A TWO-STAGE AUTOENCODER FOR VISUAL ANOMALY DETECTION

Yezhou Zhu, Jianzhu Wang, Jing Zhang, Qingyong Li

Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing, China

2021 IEEE International Conference on Image Processing (ICIP) | 978-1-6654-4115-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICIP42928.2021.9506538

ABSTRACT

Deep convolutional autoencoder (DCAE) is usually optimized to minimize the difference between the input and the reconstruction, and the reconstruction error has been widely used as an indicator for visual anomaly detection. However, DCAE sometimes can reconstruct anomalies very well and thus may yield misdetections. To tackle this issue, we propose a novel non-symmetrical DCAE, which is trained in a two-stage manner. Specifically, a single RotNet is first trained to serve as encoder. Then, discriminative representations generated by the frozen encoder are used to train two parallel decoders for image reconstruction. Finally, the reconstruction errors obtained by the two decoders are combined as the anomaly score. Massive experiments on three public datasets and one practical industrial dataset demonstrate the superiority of the proposed method among existing reconstruction based methods.

***Index Terms*—** Autoencoder, RotNet, Anomaly Detection

1. INTRODUCTION

Visual anomaly detection aims to identify samples that do not conform to expected behaviors [1], and it has found wide applications in practice, such as medical marker discovery [2], surface defect inspection [3] and video event detection [4]. In the field of machine learning and computer vision, anomaly detection has often been formulated as an unsupervised or semi-supervised problem because of the rareness of anomalous samples. That is, in many scenarios, only few or none anomalous samples are available for model training. In recent years, with the development of deep neural networks, many anomaly detection methods [5, 6, 7, 8] have been proposed, and they can be broadly classified into classification based methods (CBMs) and reconstruction based methods (RBMs).

Considering that normal and abnormal samples should have different representations, CBMs try to find the boundary that can separate normal and abnormal samples in the latent space, and some effective methods have been proposed based on unsupervised learning. Typically, Ruff *et al.* [5] train a neural network to minimize the volume of a hypersphere such that the representations can be enclosed. To make the learned representations more discriminative, Hendrycks *et al.* [6] train the model with the help of auxiliary tasks, whose probability distributions are then used as a criterion for anomaly detection.

Fig. 1. From top to bottom: input images, reconstructions of DCAE and the proposed method and the heat map, in which a deeper color means a lower relative reconstruction error compared with images on the same row. Here, images of digit 8 are normal and used for model training, while images of other digits are used as anomalies.

Different from CBMs, RBMs assume that models trained on normal samples have a better reconstruction capability for normal samples, making reconstruction errors of normal samples lower than that of anomalies. It is worth mentioning that DCAE plays an important role in these methods. However, as illustrated in Figure 1, a DCAE trained on normal samples (i.e., images of digit 8) can still restore some unseen anomalous images (i.e., images of other digits) well. In order to avoid the problem, Akcay *et al.* [7] additionally integrate a discriminator sub-network and an encoder sub-network with a DCAE, all of which can provide a better metric for anomaly detection. Perera *et al.* [8] try to restrict the latent space to follow a uniform distribution. Therefore, the model is able to restore anomalous images as normal ones, which can enlarge the reconstruction errors and thus anomalies can be identified.

It can be found that most DCAE models are trained in an end-to-end manner. In other words, these models solely try to reconstruct an input after it passes through a low-dimensional bottleneck layer. As reported in [9], this process is likely to just compress the image content without learning a semantically meaningful representation. Especially, if there exist large variations among normal images, it will be hard for these models to map them to a compact latent space. Coincidentally, it has been proved that RotNet [10] can effectively achieve semantic feature learning. Thus, one feasible way

to avoid a DCAE learning trivial representations is to assign RotNet to take the role of the encoder. Based on the above analysis, we propose a two-stage non-symmetrical DCAE for visual anomaly detection. In our proposed architecture, a single RotNet [10] is first trained on normal images to serve as the encoder. Then, we freeze the encoder and feed the resulting representations to two decoders, which are trained with two different loss functions for image reconstruction. As shown in Figure 1, our method can effectively enlarge the relative errors between the normal and anomalous images. The main contributions of our work are summarized below:

- To the best of our knowledge, we are the first to use RotNet to build a DCAE architecture for anomaly detection.
- We use the mean square error (MSE) and the structural similarity (SSIM) [11] to optimize two decoders, which are expected to provide both global and local views for anomaly determination.

2. PRELIMINARIES

2.1. RotNet

RotNet [10] is first introduced only for self-supervised representation learning. To be specific, it trains a deep convolutional neural network model $F(\cdot)$ to predict the transformation applied to an image. Formally, a set of K discrete geometric transformations $H = \{h(\cdot|t)\}_{t=1}^{K}$ is firstly defined, where $h(\cdot|t)$ is the operator applied to images and t denotes the transformation label. Therefore, the transformed version of a given image x can be formulated as $x^t = h(x|t)$. For an input image x^{t*} (where $t*$ is unknown to $F(\cdot)$), $F(\cdot)$ yields a probability distribution over all possible geometric transformations:

$$F(x^{t*}|\theta) = \left\{F^t(x^{t*}|\theta)\right\}_{t=1}^{K} \tag{1}$$

where $F^t(x^{t*}|\theta)$ is the predicted probability for the geometric transformation with label t and θ are the learnable parameters of model $F(\cdot)$. The loss function L_r is defined as:

$$L_r = -\frac{1}{K}\sum_{t=1}^{K} log(F^t(h(x|t)|\theta)) \tag{2}$$

Specifically, the 0, 90, 180, and 270 degree rotations are adopted as all transformations in [10]. The motivation behind RotNet is intuitive. If one model can predict the rotation applied on an image, it should first understand what content is contained in that image. This is the core that RotNet can successfully learn semantic representations. As discussed in [6], RotNet can also be used to detect anomalies. Given a test image x, one can apply all K transformations to get a set of images, then take the sum of probability assigned to certain transformation t as the anomaly score, but the performance of such approaches is heavily dependent on the interaction between transformations and the dataset [12].

2.2. Structural Similarity

Under the assumption that human visual perception is highly adapted for extracting structural information, SSIM is initially proposed to quantify the visibility of errors between a distorted image and a reference image by simultaneously considering the luminance, contrast and structure. Given a pair of images y and $\hat{y}$, SSIM between them can be simply computed by:

$$SSIM(y,\hat{y}) = \frac{(2\mu_y\mu_{\hat{y}} + c_1)(2\sigma_{y\hat{y}} + c_2)}{(\mu_y^2 + \mu_{\hat{y}}^2 + c_1)(\sigma_y^2 + \sigma_{\hat{y}}^2 + c_2)} \tag{3}$$

where μ_y and $\mu_{\hat{y}}$ denote the mean intensity of y and $\hat{y}$, respectively. Similarly, σ_y and $\sigma_{\hat{y}}$ represent the standard deviation of y and $\hat{y}$, and $\sigma_{y\hat{y}}$ signifies the covariance between y and $\hat{y}$. c_1 and c_2 are two variables to stabilize the division.

3. PROPOSED METHOD

3.1. Network Architecture

As shown in Figure 2, the designed network is a two-stage encoder-decoder architecture. Different from previous models that have pairwise encoder-decoder layers, the depth and network structure of our encoder and decoders are totally different, i.e., non-symmetrical. More importantly, we do not train the network in an end-to-end manner. On the contrary, we first train a RotNet to serve as the encoder for discriminative representation learning. Freezing the encoder, we then use the resulting representations to train two decoders, which are optimized with two different loss functions. More details are elaborated in the subsequent section.

3.2. Training

Formally, we use X and Z to denote the distribution of input images and corresponding representations. Then, the encoding process can be formulated as:

$$F(\cdot|\theta) : X \rightarrow Z \tag{4}$$

where F signifies the RotNet-based encoder parameterized by θ. Detailed training process of the encoder can be referred from Section 2.1. Given an input image x sampled from X, it can be encoded as:

$$z = F(x|\theta) \tag{5}$$

where z is the learned representation of x. Instead of simply using one decoder, we have two decoder branches in our proposed method. Therefore, the decoding process can be described as:

$$G_i(\cdot|\varphi_i) : Z \rightarrow \hat{X}_i, i \in \{1,2\} \tag{6}$$

where G_i is the i-th decoder parameterized by φ_i, and $\hat{X}_i$ denotes the distribution of the images reconstructed by G_i.

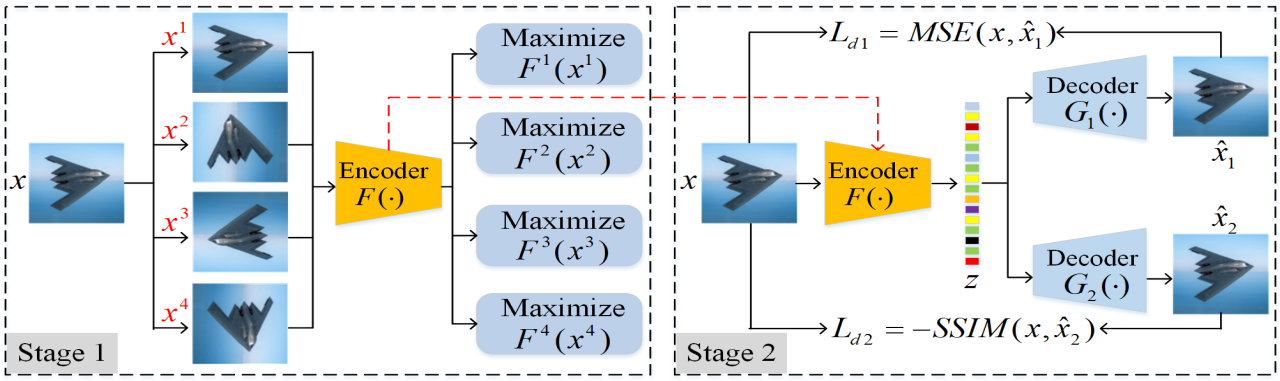


Fig. 2. Illustration of the proposed two-stage non-symmetrical autoencoder architecture. Stage 1: Train a RotNet as the encoder. Stage 2: Freeze the encoder and use the resulting representations to train two decoders with two different loss functions.

With the learned representation z, it can be decoded as:

$$\hat{x}_i = G_i(z|\varphi_i), i \in \{1, 2\} \tag{7}$$

where $\hat{x}_i$ represents the reconstruction obtained by the i-th decoder. As for the first decoder, the loss function is defined as:

$$L_{d1} = MSE(x, \hat{x}_1) \tag{8}$$

where MSE globally computes the mean square error between x and $\hat{x}_1$. To locally measure the similarity between the input and the reconstruction, the second decoder is restricted by a structural loss function. Considering that the more similar the input and the reconstruction are, the lower the loss value should be. Therefore, the loss function here is defined as the negative form of SSIM:

$$L_{d2} = -SSIM(x, \hat{x}_2) \tag{9}$$

where $SSIM$ is defined in Equa.(3). It should be noted that two decoders are trained with the same representations, but they are parallel and not involved in the training process of each other.

3.3. Testing

In general, the larger the difference between the input and the reconstruction is, the more likely the sample is an anomaly. Therefore, the anomaly score is defined as:

$$AS(x) = MSE(x, \hat{x}_1) - \lambda SSIM(x, \hat{x}_2) \tag{10}$$

where λ is a tradeoff parameter to balance the importance of global and local reconstruction differences. In our experiment, we enumerate λ and find that simply setting λ to 1 can achieve a satisfactory overall performance.

4. EXPERIMENT

4.1. Datasets and Settings

To evaluate the performance of the proposed method, three public datasets and one practical industrial dataset are adopted and the details of these datasets are as follows.

- **MNIST** [13]: It contains 28×28 grayscale images of handwritten digits from 0 to 9. There are totally 60,000 images for training and 10,000 images for testing.
- **FMNIST** [14]: It can be regarded a new version of MNIST. This dataset contains the same number of images with the same size and same partition for training and testing, but the digits are replaced with 10 categories of fashion apparels.
- **CIFAR10** [15]: It consists of 32×32 RGB images from 10 categories. There are totally 50,000 images for training and 10,000 images for testing.
- **Fastener**: With the help of a track inspection vehicle, we collect a set of fastener images. In this paper, a total of 22,000 normal fastener images are used for training, and 17,000 images including 9,000 anomalous images are used for testing.

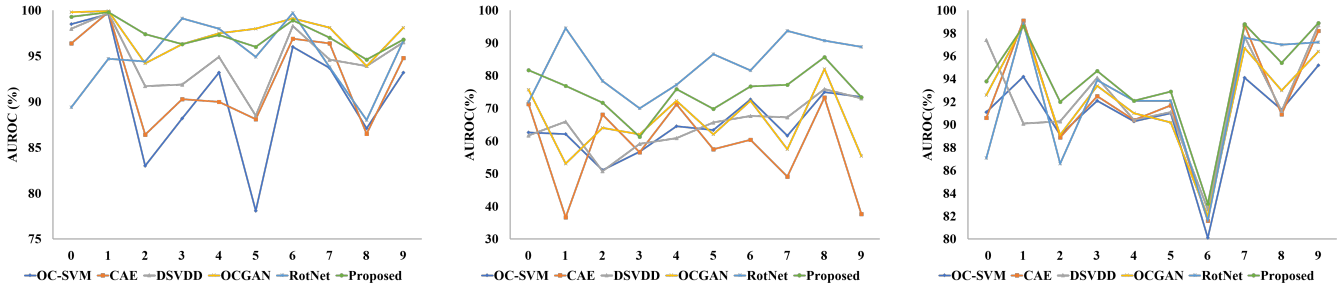
For the three public datasets, we use their default partitions for model training and testing. Following the protocol used in [5, 8, 16], we choose one class of images as normal samples in turn, and images from other classes are considered as the anomalous samples.

4.2. Evaluation Metric

We use area under the receiver operating characteristic curve (AUROC) [17] to measure the performance. The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. For anomaly detection task, TPR and FPR are the percentages of images that are correctly and wrongly classified as anomalies, respectively. By definition, a higher score of AUROC means a better performance.

4.3. Implementation Details

We adopt ResNet18 [18] as the backbone of RotNet, i.e., the encoder. As for the two decoders, they share a same structure, and details can be found in [7]. In terms of optimiza-



(a) MNIST (b) CIFAR10 (c) FMNIST

Fig. 3. AUROC values of specific class of images being normal samples.

tion, SGD optimizer with an initial learning rate $lr_1 = 0.1$ is adopted for the encoder, and Adam [19] optimizer with a learning rate $lr_2 = 0.0002$ is employed for two decoders. All of the encoder and two decoders are trained on a single NVIDIA TITAN Xp graphics card with 12GB memory.

4.4. Comparison

We compare the proposed method with several approaches, including OC-SVM [20], CAE [21], DSVDD [5], OCGAN [8], and a method based on RotNet [6]. The overall results on the four datasets can be found in Table 1, and the detailed results on MNIST, CIFAR10 and FMNIST can be observed in Figure 3. Note that results on Fastener dataset are not plotted on Figure 3, since only two classes of images are contained in that dataset.

As shown in Table 1, in MNIST dataset, the proposed method is comparable with OCGAN and outperforms all other methods. Especially, the proposed model shares a similar structure with CAE, but achieves an average AUROC of 97.3% and exceeds CAE for nearly 5% in average. Similar results can also be observed on other datasets, that is, the proposed method always beats CAE by a significant margin, which demonstrates the effectiveness of our designed two-stage non-symmetrical structure. In CIFAR10 dataset, our method achieves an average AUROC of 75.0% and outperforms all other RBMs. In addition, our proposed method outperforms all baseline methods on both FMNIST and Fastener datasets.

However, we can find that there is a clear performance gap between our proposed method and RotNet on CIFAR10 dataset. One possible explanation is that images in CIFAR10 dataset have various backgrounds, which may be less important for rotation prediction. Therefore, RotNet may ignore such information in the process of semantic representation learning. That is, the resulting representations may not contain enough information for image reconstruction. As a result, there may exist overfitting in the training process of two decoders. Nevertheless, our method outperforms RotNet on all other datasets. More precisely, our method increases average AUROC values by 2.4%, 1.6% and 5.3% on MNIST, FMNIST and Fastener, respectively.

Table 1. Average AUROC (%) of all classes for anomaly detection on all datasets. Numbers in bold indicate the best results.

Datasets	MNIST	CIFAR10	FMNIST	Fastener
OC-SVM [20]	91.1	64.3	90.9	87.4
CAE [21]	92.6	58.2	92.3	90.0
DSVDD [5]	94.8	64.8	92.4	91.7
OCGAN [8]	**97.5**	65.6	92.3	91.0
RotNet [6]	94.9	**83.3**	92.4	93.4
Proposed	97.3	75.0	**94.0**	**98.7**

5. CONCLUSION

In this paper, we propose a novel autoencoder architecture model for visual anomaly detection. In our proposed method, the encoder and two decoders are non-symmetrical and trained in a two-stage manner. We adopt a RotNet as the encoder, which is able to learn meaningful representations of normal samples. In addition, two decoders trained with two different metrics enable us to evaluate the reconstruction errors more comprehensively. Massive experiments on four datasets demonstrate the effectiveness of the proposed method. However, for images with dynamic backgrounds, the performance of our method remains to be improved, and how to solve this will be our future work.

6. ACKNOWLEDGEMENTS

This work is supported in part by the National Natural Science Foundation of China under Grant U2034211, 62006017, in part by the Fundamental Research Funds for the Central Universities under Grant 2020JBZD010, in part by the Beijing Natural Science Foundation under Grant L191016 and in part by the China railway R&D Program under Grand P2020T001.

7. REFERENCES

[1] Varun Chandola, Arindam Banerjee, and Vipin Kumar, "Anomaly detection: A survey," *ACM Computing Surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[2] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *International Conference on Information Processing in Medical Imaging*. Springer, 2017, pp. 146–157.

[3] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger, "MVTec AD – A comprehensive real-world dataset for unsupervised anomaly detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9592–9600.

[4] Weixin Luo, Wen Liu, and Shenghua Gao, "A revisit of sparse coding based anomaly detection in stacked RNN framework," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 341–349.

[5] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft, "Deep one-class classification," in *International Conference on Machine Learning*, 2018, pp. 4393–4402.

[6] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song, "Using self-supervised learning can improve model robustness and uncertainty," in *Advances in Neural Information Processing Systems*, 2019, pp. 15663–15674.

[7] Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon, "GANomaly: Semi-supervised anomaly detection via adversarial training," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 622–637.

[8] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang, "OCGAN: One-class novelty detection using GANs with constrained latent representations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2898–2906.

[9] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2536–2544.

[10] Spyros Gidaris, Praveer Singh, and Nikos Komodakis, "Unsupervised representation learning by predicting image rotations," in *International Conference on Learning Representations*, 2018.

[11] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[12] Sachin Goyal, Aditi Raghunathan, Moksh Jain, Harsha Vardhan Simhadri, and Prateek Jain, "DROCC: Deep robust one-class classification," *arXiv preprint arXiv:2002.12718*, 2020.

[13] Yann LeCun, "The MNIST database of handwritten digits," *http://yann. lecun. com/exdb/mnist/*, 1998.

[14] Han Xiao, Kashif Rasul, and Roland Vollgraf, "Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[15] Alex Krizhevsky, Geoffrey Hinton, et al., "Learning multiple layers of features from tiny images," 2009.

[16] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel, "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1705–1714.

[17] Jesse Davis and Mark Goadrich, "The relationship between Precision-Recall and ROC curves," in *International Conference on Machine Learning*, 2006, pp. 233–240.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 630–645.

[19] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[20] Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[21] Jonathan Masci, Ueli Meier, Dan Cireşan, and Jürgen Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *International Conference on Artificial Neural Networks*. Springer, 2011, pp. 52–59.