Learning Semantic Context from Normal Samples for Unsupervised Anomaly Detection

Xudong Yan,[1*] **Huaidong Zhang,**[1*] **Xuemiao Xu,**[1,2,3,4†] **Xiaowei Hu,**[5] **Pheng-Ann Heng**[5,6]

[1]South China University of Technology
[2]Ministry of Education Key Laboratory of Big Data and Intelligent Robot
[3]Guangdong Provincial Key Lab of Computational Intelligence and Cyberspace Information
[4]State Key Laboratory of Subtropical Building Science
[5]The Chinese University of Hong Kong
[6]Shenzhen Key Laboratory of Virtual Reality and Human Interaction Technology,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
{csyanxd, z.huaidong}@mail.scut.edu.cn, xuemx@scut.edu.cn, {xwhu, pheng}@cse.cuhk.edu.hk

Abstract

Unsupervised anomaly detection aims to identify data samples that have low probability density from a set of input samples, and only the normal samples are provided for model training. The inference of abnormal regions on the input image requires an understanding of the surrounding semantic context.This work presents a *Semantic Context based Anomaly Detection Network*, SCADN, for unsupervised anomaly detection by learning the semantic context from the normal samples. To achieve this, we first generate multi-scale striped masks to remove a part of regions from the normal samples, and then train a generative adversarial network to reconstruct the unseen regions. Note that the masks are designed in multiple scales and stripe directions, and various training examples are generated to obtain the rich semantic context. In testing, we obtain an error map by computing the difference between the reconstructed image and the input image for all samples, and infer the abnormal samples based on the error maps. Finally, we perform various experiments on three public benchmark datasets and a new dataset LaceAD collected by us, and show that our method clearly outperforms the current state-of-the-art methods.

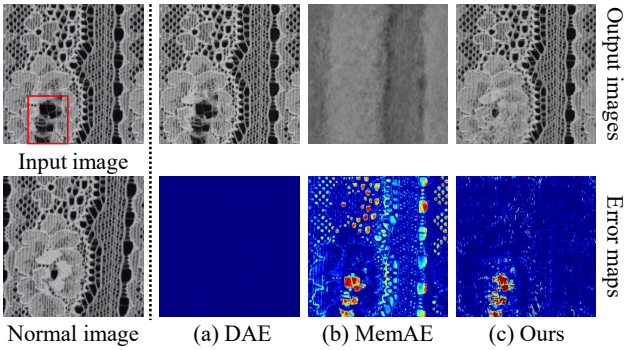

Figure 1: The visual comparison results among DAE (Zhou and Paffenroth 2017), MemAE (Gong et al. 2019), and our method. The error map of DAE fails to highlight the abnormal region in (a) while the error map of MemAE tends to include noises and artifacts in (b). In contrast, our method in (c) is able to recover the anomaly region with normal patterns by exploring the semantic context in surroundings.

Introduction

Anomaly samples are the data, which are residing in low probability density from a set of input samples. Knowing the anomaly samples helps lots of applications, e.g., defect detection (Bergmann et al. 2019, 2020), security check (Akcay, Atapour-Abarghouei, and Breckon 2018), intruder detection (Oza and Patel 2019), medical image inspection (Schlegl et al. 2017), and outlier removal (Xia et al. 2015). Thus, anomaly detection has long been a fundamental research problem.

However, there are various types of anomalies in the real world, and it is difficult to obtain the training samples of the anomaly, which rarely appeared. To address these problems, the research works explore the unsupervised learning strategy, which only learns the characteristics of the normal samples and infer the anomaly samples directly. Among them, early works adopt one-class classification (Wang and Cherian 2019; Ruff et al. 2018) or probability estimation (Zong et al. 2018; Golan and El-Yaniv 2018; Pidhorskyi, Almohsen, and Doretto 2018; Abati et al. 2019) to build a model that represents the normal distribution, and detects the abnormal samples that have different distributions with the model.

Recently, methods based on the deep autoencoder (DAE) (Sabokrou et al. 2018; Zhou and Paffenroth 2017) show remarkable performance on the benchmark datasets (Krizhevsky and Hinton 2009). The key factor for the successes is that the encoder first learns to map the data samples into a latent space and then transfer them back to the original images by the decoder. Hence, after training the autoencoder on normal samples, they are able to well reconstruct the normal images but fail to obtain the reconstructed

Copyright © 2021, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.
[*] Both authors contributed equally.
[†] Corresponding author: Xuemiao Xu.

abnormal images. By computing the error map between the input images and the reconstructed images, we are able to perform anomaly detection; see Fig. 1 (a), (b) for an example. However, these approaches by reconstructing images for anomaly detection have some limitations. As described in the existing literature (Perera, Nallapati, and Xiang 2019; Gong et al. 2019), if anomalies share common local patterns with the normal regions, vanilla DAE tends to directly use the original image as the output; see the results in Fig. 1 (a). To avoid this issue, MemAE (Gong et al. 2019) constraints the coding space in the autoencoder. However, it is easy to ignore the image details and leads to a blurry output image; see the results in Fig. 1 (b). Hence, the computed error maps may fail to capture the abnormal regions or include lots of noises and artifacts.

As shown in Fig. 1 (c), our method successfully detects the abnormal region (the hole in red box) from the input image by generating the similar patterns with the surroundings in such a region. This is because our method is able to capture the semantic context, which provides guidance on how to recover the anomaly regions that have large differences with the normal surroundings. To this end, we design a novel deep network architecture via learning semantic context from normal samples, i.e., *Semantic Context based Anomaly Detection Network* (SCADN), which learns the semantic context from the normal samples by reconstructing a part of the unseen region, and infers anomaly samples in an unsupervised manner. Given an input, we first remove a part of the image and then feed it into a deep neural network to learn to recover the missing regions. Since all the pixel values in missing regions are lost, our network is forced to learn surrounding semantic features to complete this region. We further design the multi-scale striped masks to determine where to remove and recover, which encourages our model to learn the semantic context from different locations, scales, and directions. We summarize our contributions as followed:

1. We present a new network architecture to learn the semantic context features for anomaly detection in an unsupervised manner.
2. We construct a multi-scale semantic context learning framework to detect anomalies across the whole image with different scales.
3. We build a large-scale real-world unsupervised image anomaly detection dataset (*LaceAD*) that contains $9,176$ high-resolution lace fabric images belonging to 17 patterns, which helps to enrich the datasets of unsupervised anomaly detection.

Related Work

Model based methods for anomaly detection aim to build a model to represent the normal distribution. In this way, the anomaly is detected when the testing data have different distributions with the model. Some methods (Schölkopf et al. 2001; Chen, Zhou, and Huang 2001; Tax and Duin 2004; Wang and Cherian 2019; Ruff et al. 2018; Perera and Patel 2019; Chalapathy, Menon, and Chawla 2018) adopted the one-class classification to build the model, which learns a feature space that has a low intra-class variance and aims to obtain a discriminative hyperplane surrounding the normal samples. Others (Parzen 1962; Kim and Scott 2012; Zong et al. 2018; Zhai et al. 2016; Golan and El-Yaniv 2018; Ahmed and Courville 2020) set up a parametric or non-parametric probability estimator about normal features. However, these methods may fail to model the complex data samples and the decision boundary between the normal and abnormal samples is unclear when only the normal data are available.

Self-representation based methods for anomaly detection assume anomalies yield higher representation error by a model learned from the normal data. Sparse representation (Zhao, Fei-Fei, and Xing 2011; Lu, Shi, and Jia 2013; Cong, Yuan, and Liu 2011) and PCA (Kim and Grauman 2009; Candès et al. 2011) were developed for anomaly detection. In recent works, deep autoencoder was widely used in self-representation (Vincent et al. 2008; Akcay, Atapour-Abarghouei, and Breckon 2018; Xia et al. 2015; Zhou and Paffenroth 2017; Tian et al. 2019). (Sabokrou et al. 2018) used a discriminator to train AE in an adversarial manner and then identified anomalies by discriminator output; (Liu et al. 2020) proposed the gradient-based attention for VAEs to localize anomalies in images.

OCGAN (Perera, Nallapati, and Xiang 2019) constrained the latent space to be close to a uniform distribution with the small range; MemAE (Gong et al. 2019) adopted the learnable memory to approximate and replace the output of the encoder, expecting the reconstructed result that closes to a normal sample. However, these methods intentionally constrain the representation ability of the network, un-avoiding to increase the reconstruction errors on the normal data; see the error map of normal regions in Fig. 1 (b) for an example. In contrast, our method encourages the network to recover the missing regions of the input images by exploring the semantic context. Since the information of these regions is completely lost, our method can avoid the network to learn a linear mapping between input and output images.

Image completion aims to recover the missing regions of the input image by exploring the surrounding semantic context. For example, (Yu et al. 2018) proposed the contextual attention layer to learn the relation between missing regions and surrounding context; To iteratively recover the missing regions, (Liu et al. 2018) and (Yu et al. 2019) used partial convolution and gated convolution to involve a learnable mask into the network, so that the features can be extracted from both reliable region and low-confident region. In addition, (Zheng, Cham, and Cai 2019) proposed a probabilistic framework to output pluralistic results. Inspired by image completion, we present to learn the semantic context by recovering missing regions for anomaly detection. Different from the image completion, we aim to increase the gap of reconstruction errors between normal and abnormal samples.

Methodology

Overview. Fig. 2 illustrates the architecture of the proposed network. During the training process (green line), we

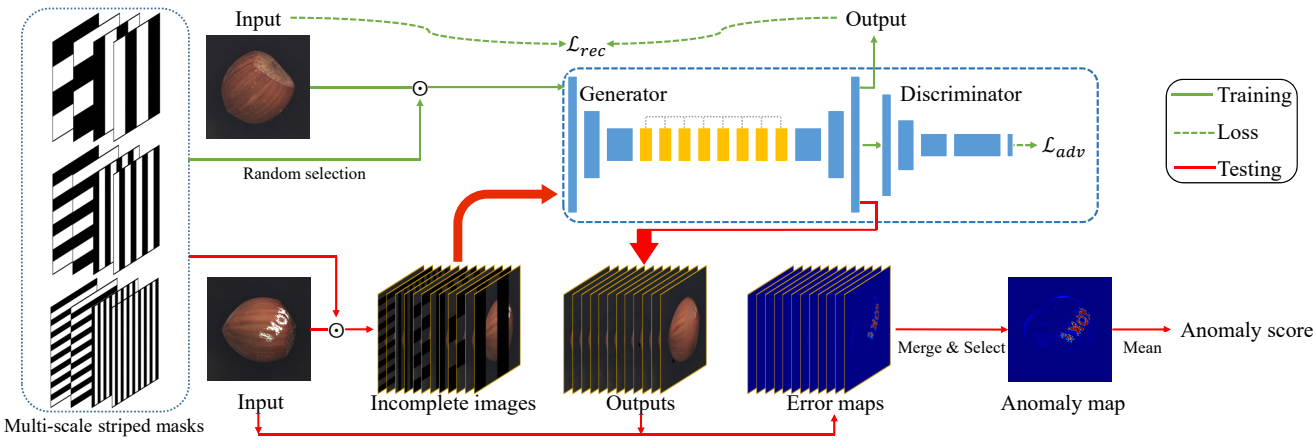

Figure 2: Overall architecture of the proposed *Semantic Context based Anomaly Detection Network* (SCADN). This network is designed to learn from normal samples by removing a part of regions in the input image and making the network to recover the missing information. We adopt the multi-scale striped masks to indicate the removed regions of the input images. In testing, we compute the errors between input and output images to obtain the anomaly score.

take an image and a binary mask as the inputs, use the mask to remove a part of regions in the image, adopt a generator to generate an image that is similar to the input image, minimize the difference between the generated image and the input image by a supervised loss, and take a discriminator to determine whether the generated image is real or not. Note that, only the normal images are used in this training process. In testing, we take a sample image with multiple masks to produce multiple incomplete images, use the trained generator to generate multiple complete images, and compute the error maps as the difference between the input image and the output images of the generator. Finally, the anomaly score is computed based on these error maps and we take the score to perform anomaly detection.

Multi-scale Context Learning

Multi-scale striped masks. As described earlier in the Introduction, we present to first remove a part of regions in the input images and then make the network learn to generate the original input images. One problem during the implementation is which part of the regions should be removed. To solve this issue, we present three design principles: (i) the image regions should have the *equal probability* to be removed, since the anomalies may appear in any position of the image; (ii) the removed regions should have *multiple scales*, since the anomalies in the real world may have different sizes; (iii) the shape of the removed regions should have *multiple directions*, and we can obtain the semantic context from surrounding in different directions.

To meet these requirements, we design several multi-scale striped masks to indicate the regions that should be removed, and we set the pixel values at these regions as zero. As shown in Fig. 3, we adopt the black color to indicate the regions to be removed, and set the ratio between the white and black regions as $1:1$. By swapping the black and white regions, we can obtain a pair of complementary masks to make each image region have an equal probability to be removed. By changing the width of the white strips, we can obtain the masks in different scales. By setting the stripes in vertical and horizontal directions, we can aggregate the semantic context from different directions.

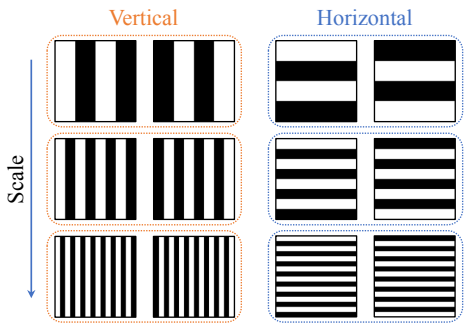

Figure 3: Visualization of our multi-scale striped masks.

Network architecture. We followed (Nazeri et al. 2019; Yu et al. 2018) to design the generator and discriminator in our network. The generator down-samples the input images as one-eighth of the input size through convolutions with the stride size of two, uses eight residual blocks to extract the feature maps, and upsamples the feature maps to the size of the input image. In the residual block, dilated convolutions with larger receptive fields are used to obtain more semantic context features. Finally, a global discriminator is adopted to determine whether the generated image is real or not.

Loss function. Let $\mathbf{I}$ be an input image. $\mathbf{M}$ is a binary mask (0 for for region to remove and 1 otherwise). The generator is denoted by G, which produces the result $\hat{\mathbf{I}}$ that aims to recover the input image:

$$\hat{\mathbf{I}} = G(\mathbf{I} \odot \mathbf{M}) , \quad (1)$$

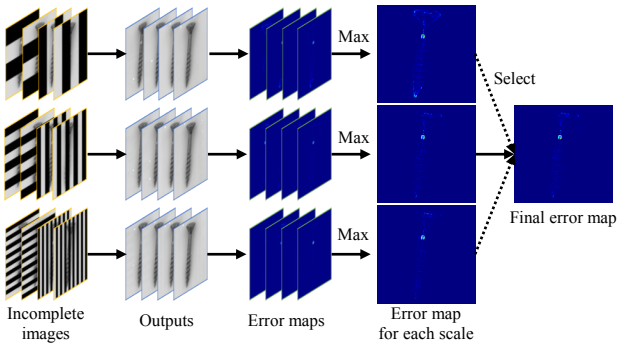

Figure 4: Illustration of our inference strategy.

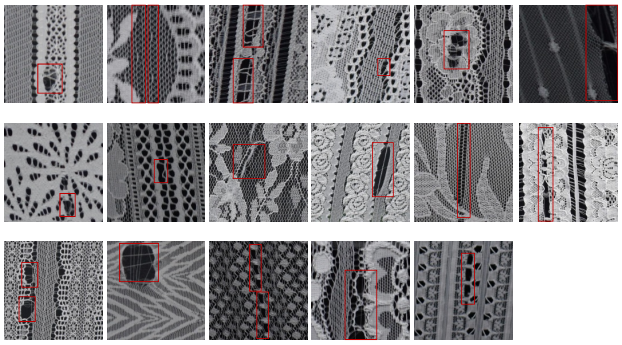

Figure 5: Examples for 17 different patterns in the LaceAD dataset. Abnormal regions are marked by red boxes.

where $\odot$ denotes the element-wise multiplication. The generator is trained over a joint loss that consists of a reconstruction loss and an adversarial loss (Goodfellow et al. 2014). For the reconstruction loss, we use the mean square error (MSE) of the whole image and add an additional weight on the removed regions $(\mathbf{1}-\mathbf{M})$ to emphasize these regions:

$$\mathcal{L}_{rec} = [\mathbf{1}+\lambda_{rm}(\mathbf{1}-\mathbf{M})] \odot \left\|\hat{\mathbf{I}}-\mathbf{I}\right\|_2^2 , \quad (2)$$

where λ_{rm} is a weight that balances these two items in the loss function. The adversarial learning loss is formulated as:

$$\mathcal{L}_{adv} = \mathbb{E}_{\mathbf{I}}[logD(\mathbf{I})] + \mathbb{E}_{\mathbf{I}}\left[log(1-D(\hat{\mathbf{I}}))\right] , \quad (3)$$

where D denotes discriminator.

Finally, our overall loss function is defined as:

$$\mathcal{L}_{total} = \lambda_{rec}\mathcal{L}_{rec} + \lambda_{adv}\mathcal{L}_{adv} , \quad (4)$$

where λ_{rec} and λ_{adv} are the hyper-parameters to balance the weights of reconstruction loss and adversarial loss. During training, the generator is optimized by minimizing $\mathcal{L}_{total}$ and the discriminator is optimized by maximizing $\mathcal{L}_{adv}$.

Multi-scale Aggregated Inference

In testing, we adopt multiple masks with different locations and scales to remove different parts of regions from the input images, and merge the multiple outputs from our SCADN to compute the final error map. First, the output result $\hat{\mathbf{I}}_{i,j}$ is computed by:

$$\hat{\mathbf{I}}_{i,j} = G(\mathbf{I} \odot \mathbf{M}_{i,j}) , \quad (5)$$

where G is the generator network of our SCADN, $\odot$ denotes the element-wise multiplication, and $\mathbf{M}_{i,j}$ represents the mask with j-th type (horizontal or vertical stripes & different mask regions) and i-th scale. The sampled masks are shown in Fig. 3 for reference. Then, the error map $\mathbf{E}_{i,j}$ is defined as the square value of the difference between the input image and the output result of the network, and only the pixel values at the mask region $(\mathbf{1}-\mathbf{M}_{i,j})$ is considered:

$$\mathbf{E}_{i,j} = (\hat{\mathbf{I}}_{i,j}-\mathbf{I}) \odot (\hat{\mathbf{I}}_{i,j}-\mathbf{I}) \odot (\mathbf{1}-\mathbf{M}_{i,j}) . \quad (6)$$

As shown in Fig. 4, we have four types of masks with different scales; see Fig. 3 for these masks. Then, we adopt the maximum value at each position of the error maps as the final result $\mathbf{E}_i$ for the i-th scale:

$$\mathbf{E}_i = max(\mathbf{E}_{i,0}, \mathbf{E}_{i,1}, \mathbf{E}_{i,2}, \mathbf{E}_{i,3}) . \quad (7)$$

To merge the error maps at different scales, we present to choose one error map from them, which has the value that is far from the normal samples in the training set. To do so, we first compute the average value of the error maps with different scales on the training set as a reference value for each scale, which is denoted as μ_i. Then, we adopt the error map with the largest distance from μ_i as the final error map:

$$\mathbf{E}_{final} = \mathbf{E}_f , \quad (8)$$

$$f = \underset{i}{argmax}\,(\text{average}(\,\mathbf{E}_i\,) - \mu_i) . \quad (9)$$

The final score is defined as the mean value of $\mathbf{E}_{final}$:

$$S = \text{average}(\,\mathbf{E}_{final}\,) . \quad (10)$$

The inference strategy is illustrated in Fig. 4. The large scale mask helps to discover the big abnormal region while the small scale mask helps to find the tiny abnormal region. By merging the results produced from multiple scales and multiple locations, we can fully explore the semantic context from the input image, thus improving the overall performance for anomaly detection.

LaceAD Dataset

Due to the lack of real-world datasets for anomaly detection, the previous works use the classification datasets to evaluate the performance of anomaly detection. They adopt one or several categories in the classification dataset as normal, and treat the remaining categories as abnormal. However, the categories in the classification datasets are largely different, and the data samples are not common for the real-world anomaly detection, where the difference between the normal and abnormal samples is small; see Fig. 1 as an example. To solve the above issues, a real-world dataset MVTecAD is prepared by (Bergmann et al. 2019), which contains over $8,000$ images with common objects, such as bottles, toothbrushes, and cables.

To further enrich the real-world dataset for anomaly detection, in this work, we build a new dataset on lace fabrics, which is denoted as *LaceAD*. To build this dataset,

(i) we collected 9,176 images from the top 10 lace fabric manufacturing companies worldwide, where the images are captured in the real production environment by a high-resolution DSLR camera; (ii) we split them into 17 subsets based on their patterns; (iii) we cropped the raw images into multiple 512×512 patches; (iv) we hired the professional workers to classify these patches into normal and abnormal sets. Samples in our dataset are shown in Fig. 5 and we follow the MvtecAD dataset to organize the structure of our dataset. Note that the training set contains only normal samples, and the testing set has normal and abnormal samples.

Experimental Results

Implementation Details

We describe the implementation in details in this subsection. We trained and tested the network on a single NVIDIA RTX2080Ti with 64GB RAM on the Ubuntu16.04 system. Our code were implemented based on PyTorch 1.10 framework with Python 3.6; The generator contains an encoder that down-samples the input image three times, followed by eight residual blocks (He et al. 2016) and a decoder that up-samples features back to the original size. We replace plain convolutions in the first residual block with dilated convolutions. Instance normalization (Ulyanov, Vedaldi, and Lempitsky 2017) is adopted in all layers of the generator. For the discriminator, we use PatchGAN (Isola et al. 2017) with spectral normalization (Miyato et al. 2018).

For the training settings, the network was optimized by an Adam optimizer (Kingma and Ba 2015) with $\beta_1 = 0$, $\beta_2 = 0.9$, and learning rate as 0.0001. The network stopped the training after 200 epochs. The weights in the loss function were set as $\lambda_{rm} = 3$, $\lambda_{rec} = 1$ and $\lambda_{adv} = 0.001$.

Datasets and Metrics

We evaluate the proposed framework on four datasets: our newly collected LaceAD dataset, MVTec AD (Bergmann et al. 2019), CIFAR-10 (Krizhevsky and Hinton 2009), and MINST (LeCun 1998). Note that CIFAR-10 and MINST are built for image classification and we follow the previous works to use it for anomaly detection. It is worth to be mentioned that each subset with different patterns in LaceAD and MVTec AD is evaluated separately, and we compute the average value of these subsets to evaluate the performance of the anomaly detection algorithms. For MVTecAD and LaceAD datasets, we resize both training and testing images to the size of 512×512, and each batch of training contains four images; For CIFAR and MINST datasets, we resize the image to the size of 32×32, and set the batch size to 512.

Following the previous works (Perera, Nallapati, and Xiang 2019; Gong et al. 2019), we adopt the Area Under the Curve (AUC) of Receiver Operating Characteristics (ROC) curve to quantitatively evaluate the performance of different methods for anomaly detection.

Experiments on Real-World Datasets

Compared Methods We compare the proposed method with several recent methods: Deep autoencoder (DAE), ALOCC (Sabokrou et al. 2018), OCGAN (Perera, Nallapati, and Xiang 2019), and MemAE (Gong et al. 2019). All of them are deep-learning-based methods. For a fair comparison, we try our best to implement these methods and carefully finetune the training parameters to obtain the best results. For DAE, we removed the discriminator and residual blocks in our generator, only reserved the encoder-decoder structure. For MemAE, we further added the memory module between encoder and decoder. The image size used in DAE, MemAE, and our method was set as 512×512. We trained OCGAN following its official code released on the website. Due to the limitation of the network setting in their code, we set the image size used in OCGAN as 509×509, which is close to 512×512.

Pattern	ALOCC D	ALOCC DR	DAE	OCGAN	MemAE	Ours
1	0.929	**0.986**	0.604	0.648	0.646	0.975
2	**0.847**	0.825	0.638	0.725	0.833	0.843
3	0.870	0.635	0.878	0.380	0.969	**0.984**
4	0.924	0.829	0.810	0.648	0.921	**0.956**
5	0.470	0.367	0.583	0.689	0.810	**0.927**
6	0.821	0.702	0.930	0.898	0.811	**0.949**
7	0.460	0.677	0.647	0.578	0.604	**0.731**
8	0.730	**0.730**	0.478	0.415	0.500	0.660
9	0.807	0.717	0.746	0.671	0.806	**0.935**
10	0.886	0.938	**1.000**	0.937	0.821	**1.000**
11	0.106	0.950	**0.992**	0.935	0.956	0.987
12	0.576	0.499	0.568	0.443	0.648	**0.762**
13	0.799	0.708	0.898	0.840	0.834	**0.998**
14	0.618	0.637	0.686	0.488	0.591	**0.800**
15	0.511	0.791	0.734	0.733	0.729	**0.889**
16	0.930	0.965	0.825	0.816	0.921	**0.998**
17	0.503	0.478	0.761	0.329	0.460	**0.866**
Mean	0.693	0.731	0.752	0.657	0.756	**0.898**

Table 1: Anomaly detection on the LaceAD dataset.

Quantitative comparison. Table 1 shows the AUC results of anomaly detection on different subsets with different patterns in our LaceAD. Compared with other methods, our method has achieved obvious advantages in most patterns. This is because lace has complex patterns, and its anomaly detection requires contextual information in many cases, which can be extracted by our method. For example, holes are very common in lace images, some of them are contained in the pattern itself, while others are abnormal holes. By exploring the semantic context, we are easy to find this region is different from the surroundings.

Table 2 shows the AUC results of anomaly detection on different category subsets in MVTec AD. Our method has a significantly better result on the mean value and outperforms the compared methods in most categories, especially on the Grid and Transistor, which has complex structures.

Since the MVTec AD dataset contains ground truth images of abnormal regions, it is also valuable to measure performance of the segmentation capability of abnormal re-

Category	ALOCC D	ALOCC DR	DAE	OCGAN	MemAE	Ours
Bottle	0.421	0.460	0.860	0.592	0.930	**0.957**
Cable	0.611	0.531	0.648	0.496	0.785	**0.856**
Capsule	0.711	0.487	0.534	0.714	0.735	**0.765**
Carpet	**0.614**	0.423	0.588	0.348	0.386	0.504
Grid	0.396	0.781	0.858	0.855	0.805	**0.983**
Hazelnut	0.449	**0.993**	0.513	0.753	0.769	0.833
Leather	0.488	**0.768**	0.497	0.624	0.423	0.659
Metal nut	0.749	0.705	**0.793**	0.295	0.654	0.624
Pill	0.781	0.726	0.693	0.702	0.717	**0.814**
Screw	**1.000**	0.995	0.719	0.505	0.257	0.831
Tile	0.389	0.526	**0.894**	0.806	0.718	0.792
Toothbrush	0.567	0.642	0.942	0.594	0.967	**0.981**
Transistor	0.195	0.751	0.376	0.477	0.791	**0.863**
Wood	0.319	0.279	0.882	0.959	0.954	**0.968**
Zipper	0.456	0.547	0.819	0.364	0.710	**0.846**
Mean	0.543	0.641	0.707	0.606	0.707	**0.818**

Table 2: Anomaly detection on the MVTec AD dataset.

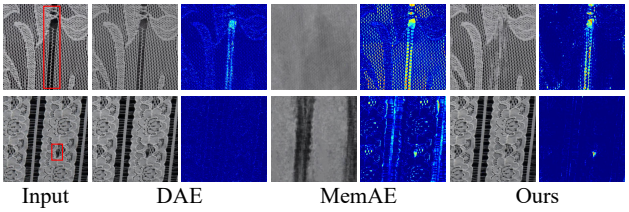

Figure 6: Visualization of the output images and error maps.

gions. The anomaly ground truth is a binary mask that equal to one at abnormal region and zero otherwise. We use error maps of different methods to calculate the AUC of all the pixels over each image, and then compute the mean value of each category for comparison. As shown in Table 3, our method outperforms the compared methods on most of the categories with a significant superiority. Although our method is not designed for abnormal region segmentation, this experimental result proves that the error map obtained by our method is meaningful, indicating that our method can successfully find the abnormal regions from the inputs.

Visual comparisons. To compare the reconstruction results, we visualize the output images and error maps produced from DAE, MemAE, and our method in Fig. 6 and Fig. 7. For fair comparisons, we normalized each error map by dividing its maximum value for the visualization. From the results, we can see that our method generally produces more reasonable error maps that indicate the anomaly regions and our output images are more similar to the normal samples. This proves that our method can explore the semantic context from normal patterns and focus on the discriminative regions. As shown in the first lace sample in Fig. 6, the long black hole is appeared in the abnormal sample. As the black hole shares the local content with the normal holes, DAE keeps the black color and cannot conduct high error value for this abnormal region. A similar phenomenon also appears in the second lace sample. Although MemAE conducts a large error over the anomaly regions, their reconstructed results are blurry, thus failing to localize the anomaly region. In contrast, our method can explore the semantic context to complete the black regions of the input image. As a result, our method has lower error on the normal regions, which avoids mis-classifying the normal samples. Similarly, our method can successfully generate a reasonable error map for the screw sample in Fig. 7, but other methods may fail.

Category	DAE	OCGAN	MemAE	Ours
Bottle	0.544	0.567	**0.724**	0.696
Cable	0.535	0.564	0.814	**0.814**
Capsule	0.542	0.637	0.673	**0.687**
Carpet	0.528	0.546	0.574	**0.649**
Grid	0.550	0.652	0.468	**0.796**
Hazelnut	0.664	0.841	0.846	**0.884**
Leather	**0.783**	0.749	0.686	0.763
Metal nut	0.539	0.534	**0.769**	0.754
Pill	0.555	0.596	0.737	**0.747**
Screw	0.570	0.708	0.732	**0.876**
Tile	0.630	0.592	0.647	**0.677**
Toothbrush	0.616	0.763	0.886	**0.901**
Transistor	0.532	0.582	**0.714**	0.689
Wood	0.612	0.655	0.652	**0.672**
Zipper	0.536	0.624	0.643	**0.670**
Mean	0.582	0.641	0.704	**0.752**

Table 3: Anomaly region detection in terms of mean AUC on the MVTec AD dataset.

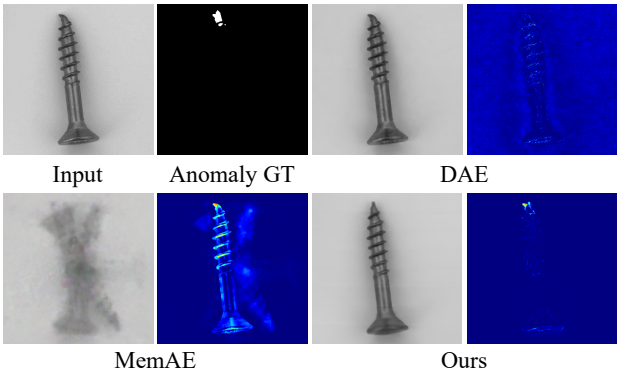

Figure 7: Visualization of the output images and error maps.

Ablation study. To analyze the effectiveness of each component in the proposed network, we perform an ablation experiment on the MVTec AD dataset and report the results in Table 4. In this table, we use ✓ to indicate which component in our network design is reserved. Therefore, the last row with all items marked with ✓ indicates our full pineline. In the first column "L_{adv}", ✓ indicates we

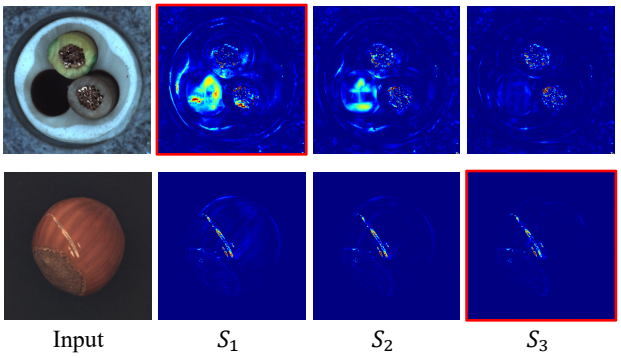

Figure 8: Visualization of the output images and error maps.

$\mathcal{L}_{adv}$	S_1	S_2	S_3	Merge type	sub μ_i	Mean AUC
	✓	✓	✓	max	✓	0.8154
✓	✓			-		0.8101
✓		✓		-		0.8019
✓			✓	-		0.7901
✓	✓	✓	✓	mean		0.8083
✓	✓	✓	✓	max		0.8136
✓	✓	✓	✓	max	✓	**0.8183**

Table 4: Ablation study on the MVTec AD dataset.

adopt the discriminative loss as well as the reconstruction loss in Equation 4. In the second to fourth columns, we use S_1, S_2, and S_3 to indicate the mask with the large, middle, and small scales, respectively; see Fig. 3 for these masks with different scales. In the fifth column, "max" indicates we merge the error maps by adopting the maximum value as shown in Equation 7, and "mean" indicates we adopt the mean value instead. In the sixth column, sub μ_i means we adopt average($\mathbf{E}_i$) $-$ μ_i in Equation 9, otherwise, we use average($\mathbf{E}_i$) to replace this equation.

The results in Table 4 show that our network with all the components produce the best results. In addition, by comparing the results of the first row and last row, we can see that the discriminator loss helps to improve the results; By comparing the results in the second and sixth rows, we can see that aggregating multi-scale results can improve the performance by a large margin. Moreover, the comparison between the results reported in the last two rows show that, utilizing the reference value from training set to compute the error map also benefits to the results. We also visualize the scale-selection results of two abnormal samples; please refer to Fig. 8 for reference.

Experiments on Classification Datasets

CIFAR-10 and MINST are collected for the image classification task, and we followed the previous work (Perera, Nallapati, and Xiang 2019) to iteratively set one class in this dataset as the normal samples and treat others as abnormal samples. Following (Perera, Nallapati, and Xiang 2019), we used the original training-testing splits of CIFAR-10 and MINST to conduct training, where only the normal samples were used. For a fair comparison, we reported the results of other methods from the paper of OCGAN (Perera, Nallapati, and Xiang 2019), and reported the average results of our method that was tested three times with different seeds. The results are shown in Table 5, where our method clearly outperforms the others, which shows the generalization capability of the proposed algorithm.

	MINST	CIFAR-10
OCSVM	0.9513	0.5856
KDE	0.8143	0.6097
DAE	0.8766	0.5358
VAE	0.9696	0.5833
Pix CNN	0.6183	0.5506
GAN	0.8662	0.5916
AND	0.9671	0.6172
AnoGAN	0.9127	0.6179
DSVDD	0.9480	0.6481
OCGAN	0.9750	0.6566
Ours	**0.9771±0.0005**	**0.6690±0.0019**

Table 5: Experiments on the MINST & CIFAR-10.

Conclusion

We present a novel approach to learn the semantic context from normal samples for unsupervised anomaly detection. Our key observation is that the abnormal regions usually have different semantic patterns from the surrounding regions. To this end, we present a deep neural network architecture to learn the semantic context by designing multi-scale striped masks to remove a part of regions from the normal samples and reconstructing the missing regions to match to the input images. In testing, we infer the abnormal samples based on the error maps, which are computed as the difference between the reconstructed images and the input images for both normal and abnormal samples. Furthermore, we construct a new dataset, i.e., LaceAD, for real-world unsupervised image anomaly detection, which includes $9,176$ high-resolution lace fabric images that belong to 17 patterns. Finally, we test our method on three public benchmark datasets and LaceAD dataset, compare our network with various methods and show its superiority over the state-of-the-art methods. Our method largely outperforms the compared methods on LaceAD dataset and achieves 89.8% on AUC evaluation, indicating the effectiveness of our work in industry environment.

Acknowledgements

The work was supported by Key-Area Research and Development Program of Guangdong Province, China (2020B010166003, 2020B010165004, 2018B010107003), Guangdong High-level personnel program (2016TQ03X319), NSFC (61772206, U1611461, 61472145), and CUHK Direct Grant for Research 2018/19.

References

Abati, D.; Porrello, A.; Calderara, S.; and Cucchiara, R. 2019. Latent Space Autoregression for Novelty Detection. In *Proc. CVPR*, 481–490.

Ahmed, F.; and Courville, A. 2020. Detecting semantic anomalies. In *Proc. AAAI*, volume 34, 3154–3162.

Akcay, S.; Atapour-Abarghouei, A.; and Breckon, T. P. 2018. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Proc. ACCV*, 622–637.

Bergmann, P.; Fauser, M.; Sattlegger, D.; and Steger, C. 2019. MVTec AD–A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In *Proc. CVPR*, 9592–9600.

Bergmann, P.; Fauser, M.; Sattlegger, D.; and Steger, C. 2020. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proc. CVPR*, 4183–4192.

Candès, E. J.; Li, X.; Ma, Y.; and Wright, J. 2011. Robust principal component analysis? *Journal of the ACM* 58(3): 11.

Chalapathy, R.; Menon, A. K.; and Chawla, S. 2018. Anomaly detection using one-class neural networks. *arXiv preprint arXiv:1802.06360* .

Chen, Y.; Zhou, X. S.; and Huang, T. S. 2001. One-class SVM for learning in image retrieval. In *Proc. ICIP*, 34–37.

Cong, Y.; Yuan, J.; and Liu, J. 2011. Sparse reconstruction cost for abnormal event detection. In *Proc. CVPR*, 3449–3456.

Golan, I.; and El-Yaniv, R. 2018. Deep anomaly detection using geometric transformations. In *Proc. NeurIPS*, 9758–9769.

Gong, D.; Liu, L.; Le, V.; Saha, B.; Mansour, M. R.; Venkatesh, S.; and Hengel, A. v. d. 2019. Memorizing Normality to Detect Anomaly: Memory-Augmented Deep Autoencoder for Unsupervised Anomaly Detection. In *Proc. ICCV*, 1705–1714.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Proc. NeurIPS*, 2672–2680.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proc. CVPR*, 770–778.

Isola, P.; Zhu, J.-Y.; Zhou, T.; and Efros, A. A. 2017. Image-to-image translation with conditional adversarial networks. In *Proc. CVPR*, 1125–1134.

Kim, J.; and Grauman, K. 2009. Observe locally, infer globally: a space-time MRF for detecting abnormal activities with incremental updates. In *Proc. CVPR*, 2921–2928.

Kim, J.; and Scott, C. D. 2012. Robust kernel density estimation. *Journal of Machine Learning Research* 13(Sep): 2529–2565.

Kingma, D. P.; and Ba, J. 2015. Adam: A method for stochastic optimization. In *Proc. ICLR*.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. Technical report, Citeseer.

LeCun, Y. 1998. The MNIST database of handwritten digits. *http://yann. lecun. com/exdb/mnist/* .

Liu, G.; Reda, F. A.; Shih, K. J.; Wang, T.-C.; Tao, A.; and Catanzaro, B. 2018. Image inpainting for irregular holes using partial convolutions. In *Proc. ECCV*, 85–100.

Liu, W.; Li, R.; Zheng, M.; Karanam, S.; Wu, Z.; Bhanu, B.; Radke, R. J.; and Camps, O. 2020. Towards visually explaining variational autoencoders. In *Proc. CVPR*, 8642–8651.

Lu, C.; Shi, J.; and Jia, J. 2013. Abnormal event detection at 150 fps in matlab. In *Proc. ICCV*, 2720–2727.

Miyato, T.; Kataoka, T.; Koyama, M.; and Yoshida, Y. 2018. Spectral Normalization for Generative Adversarial Networks. In *Proc. ICLR*.

Nazeri, K.; Ng, E.; Joseph, T.; Qureshi, F.; and Ebrahimi, M. 2019. Edgeconnect: Generative image inpainting with adversarial edge learning. *arXiv preprint arXiv:1901.00212* .

Oza, P.; and Patel, V. M. 2019. Active Authentication using an Autoencoder regularized CNN-based One-Class Classifier. In *Proc. FG*, 1–8.

Parzen, E. 1962. On estimation of a probability density function and mode. *The annals of mathematical statistics* 33(3): 1065–1076.

Perera, P.; Nallapati, R.; and Xiang, B. 2019. Ocgan: One-class novelty detection using gans with constrained latent representations. In *Proc. CVPR*, 2898–2906.

Perera, P.; and Patel, V. M. 2019. Learning deep features for one-class classification. *IEEE TIP* 28(11): 5450–5463.

Pidhorskyi, S.; Almohsen, R.; and Doretto, G. 2018. Generative probabilistic novelty detection with adversarial autoencoders. In *Proc. NeurIPS*, 6822–6833.

Ruff, L.; Vandermeulen, R.; Goernitz, N.; Deecke, L.; Siddiqui, S. A.; Binder, A.; Müller, E.; and Kloft, M. 2018. Deep one-class classification. In *Proc. ICML*, 4393–4402.

Sabokrou, M.; Khalooei, M.; Fathy, M.; and Adeli, E. 2018. Adversarially learned one-class classifier for novelty detection. In *Proc. CVPR*, 3379–3388.

Schlegl, T.; Seeböck, P.; Waldstein, S. M.; Schmidt-Erfurth, U.; and Langs, G. 2017. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*, 146–157.

Schölkopf, B.; Platt, J. C.; Shawe-Taylor, J.; Smola, A. J.; and Williamson, R. C. 2001. Estimating the support of a high-dimensional distribution. *Neural computation* 13(7): 1443–1471.

Tax, D. M.; and Duin, R. P. 2004. Support vector data description. *Machine learning* 54(1): 45–66.

Tian, K.; Zhou, S.; Fan, J.; and Guan, J. 2019. Learning competitive and discriminative reconstructions for anomaly detection. In *Proc. AAAI*, volume 33, 5167–5174.

Ulyanov, D.; Vedaldi, A.; and Lempitsky, V. 2017. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proc. CVPR*, 6924–6932.

Vincent, P.; Larochelle, H.; Bengio, Y.; and Manzagol, P.-A. 2008. Extracting and composing robust features with denoising autoencoders. In *Proc. ICML*, 1096–1103.

Wang, J.; and Cherian, A. 2019. GODS: Generalized One-Class Discriminative Subspaces for Anomaly Detection. In *Proc. ICCV*, 8201–8211.

Xia, Y.; Cao, X.; Wen, F.; Hua, G.; and Sun, J. 2015. Learning discriminative reconstructions for unsupervised outlier removal. In *Proc. ICCV*, 1511–1519.

Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; and Huang, T. S. 2018. Generative image inpainting with contextual attention. In *Proc. CVPR*, 5505–5514.

Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; and Huang, T. S. 2019. Free-form image inpainting with gated convolution. In *Proc. ICCV*, 4471–4480.

Zhai, S.; Cheng, Y.; Lu, W.; and Zhang, Z. 2016. Deep Structured Energy Based Models for Anomaly Detection. In *Proc. ICML*, 1100–1109.

Zhao, B.; Fei-Fei, L.; and Xing, E. P. 2011. Online detection of unusual events in videos via dynamic sparse coding. In *Proc. CVPR*, 3313–3320.

Zheng, C.; Cham, T.-J.; and Cai, J. 2019. Pluralistic Image Completion. In *Proc. CVPR*, 1438–1447.

Zhou, C.; and Paffenroth, R. C. 2017. Anomaly detection with robust deep autoencoders. In *Proc. SIGKDD*, 665–674.

Zong, B.; Song, Q.; Min, M. R.; Cheng, W.; Lumezanu, C.; Cho, D.; and Chen, H. 2018. Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection. In *Proc. ICLR*.