IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 11, 2020, accepted December 5, 2020, date of publication December 9, 2020,
date of current version December 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3043473

Unsupervised Anomaly Detection Using Style Distillation

HWEHEE CHUNG[1,*], JONGHO PARK[1,*], JONGSOO KEUM[1,*], HONGDO KI[1], AND SEOKHO KANG[2]
[1]Cognex Deep Learning Lab, Seoul 06655, South Korea
[2]Department of Industrial Engineering, Sungkyunkwan University, Suwon 16419, South Korea

Corresponding authors: Hongdo Ki (hongdo.ki@cognex.com) and Seokho Kang (s.kang@skku.edu)

(**Hwehee Chung, Jongho Park, and Jongsoo Keum contributed equally to this work.*)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea Government, Ministry of Science and ICT (MSIT), under Grant NRF-2019R1A4A1024732 and Grant NRF-2020R1C1C1003232.

ABSTRACT Autoencoders (AEs) have been widely used for unsupervised anomaly detection. They learn from normal samples such that they produce high reconstruction errors for anomalous samples. However, AEs can exhibit the over-detection issue because they imperfectly reconstruct not only anomalous samples but also normal ones. To address this issue, we introduce an outlier-exposed style distillation network (OE-SDN) that mimics the mild distortions caused by an AE, which are termed as style translation. We use the difference between the outputs of the OE-SDN and AE as an alternative anomaly score. Experiments on anomaly classification and segmentation tasks show that the performance of our method is superior to existing methods.

INDEX TERMS Anomaly detection, style distillation, autoencoder, knowledge distillation, outlier exposure.

I. INTRODUCTION

The objective of unsupervised anomaly detection is to identify anomalous samples from data. Unsupervised anomaly detection assumes that only normal samples are present while anomalous samples are absent in the training dataset. This formulation is useful when it is difficult to collect sufficient anomalous samples in advance or to obtain all possible anomaly patterns. Real-world examples of such scenarios include video surveillance [1], medical diagnosis [2], equipment failure detection [3], and manufacturing inspection [4].

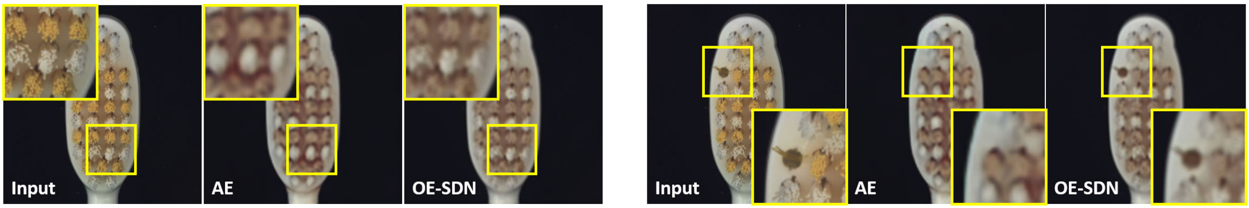
There have been many research attempts to investigate unsupervised anomaly detection using deep neural networks. Among them, reconstruction-based anomaly detection [5]–[7] using autoencoders (AEs) is an intuitive and promising method to detect anomalies in the image domain. An AE is trained on normal samples to reconstruct them through a bottleneck layer, so that an anomalous sample is extremely distorted whereas a normal sample is not. For a new sample, the difference between the input and its reconstruction is used as the anomaly score. A sample with a score higher than a predefined threshold is rejected as an anomaly.

However, owing to the bottleneck architecture of an AE, even the reconstructions of normal samples are mildly distorted. The performance of reconstruction-based anomaly detection is strongly influenced by the size of the bottleneck in the AE. If the size of the bottleneck is excessively large, the reconstruction performance will be improved, but anomalous samples will also be well restored, defeating the purpose of the AE. By contrast, reducing the size of the bottleneck significantly affects the results corresponding to both normal and anomalous samples. The anomaly scores of some normal samples can be higher than those of anomalous samples. This phenomenon causes the over-detection issue, resulting in deterioration of the overall anomaly detection performance.

In this paper, we introduce an outlier-exposed style distillation network (OE-SDN). We identify two components of the distortions by the AE: style translation and content translation. The OE-SDN has an extensive architecture and is trained based on knowledge distillation and outlier exposure regularization to mimic style translation while suppressing content translation. To detect anomalies, we measure the difference between the outputs of the AE and OE-SDN to capture the degree of content translation while style translation is canceled out, thereby alleviating the over-detection issue.

Fig. 1 shows examples that compare the outputs of the AE and OE-SDN. In Fig. 1a, the AE blurs the bristles and changes the overall tone from yellow to red. We regard these mild distortions as style translation. In Fig. 1b, the AE transforms abnormal areas such that they resemble normal

The associate editor coordinating the review of this manuscript and approving it for publication was Tyson Brooks.



(a) Normal region

(b) Anomalous region

FIGURE 1. Examples of distortions arising from the AE and OE-SDN in normal and anomalous regions. In (a) and (b), the left, middle, and right represent the input image, the output of the AE, and the output of the OE-SDN, respectively.

areas by generating some bristles to replace the missing ones. These extreme distortions are regarded as content translation. Fig. 1a shows that the OE-SDN blurs the bristles and changes the color as the AE does. However, as shown in Fig. 1b, the OE-SDN does not replace the missing bristles, unlike the AE.

II. RELATED WORK

A. RECONSTRUCTION-BASED UNSUPERVISED ANOMALY DETECTION

A prevalent choice for anomaly detection is reconstruction-based anomaly detection using such models as an AE [5], a variational autoencoder (VAE) [6], and a generative adversarial network (GAN) [8]. It identifies a sample as an anomaly if the reconstruction error is above a certain threshold. The anomaly detection performance degrades if the reconstruction error of an anomaly is lower than the threshold. Gong *et al.* [7] used an AE with a memory module to calibrate the reconstruction error of anomalous samples. Zong *et al.* [9] considered both the distance of features and the reconstruction error to detect anomalous samples.

Unlike previous studies [7], [9] that attempted to detect anomalous samples with a low reconstruction error, this study targets normal samples with a high reconstruction error.

B. OUT-OF-DISTRIBUTION DETECTION ON LABELED DATA

The aim of out-of-distribution (OOD) detection on labeled data is to construct a classifier to identify whether input data were sampled from the distribution of a training set or from a novel distribution [10]–[13]. Hendrycks *et al.* [13] suggested that the confidence can be attributed to samples based on the maximum prediction value by the classifier and that samples with a confidence value less than a fixed threshold can be rejected and regarded as OOD samples. Liang *et al.* [12] used adversarial perturbation [14] and temperature scaling to lower the confidence of a classifier when OOD samples were inferred. Some previous studies used regularization techniques to calibrate the confidence of the classifier. Lee *et al.* [10] set cross-entropy loss as a penalty term. Hendrycks *et al.* [11] employed margin ranking loss in a similar manner.

Without the notion of a classifier, OOD detection is highly similar to anomaly detection. We borrow ideas from OOD detection to address unsupervised anomaly detection.

C. KNOWLEDGE DISTILLATION

Knowledge distillation is a method of transferring knowledge from a teacher network to a student network. Its applications mainly entail network compression. Hinton *et al.* [15] employed the predictions of a teacher network as soft labels and trained a smaller student network with these labels for a classification task. Chen *et al.* [16] and Fukuda *et al.* [17] applied knowledge distillation for object detection and speech recognition tasks, respectively.

The aforementioned applications use knowledge distillation to distill the knowledge of heavy ensemble models, achieving state-of-the-art performance with a lighter, faster network. To minimize the loss of accuracy, as much knowledge as possible should be transferred.

Unlike the conventional knowledge distillation methods, the objective of knowledge distillation for the proposed method is not compression but style mimicking. Thus, instead of transferring all knowledge from the teacher network, the proposed method aims to extract and distill only a small portion of knowledge that corresponds to style translation.

III. METHOD

A. OVERVIEW

The proposed anomaly detector comprises of two neural networks, as illustrated in Fig. 2. The first network is the *autoencoder* (AE), which reconstructs the input using a bottleneck structure. The second network is the *outlier exposed style distillation network* (OE-SDN), which imitates the output of the AE with an extensive non-bottleneck structure. Given a test sample, the proposed method calculates the anomaly score by comparing the AE and OE-SDN outputs.

B. AUTOENCODER

AE $f_{AE} : \mathbb{X} \rightarrow \mathbb{X}$ is a neural network that is trained to reconstruct its input, where $\mathbb{X}$ is the input data space. An AE has a bottleneck layer with fewer hidden units than the input for dimensionality reduction.

1) TRAINING

Suppose the training set consisting only of normal data, denoted by $\mathcal{X}_{\text{normal}} \subset \mathbb{X}$, is given. Then, the objective function for the AE f_{AE}, denoted by $\mathcal{J}_{AE}$, is as follows:

$$\mathcal{J}_{AE}(f_{AE}) = \sum_{x \in \mathcal{X}_{\text{normal}}} d\left(x, f_{AE}(x)\right). \tag{1}$$

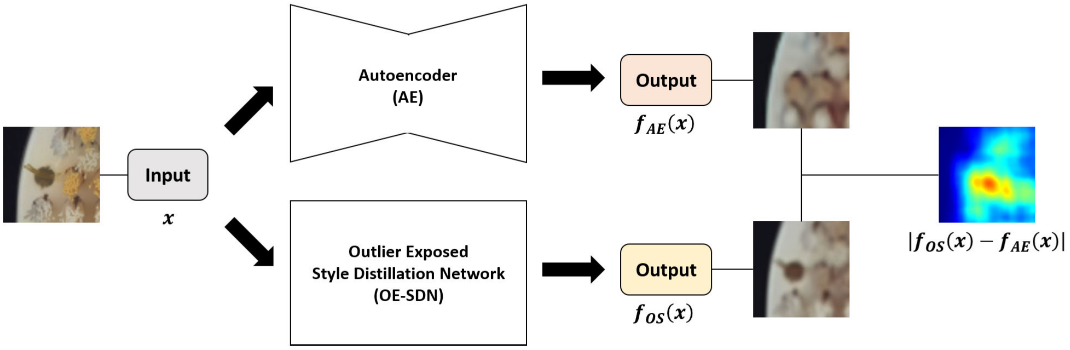
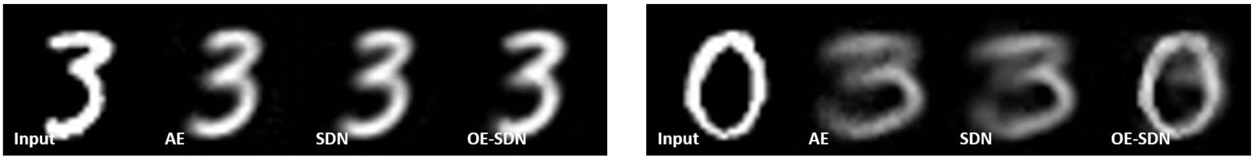

Input
x
Autoencoder (AE)
Output
$f_{AE}(x)$
Outlier Exposed Style Distillation Network (OE-SDN)
Output
$f_{OS}(x)$
$|f_{OS}(x) - f_{AE}(x)|$

FIGURE 2. Overview of the proposed anomaly detector.



(a) Results of normal class "3"

(b) Results of anomalous class "0"

FIGURE 3. The AE, SDN and OE-SDN are trained on the class "3" subset of the MNIST dataset, where the SDN is the network trained without the OER term. Due to the bottleneck, the AE can only reconstruct the class "3" samples and it transforms anomalous samples ("0" in this case) to the normal class "3." (a) Both the SDN and OE-SDN successfully imitate the blurring style of the AE. (b) The output of the SDN is the same as the output of the AE even for anomalous data; this means that the SDN mimics extreme distortion as well as mild distortion. On the other hand, the OE-SDN successfully reproduces anomalous data without extreme distortion.

where $d : \mathbb{X} \times \mathbb{X} \rightarrow \mathbb{R}$ is a loss function, which can be l_1 distance, l_2 distance, mean squared displacement (MSD), or structural dissimilarity (DSSIM) [18]. The reconstruction error is expected to be low for normal samples, whereas it is high for anomalous samples because the bottleneck may cause the network to be unable to reconstruct anomalous samples.

2) ANOMALY DETECTION

Given a test sample x_{test}, the AE identifies whether it is anomalous by using the reconstruction error as the anomaly score:

$$\epsilon(x_{\text{test}}) = ||x_{\text{test}} - f_{AE}(x_{\text{test}})||. \tag{2}$$

If the anomaly score is higher than a predefined threshold, then the sample is identified as an anomaly.

The anomaly detection performance is affected by the size of the bottleneck layer. If the size of the bottleneck layer is small, the AE will considerably distort anomalous samples. However, the AE will also distort normal samples, especially those with infrequent or complex patterns. The unintended distortions of normal samples will deteriorate the overall anomaly detection performance.

C. *OUTLIER-EXPOSED STYLE DISTILLATION NETWORK*

OE-SDN $f_{OS} : \mathbb{X} \rightarrow \mathbb{X}$ is a large neural network with no bottleneck layer that mimics the style translation of an AE f_{AE}. To train an OE-SDN, we propose an objective function based on knowledge distillation with regularization.

1) KNOWLEDGE DISTILLATION

To make the OE-SDN imitate the style translation of AE, we adopt knowledge distillation [15]. We distill the knowledge of the AE and provide it to the OE-SDN. Given a training dataset $\mathcal{X}_{\text{normal}}$ containing normal samples, we define the knowledge distillation term $\mathcal{L}_{KD}$ as:

$$\mathcal{L}_{KD}(f_{OS}) = \sum_{x \in \mathcal{X}_{\text{normal}}} d\left(f_{OS}(x), f_{AE}(x)\right). \tag{3}$$

2) OUTLIER EXPOSURE REGULARIZATION

Knowledge distillation from the small AE to the large OE-SDN results in the OE-SDN learning the style translation of the AE. However, the trained OE-SDN may also imitate some extreme distortions of the AE, as shown in Fig. 3. Hence, we adapt the concept of outlier exposure [11] to the regularization for the OE-SDN. We define the outlier exposure regularization (OER) term $\mathcal{L}_{OER}$ with an auxiliary dataset $\mathcal{X}_{\text{aux}}$ as follows:

$$\mathcal{L}_{OER}(f_{OS}) = \sum_{\tilde{x} \in \mathcal{X}_{\text{aux}}} d\left(\tilde{x}, f_{OS}(\tilde{x})\right). \tag{4}$$

This term regularizes the OE-SDN to reproduce the input as much as possible, preventing extreme changes in anomalous data. This term does not negatively affect the style-mimicking tendency because it is based on auxiliary data that is disjointed from the training data.

The auxiliary dataset $\mathcal{X}_{\text{aux}}$ can be obtained in various ways, such as synthetic data generated using a GAN [10], real data adopted from a similar domain [11], geometrically transformed data, and blurred data [19]. In this study, we use rotation transformation according to the results of the preliminary experiment discussed in subsection IV-C. The 90°, 180°, and 270° rotations of the original samples in $\mathcal{X}_{\text{normal}}$ are generated to create $\mathcal{X}_{\text{aux}}$.

3) TRAINING

The objective function for the OE-SDN f_{OS}, denoted by $\mathcal{J}_{OS}$, is as follows:

$$\mathcal{J}_{OS}(f_{OS}) = (1-\lambda)\cdot\mathcal{L}_{KD}(f_{OS}) + \lambda\cdot\mathcal{L}_{OER}(f_{OS}), \qquad (5)$$

where λ is the hyperparameter for balancing the knowledge distillation term and the OER term.

Given the training dataset $\mathcal{X}_{\text{normal}}$, auxiliary dataset $\mathcal{X}_{\text{aux}}$, and pre-trained AE f_{AE}, the OE-SDN f_{OS} is trained to minimize the objective function $\mathcal{J}_{OS}$. The AE f_{AE} is fixed during the training.

4) ANOMALY DETECTION

To detect anomalies, we adopt an alternate anomaly score instead of a reconstruction error. Given a test sample x_{test}, the anomaly score is calculated by measuring the difference between the outputs of the OE-SDN and AE:

$$\epsilon'(x_{\text{test}}) = ||f_{OS}(x_{\text{test}}) - f_{AE}(x_{\text{test}})||. \qquad (6)$$

IV. EXPERIMENTS

This section describes the evaluation of the proposed method for two unsupervised anomaly detection tasks: classification and segmentation. Each task has different benchmark datasets and baseline methods. All the experiments were implemented using PyTorch [20].

The common configurations for the two tasks are as follows. We used DSSIM [18] and MSD for the loss functions of the AE and OE-SDN, respectively. We set the hyperparameter λ to 0.5 for the OE-SDN. We used the Adam [21] optimizer for training. To calculate the anomaly scores corresponding to the AE and OE-SDN, DSSIM with a window size of 7 was used.

We evaluated the performance of each method in terms of the area under the receiver operating characteristic curve (AUROC), which is calculated independently of the threshold. For the anomaly classification task, the AUROC is calculated using image-wise anomaly scores. For the anomaly segmentation task, the AUROC is calculated using pixel-wise anomaly scores.

A. UNSUPERVISED ANOMALY CLASSIFICATION

We evaluated our method for the MNIST [22] and CIFAR-10 [23] datasets for the classification task of unsupervised anomaly detection. Both datasets had 10 classes from which we created 10 setups similar to those created by Ruff *et al.* [24]. In each setup, one class was chosen as the normal class and the remaining were the anomalous classes. Every setup had approximately 6,000 training images in the MNIST dataset or 5,000 in the CIFAR-10 dataset. The number of test images was 10,000 for both sets.

For both the MNIST and CIFAR-10 datasets, we implemented an AE with a simple architecture composed of fully-connected layers. The encoder consisted of four layers of 128, 64, 32, and 10 units. The decoder was designed symmetrically to the encoder. Leaky rectified linear units (LReLUs) with a slope of 0.005 were applied after each layer except for the output layer. Regarding the OE-SDN, we constructed the architecture using the residual attention block of residual non-local attention networks (RNAN) [25], which has proven to be effective in general image translation tasks. Fig. 4 shows the detailed architecture of the OE-SDN. We set an initial learning rate of 0.001 for the AE and 0.0001 for the OE-SDN. We trained the AE in 50k iterations and the OE-SDN in 10k iterations, which was sufficient for the loss of each network to converge. The batch size was 20 for both networks.

We compared our method with four baseline methods: one-class support vector machine (OC-SVM) [26], isolation forest (IF) [27], GAN for anomaly detection (AnoGAN) [8], and deep support vector data description (DeepSVDD) [24]. The experimental results for these four baseline methods were

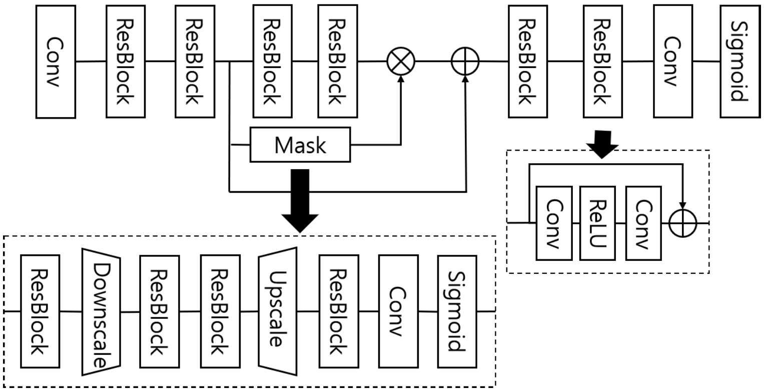


FIGURE 4. Architecture diagram for the OE-SDN. The architecture adapts the RNAN [25], in which we set the number of filters to 32 in all convolution layers.

TABLE 1. **AUROC of each method for unsupervised anomaly classification on MNIST and CIFAR-10 datasets. The highest AUROC for each setup is highlighted in bold.**

Dataset	Normal Class	OC-SVM [26]	IF [27]	AnoGAN [8]	DeepSVDD [24]	AE	SDN	OE-SDN
MNIST	0	0.986	0.980	0.966	0.980	**0.991**	0.989	0.984
	1	0.995	0.973	0.992	0.997	**0.998**	**0.998**	**0.998**
	2	0.825	0.886	0.850	0.917	0.965	0.952	**0.983**
	3	0.881	0.899	0.887	0.919	0.950	0.933	**0.974**
	4	0.949	0.927	0.894	0.949	0.923	0.933	**0.977**
	5	0.771	0.855	0.883	0.885	0.958	0.953	**0.973**
	6	0.965	0.956	0.947	0.983	0.992	0.991	**0.995**
	7	0.937	0.920	0.935	0.946	0.946	0.948	**0.971**
	8	0.889	0.899	0.849	**0.939**	0.912	0.920	0.921
	9	0.931	0.935	0.924	0.965	0.970	0.968	**0.972**
	Average	0.913	0.923	0.913	0.948	0.961	0.958	**0.975**
CIFAR-10	Airplane	0.616	0.601	0.671	0.617	**0.790**	0.677	0.774
	Automobile	0.638	0.508	0.547	0.659	0.661	0.671	**0.821**
	Bird	0.500	0.492	0.529	0.508	**0.647**	0.598	0.638
	Cat	0.559	0.551	0.545	**0.591**	0.515	0.520	0.569
	Deer	**0.660**	0.498	0.651	0.609	0.587	0.582	0.594
	Dog	0.624	0.585	0.603	0.657	0.571	0.535	**0.659**
	Frog	**0.747**	0.429	0.585	0.677	0.596	0.648	0.629
	Horse	0.626	0.551	0.625	0.673	0.630	0.611	**0.755**
	Ship	0.749	0.742	0.758	0.759	0.783	0.707	**0.844**
	Truck	0.759	0.589	0.665	0.731	0.623	0.608	**0.784**
	Average	0.648	0.555	0.618	0.648	0.640	0.616	**0.707**

obtained from [24]. We also used the AE and SDN (OE-SDN with $\lambda = 0$) as baselines.

The results are shown in Table 1. The proposed OE-SDN outperformed the baseline methods in terms of the average AUROC for the MNIST and CIFAR-10 datasets. For the setups with Dog, Horse, or Truck as a normal class on the CIFAR-10, the AE performed worse than DeepSVDD, whereas the OE-SDN yielded superior results. This demonstrates the effectiveness of the OE-SDN. The SDN performed worse than the AE in many cases for both datasets. The SDN learned not only style translation but also extreme distortions from the AE, as shown in Fig. 3. The OER mitigated this problem, resulting in a higher AUROC. For the setups with Airplane or Bird as a normal class in the CIFAR-10 dataset, the OE-SDN performed worse than the AE. There were more normal images that the OE-SDN can easily reproduce while the AE reconstructed with distortions, leading to more false positive samples than in other setups. An image of an airplane flying without landing on the ground is an example.

B. UNSUPERVISED ANOMALY SEGMENTATION

We used the MVTec-AD dataset [4] to assess unsupervised anomaly segmentation performance. It contained 5354 images corresponding to 10 object categories and 5 texture categories that represent real-world inspection scenarios. The training set of the MVTec-AD dataset had only normal images, and the test set contained defect images with a pixel-wise ground-truth mask.

Considering that the MVTec-AD dataset contained relatively large images, we adopted a modern convolutional neural network (CNN) architecture to implement the AE. The detailed architecture is shown in Fig. 5. We used the same architecture for the OE-SDN as in the classification experiment.

Texture images were cropped to 256 × 256 patches at random locations for training. We used these patches as inputs for the networks. Object images were resized to the same size before being fed into the networks. We set the initial learning rate to 0.0001 for both the AE and OE-SDN. The networks were trained for 10k iterations for the AE and 1k iterations for the OE-SDN with a batch size of 20. In the test phase, texture images were cropped into 256 × 256 patches with a stride of 64 before inference, and the maximum anomaly score was considered when aggregating pixel-wise anomaly scores of the overlapping patches. After calculating the difference, we applied a 32 × 32 uniform filter to smooth pixels whose anomaly scores differed significantly from their surrounding pixels.

We investigated the effectiveness of the proposed method in comparison with three unsupervised anomaly segmentation methods that were used as baselines in [4]: AnoGAN [8], a method based on CNN feature similarity [28], and an AE [4] with an alternative architecture. The experimental results for these three baselines were obtained from [4]. We also used our AE and SDN as baseline methods to evaluate the validity of the components of our method similar to the classification experiment.

As shown in Table 2, the OE-SDN and SDN achieved the best performance on average for both textures and objects. The OE-SDN obtained considerable AUROC gain for categories in which the AE had low AUROC, such as Tile, Wood, Capsule, Metal Nut, and Zipper. On the other hand,

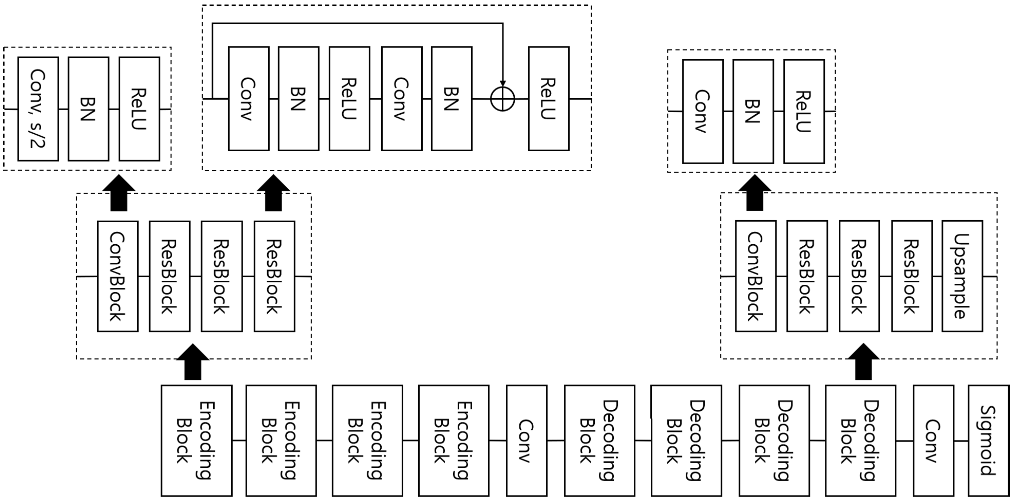


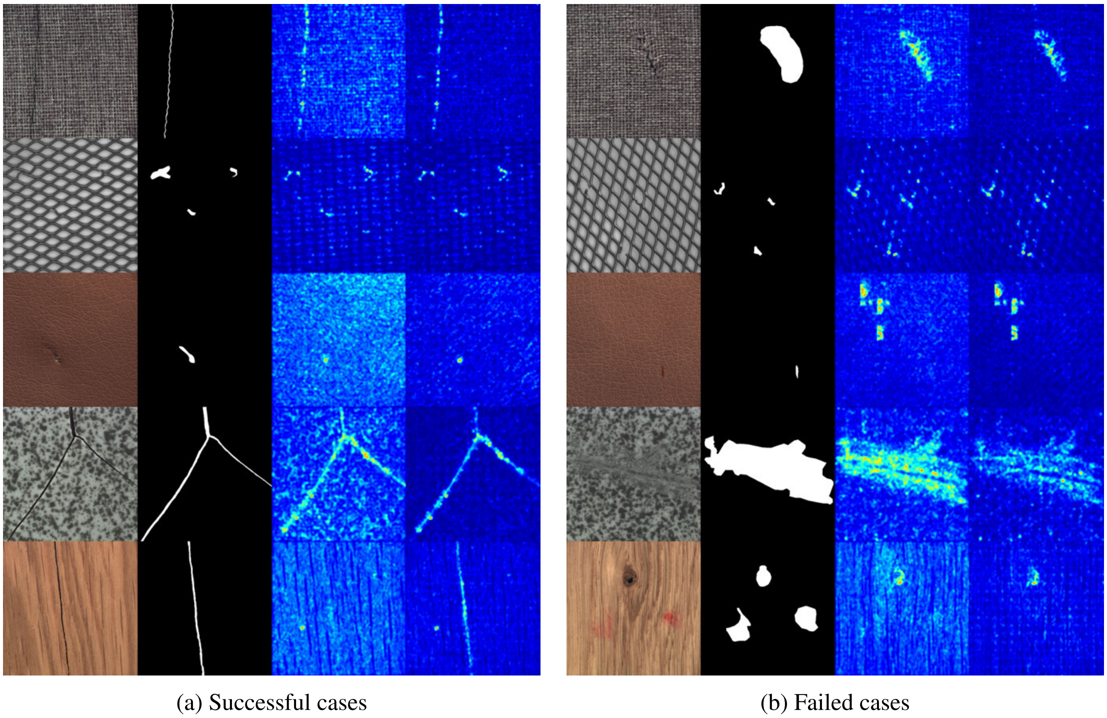
FIGURE 5. Architecture diagram for AE for segmentation experiment. Except for the last convolution layer, all convolution filters have a kernel size of 3 × 3. The number of filters in each encoding block is 8, 16, 32, and 64 in order; the number of filters in each decoding block is 64, 32, 16, and 8 in order. The convolution layer between the encoding and decoding blocks has 128 filters. For downsampling, the initial convolution in each encoding block has a stride of 2. The upsample layer in each decoding block performs bilinear interpolation to accomplish upsampling.

TABLE 2. Pixel-wise AUROC of each method for unsupervised anomaly segmentation on the MVTec-AD dataset. The highest AUROC for each category is highlighted in bold.

Category		AnoGAN [8]	CNN Feature Similarity [28]	AE [4]	AE (ours)	SDN	OE-SDN
Textures	Carpet	0.54	0.72	0.87	0.95	**0.96**	**0.96**
	Grid	0.58	0.59	0.94	**0.97**	**0.97**	**0.97**
	Leather	0.64	**0.87**	0.78	0.85	0.83	0.85
	Tile	0.50	**0.93**	0.59	0.81	0.88	0.85
	Wood	0.62	**0.91**	0.73	0.79	0.82	0.82
	Average	0.58	0.80	0.78	0.87	**0.89**	**0.89**
Objects	Bottle	0.86	0.78	0.93	0.94	**0.96**	0.95
	Cable	0.78	0.79	0.82	**0.84**	**0.84**	**0.84**
	Capsule	0.84	0.84	0.94	0.92	**0.97**	**0.97**
	Hazelnut	0.87	0.72	0.97	0.97	**0.98**	**0.98**
	Metal nut	0.76	0.82	0.89	0.89	**0.95**	0.93
	Pill	0.87	0.68	0.91	**0.94**	0.92	0.93
	Screw	0.80	0.87	0.96	0.97	**0.98**	0.97
	Toothbrush	0.90	0.77	0.92	**0.98**	**0.98**	**0.98**
	Transistor	0.80	0.66	**0.90**	0.88	0.89	0.89
	Zipper	0.78	0.76	0.88	0.87	0.89	**0.91**
	Average	0.83	0.77	0.91	0.92	**0.93**	**0.93**

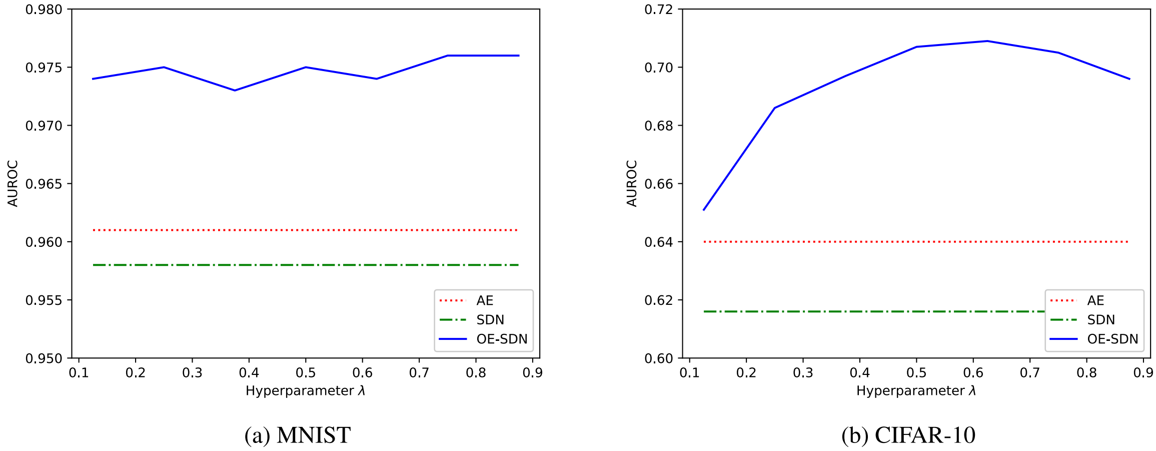
the advantage of the OE-SDN was insignificant for categories in which the AE already detected anomalies well, such as Carpet, Grid, Hazelnut, Screw, and Toothbrush. The OE-SDN yielded a slightly lower AUROC than the AE in the Pill category which contained the defect patterns that the OE-SDN cannot reproduce. For example, a slightly changed color was not reproduced by the OE-SDN, which resulted in missing abnormal pixels.

Fig. 6 shows examples of cases in which the OE-SDN exhibited improved performance and those in which it does not. In successful cases, shown in Fig. 6a, the high anomaly scores generated by the AE for the normal regions were significantly suppressed by using the OE-SDN. However, there were also cases in which the OE-SDN was not effective, as shown in Fig. 6b. In the first, second, and fourth rows, when the AE detected abnormal areas appropriately without over-detecting the normal areas, the OE-SDN did not provide further performance improvement. In the third and fifth rows, the AE originally did not produce significant anomaly scores for abnormal areas. Because the OE-SDN suppressed anomaly scores for normal areas while maintaining anomaly scores for abnormal areas, it was difficult to expect performance improvement in such cases.


(a) Successful cases

(b) Failed cases

FIGURE 6. Successful and failed cases corresponding to our method on texture categories of the MVTec-AD dataset. Each row shows the results of each category in the order Carpet, Grid, Leather, Tile, and Wood. Each column represents a raw image, ground-truth mask, pixel-wise difference of raw input and reconstruction from the AE, and pixel-wise difference of the outputs of the AE and OE-SDN.



(a) MNIST

(b) CIFAR-10

FIGURE 7. Sensitivity analysis of the hyperparameter λ.

C. EFFECT OF AUXILIARY DATASET AND HYPERPARAMETER

We investigated the effect of the auxiliary dataset $\mathcal{X}_{\text{aux}}$ and hyperparameter λ. Both were verified for the MNIST and CIFAR-10 datasets, where the OER played a significant role.

1) COMPARATIVE EXPERIMENT FOR AUXILIARY DATASET

We performed a comparative experiment with regard to the auxiliary dataset $\mathcal{X}_{\text{aux}}$ on the MNIST and CIFAR-10 datasets, as shown in Table 3. We generated auxiliary datasets using geometric transformation (random rotation, flip, and shearing), noise addition (Gaussian noise and adversarial noise [29]), blurring (Gaussian blurring and singular value decomposition blurring), and AE-based reconstruction. Most choices of $\mathcal{X}_{\text{aux}}$ improved the performance of the OE-SDN. Among them, rotation and vertical flip yielded superior performances. They are simple and prevalent techniques for data augmentation, meaning that it is not necessary to devise an entirely novel auxiliary dataset.

TABLE 3. Results of the comparative experiment for the auxiliary dataset generation methods. Results are shown in terms of average AUROC of each setup on the MNIST and CIFAR-10 datasets.

Method	MNIST	CIFAR-10
Adversarial Noise [29]	0.960	0.631
Gaussian Blurring	0.954	0.645
Gaussian Noise	0.956	0.638
Horizontal Flip	0.968	0.638
Reconstruction of AE	0.959	0.623
Rotation	**0.975**	**0.707**
Shearing	0.958	0.619
SVD Blurring	0.952	0.640
Vertical Flip	0.968	0.694
No Auxiliary Dataset (SDN)	0.958	0.616

2) SENSITIVITY ANALYSIS OF HYPERPARAMETER

We used the MNIST and CIFAR-10 datasets to study the robustness of the hyperparameter λ. We first set the auxiliary dataset $\mathcal{X}_{\text{aux}}$ as randomly rotated data, which achieved the best performance. We calculated the performance of the proposed anomaly detector by changing the value of λ. Fig. 7 shows the average AUROC of each setup with varying λ values. For every λ value, we observed that the OE-SDN surpasses both the AE and SDN; this means that our regularization is satisfactory and robust. For the MNIST and CIFAR-10 datasets, if we set a sufficiently large λ, then the OER can effectively regularize the network to prevent mimicking the extreme distortion exhibited by the AE.

V. CONCLUSION

In this study, we presented the OE-SDN to overcome the over-detection issue of conventional reconstruction-based anomaly detection methods. Considering an AE, the OE-SDN was trained with two objectives: knowledge distillation and outlier exposure regularization. Consequently, the OE-SDN preserved the style translation and suppressed the content translation of the AE. We introduced an alternate anomaly score defined as the difference between the outputs of the AE and OE-SDN. Experiments on real datasets showed that our method outperforms existing methods, including reconstruction-based anomaly detection using AEs.

In future work, we will investigate various regularization and learning methods used in style transfer studies to train the OE-SDN effectively. Further, because the OE-SDN can be used with any other reconstruction-based anomaly detection method, we will apply the OE-SDN to other recent methods to improve the anomaly detection performance.

REFERENCES

[1] W. Luo, W. Liu, and S. Gao, “A revisit of sparse coding based anomaly detection in stacked RNN framework,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 341–349.

[2] C. Baur, B. Wiestler, S. Albarqouni, and N. Navab, “Deep autoencoding models for unsupervised anomaly segmentation in brain MR images,” in *Brainlesion: Glioma, Multiple Sclerosis, Stroke Traumatic Brain Injuries*. Cham, Switzerland: Springer, 2019, pp. 161–169.

[3] A. Beghi, L. Cecchinato, C. Corazzol, M. Rampazzo, F. Simmini, and G. A. Susto, “A one-class SVM based tool for machine learning novelty detection in HVAC chiller systems,” *IFAC Proc. Volumes*, vol. 47, no. 3, pp. 1953–1958, 2014.

[4] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, “MVTec AD a comprehensive real-world dataset for unsupervised anomaly detection,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9592–9600.

[5] P. Bergmann, S. Löwe, M. Fauser, D. Sattlegger, and C. Steger, “Improving unsupervised defect segmentation by applying structural similarity to autoencoders,” in *Proc. 14th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, 2019, pp. 1–8.

[6] J. An and S. Cho, “Variational autoencoder based anomaly detection using reconstruction probability,” SNU Data Mining Center, Seoul, South Korea, Tech. Rep., 2015.

[7] D. Gong, L. Liu, V. Le, B. Saha, M. R. Mansour, S. Venkatesh, and A. Van Den Hengel, “Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1–8.

[8] 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 *Proc. Int. Conf. Inf. Process. Med. Imag.*, 2017, pp. 146–157.

[9] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho, and H. Chen, “Deep autoencoding Gaussian mixture model for unsupervised anomaly detection,” in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–8.

[10] K. Lee, H. Lee, K. Lee, and J. Shin, “Training confidence-calibrated classifiers for detecting out-of-distribution samples,” in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–16.

[11] D. Hendrycks, M. Mazeika, and T. Dietterich, “Deep anomaly detection with outlier exposure,” in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–18.

[12] S. Liang, Y. Li, and R. Srikant, “Enhancing the reliability of out-of-distribution image detection in neural networks,” in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–15.

[13] D. Hendrycks and K. Gimpel, “A baseline for detecting misclassified and out-of-distribution examples in neural networks,” in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–12.

[14] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adverasarial examples,” in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–11.

[15] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” in *Proc. NIPS Deep Learn. Represent. Learn. Workshop*, 2015, pp. 1–9.

[16] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, “Learning efficient object detection models with knowledge distillation,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 742–751.

[17] T. Fukuda, M. Suzuki, G. Kurata, S. Thomas, J. Cui, and B. Ramabhadran, “Efficient knowledge distillation from an ensemble of teachers,” in *Proc. Interspeech*, Aug. 2017, pp. 3697–3701.

[18] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[19] S. Choi and S.-Y. Chung, “Novelty detection via blurring,” in *Proc. Int. Conf. Learn. Represent.*, 2020, pp. 1–14.

[20] A. Paszke, “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.

[21] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Int. Conf. Learn. Represent.*, 2015, pp. 1–15.

[22] Y. LeCun and C. Cortes. (2010). *MNIST Handwritten Digit Database*. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[23] A. Krizhevsky, “Learning multiple layers of features from tiny images,” M.S. thesis, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, 2009.

[24] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, “Deep one-class classification,” in *Proc. Int. Conf. Mach. Learn.*, vol. 80, 2018, pp. 4393–4402.

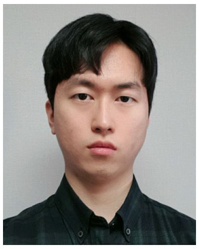
[25] Y. Zhang, K. Li, K. Li, B. Zhong, and Y. Fu, “Residual non-local attention networks for image restoration,” 2019, *arXiv:1903.10082*. [Online]. Available: http://arxiv.org/abs/1903.10082

[26] B. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson, “Estimating the support of a high-dimensional distribution,” *Neural Comput.*, vol. 13, no. 7, p. 1443–1471, 2001.

[27] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *IEEE Int. Conf. Data Mining*, 2008, pp. 413–422.

[28] P. Napoletano, F. Piccoli, and R. Schettini, "Anomaly detection in nanofibrous materials by CNN-based self-similarity," *Sensors*, vol. 18, no. 2, p. 209, Jan. 2018.

[29] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–11.



HWEHEE CHUNG received the B.S. and M.S. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), in 2016 and 2018, respectively. He is currently a Researcher with the Cognex Deep Learning Lab. His current research interest includes network debugging in the field of vision inspection systems in manufacturing.

JONGHO PARK received the B.S. and M.S. degrees in computer science and engineering from Seoul National University, in 2015 and 2017, respectively. He is currently a Researcher with the Cognex Deep Learning Lab. His research interest includes optimizing the productivity in vision inspection industry by adapting deep learning algorithms.

JONGSOO KEUM received the B.S. and M.S. degrees in electrical engineering and computer science from the Gwangju Institute of Science and Technology, in 2015 and 2017, respectively. He is currently a Researcher with the Cognex Deep Learning Lab. His main research interest includes optimized modeling of deep learning architecture in the field of vision inspection system in manufacturing.

HONGDO KI received the B.S. and M.S. degrees in industrial engineering from Seoul National University, in 2015 and 2017, respectively. He is currently a Researcher with the Cognex Deep Learning Lab. His research interest includes reducing the cost of applying a learning-based vision inspection system in manufacturing.

SEOKHO KANG received the B.S. and Ph.D. degrees in industrial engineering from Seoul National University, in 2011 and 2015, respectively. He was a Research Staff Member with the Samsung Advanced Institute of Technology. He is currently an Assistant Professor of systems management engineering (industrial engineering) with Sungkyunkwan University. His main research interest includes developing learning algorithms for efficient data-driven modeling and their applications to real-world data mining problems in manufacturing, healthcare, and materials informatics.