that defect images are not as reliably distinguishable form valid images on the basis of latent VAE codes than on the basis of reconstruction errors. In a real inspection task, it would be difficult to set a threshold, which serves as decision boundary between valid and defect images on the basis of latent codes. While drawing the decision from latent codes would speed up computations in the online phase by more than a factor of 2, it seems to be insufficiently reliable. Either more false positives or overseen defects are the consequence. Probably because the reconstruction loss is the main workhorse of autoencoder training and it is explicitly optimized to extract latent codes to images, the decoder is the distinctly smarter tool for analyzing latent codes in one-class learning. If reliability counts, then it is better to invest the computational effort and go for the reconstruction loss as the decision measure. In addition, the training procedure becomes simpler, because the search for an appropriate regularization parameter steering the influence of the latent codes' distribution can be omitted.

REFERENCES

[1] G. Alain and Y. Bengio, "What regularized auto-encoders learn from the data-generating distribution," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3563–3593, 2014. [Online]. Available: http://dl.acm.org/citation.cfm?id=2750359

[2] P. Baldi and K. Hornik, "Neural networks and principal component analysis: Learning from examples without local minima," *Neural Netw.*, vol. 2, no. 1, pp. 53–58, Jan. 1989.

[3] G. Boracchi, D. Carrera, and B. Wohlberg, "Novelty detection in images by sparse representations," in *Proceedings of the IEEE Symposium Series on Computational Intelligence (IEEE SSCI)*, Orlando, FL, USA, Dec. 2014, pp. 47–54.

[4] D. Carrera, G. Boracchi, A. Foi, and B. Wohlberg, "Detecting anomalous structures by convolutional sparse models." in *IJCNN*. IEEE, 2015, pp. 1–8.

[5] C. Doersch, "Tutorial on variational autoencoders," 2016, cite arxiv:1606.05908.

[6] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *CoRR*, vol. abs/1312.6114, 2013. [Online]. Available: http://arxiv.org/abs/1312.6114

[7] A. Makhzani, J. Shlens, N. Jaitly, and I. Goodfellow, "Adversarial autoencoders," in *International Conference on Learning Representations*, 2016. [Online]. Available: http://arxiv.org/abs/1511.05644

[8] M. A. F. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "Review: A review of novelty detection," *Signal Process.*, vol. 99, pp. 215–249, June 2014. [Online]. Available: http://dx.doi.org/10.1016/j.sigpro.2013.12.026

[9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1," D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds. Cambridge, MA, USA: MIT Press, 1986, ch. Learning Internal Representations by Error Propagation, pp. 318–362.

[10] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Information Processing in Medical Imaging - 25th International Conference, IPMI 2017, Boone, NC, USA, June 25-30, 2017, Proceedings*, 2017, pp. 146–157.

[11] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Proc. of Advances in Neural Information Processing Systems (NIPS)*, 1999, pp. 582–588.

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[13] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML '08. New York, NY, USA: ACM, 2008, pp. 1096–1103. [Online]. Available: http://doi.acm.org/10.1145/1390156.1390294