Article

Self-Attention Autoencoder for Anomaly Segmentation

Yang Yang [1,*] 0000-0002-6841-8644

[1] School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China; tony0821@shu.edu.cn
* Correspondence: tony0821@shu.edu.cn

Abstract: Anomaly detection and segmentation aim at distinguishing abnormal images from normal images and further localizing the anomalous regions. Feature reconstruction based method has become one of the mainstream methods for this task. This kind of method has two assumptions: (1) The features extracted by neural network is a good representation of the image. (2) The autoencoder solely trained on the features of normal images cannot reconstruct the features of anomalous regions well. But these two assumptions are hard to meet. In this paper, we propose a new anomaly segmentation method based on feature reconstruction. Our approach mainly consists of two parts: (1) We use a pretrained vision transformer (ViT) to extract the features of the input image. (2) We design a self-attention autoencoder to reconstruct the features. We regard that the self-attention operation which has a global receptive field is beneficial to the methods based on feature reconstruction both in feature extraction and reconstruction. The experiments show that our method outperforms the state-of-the-art approaches for anomaly segmentation on the MVTec dataset. It is both effective and time-efficient.

Keywords: anomaly detection; anomaly segmentation; self-attention; transformers; autoencoders

1. Introduction

Anomaly detection aims at distinguishing abnormal images from normal images. Due to the lack of anomaly images and the diversity of anomalies, only defect-free images are used during training. In other words, the training data defines what is normal, and those who deviate from normal distribution can be judged as anomalous. It is also known as one-class classification, because it actually tries to find the decision boundary of training data. Nowadays, image-level anomaly detection is not enough. We want to further localize the anomalous regions in pixel-level for better interpretability, which is known as anomaly segmentation. Anomaly segmentation in natural images is an important task in computer vision. It has a wide range of applications in manufacturing, medicine and security.

In recent years, a lot of anomaly segmentation methods have been proposed [1–8]. The reconstruction-based method [1–3,5] is one of the most commonly used methods. This kind of approaches uses an autoencoder to reconstruct the input image at training time. At test time, the anomaly score map can be calculated by comparing the difference between the input image and the reconstructed image. These methods suppose that the autoencoder solely trained on normal images can not reconstruct anomalous regions well. So anomalous areas will have relatively high reconstruction error. Pixel-level anomaly segment also can be obtained easily via anomaly score map. This method requires the autoencoder to have a special reconstruction ability that it can reconstruct normal images well, but it is not good at reconstructing anomalous regions, which means the reconstruction of anomaly regions does not look like the input ones.

Recently, compared to reconstruct the original image, reconstruction in feature space has reached better results [3] and has the potential to become a new paradigm. This kind of method uses pretrained deep convolution neural networks (CNN) [23] to extract the features of the input image. Then it uses autoencoder to reconstruct the feature representation to get anomaly segmentation on the feature map. Finally, the anomaly score map are upsampled to the size of the input image to get pixel-level segmentation. It assumes that the autoencoder solely trained on the representation of normal image

Citation: Yang, Y. Self-Attention Autoencoder for Anomaly Segmentation. *Preprints* **2021**, *1*, 0. https://doi.org/

Received:
Accepted:
Published:

 doi:10.20944/preprints202108.0570.v1

cannot reconstruct the representation of anomalous region well in the feature space. It also requires that the feature map has spatial correspondence to the original image, so anomaly segmentation of the whole image can come from the anomaly segmentation on the feature map. The biggest advantage of feature the method based on feature reconstruction is that the features extracted by the CNN pretrained on large datasets contain enough semantic information which is beneficial to the anomaly detection task. In addition, this method aggregates the feature maps generated from different convolution layers together to build a dense regional feature representation. Since the receptive field of the convolution kernel increases from shallow layers to deep layers, the stacked feature maps contain a multiscale description of the original image. Thus, it is a better choice to reconstruct in feature space than reconstruct in image space.

To improve the performance of feature reconstruction based method, there are two important parts: (1) construct a good feature space (2) build an effective autoencoder. A strong feature extractor should be pretrained on a large dataset and provide enough features with semantic information which can represent the input image and be transferred to anomaly detection task well. An effective autoencoder should be able to reconstruct normal representation well and suppress the reconstruction quality of the representation of the anomalous region. For the above two points, we choose the vision transformer (ViT) [9] as the feature extractor and design a self-attention autoencoder based on the transformer architecture for feature reconstruction, which has stronger reconstruction ability for anomaly detection than a normal autoencoder.

In this paper, we propose a new anomaly segmentation method based on feature reconstruction with a self-attention autoencoder(SAAE). Firstly, we use ViT to extract the features of the image. This is a way to use pretrained models. Many works used model pretrained on large datasets in recent years, such as [3,4,6,7]. These models are trained for large-scale image classification task and thus will produce discriminative features. Anomaly detection task is a one-class classification task, so these discriminative features which contain high-level semantic information are beneficial to anomaly detection and can be transferred easily. Compared to CNN, more global context information will be retained in the features extracted by ViT. At the same time, similar to CNN, each location on the feature map of ViT perceives a corresponding spatial region of the input image. Then we aggregate feature maps from different ViT layers to build a dense hierarchical feature representation.

Then, the hierarchical feature maps will pass through a self-attention autoencoder. The self-attention autoencoder will reconstruct the input feature maps. Finally, we compare the input feature maps and the reconstructed feature maps to get an anomaly score map and further upsample it to the size of the original image to get the anomaly segmentation of the whole image.

The self-attention autoencoder consists of self-attention encoder and self-attention decoder. The self-attention encoder is composed of a convolutional encoder and a transformer encoder. The self-attention decoder is composed of a transformer decoder and a convolutional decoder. It is actually a transformer operates in the latent space of a convolutional autoencoder. The self-attention encoder first does dimension reduction with convolution to extract low-level local features. Then the self-attention operation will contact the information of each point on the hierarchical feature maps and fuse global semantic information. After that, the self-attention decoder reconstructs the feature maps according to the global context information and ascend the dimension with convolution layers. The decoding process is parallel, which is more efficient than the initial transformer [10].

There are two reasons why self-attention autoencoder is more suitable for anomaly segmentation than ordinary autoencoders. Firstly, it can reconstruct normal regions well. The convolutional layers extract local and low-level features, and the self-attention layers extract global and high-level semantic information. With a combination of local and global information, the self-attention autoencoder has enough ability to reconstruct normal images well. More importantly, it can suppress the reconstruction of anomalous regions to a certain

 doi:10.20944/preprints202108.0570.v1

extent. In contrast to the locality of convolution, self-attention has global receptive field. Some abnormal regions can be reconstructed well solely with local features, but global receptive field of the self-attention force the autoencoder to consider larger areas which are mostly normal. Thus, the self-attention autoencoder which is trained solely on the normal images can reconstruct the anomalous regions to look like normal ones with the connection between each pixel. This ability can be regarded as inpainting ability and it is suitable for anomaly detection. Our experiment has proved the effect of the self-attention autoencoder.

Our anomaly segmentation method is very effective and time-efficient. The experiment shows our approach outperforms the current state-of-the-art methods on the MVTec AD dataset. It indicates that the self-attention autoencoder has a strong capability for the anomaly detection problem.

2. Related Works

2.1. Anomaly Detection Methods

Reconstruction-based methods [1–3,5] are one of the most commonly used anomaly detection and segmentation methods over the past few years. They train autoencoders (AE) [1] to reconstruct normal images. At test time, We can compare pixel-wise reconstruction error between the input image and the reconstructed image to get anomaly segmentation. These methods assume that the autoencoder solely trained on normal images can not reconstruct anomalous regions well. So anomalous regions will have relatively high reconstruction error. The reconstruction error can be measured by L2 distance or SSIM [12]. We can also use generative adversarial networks (GAN) [22] to do the same thing. There is a variant of reconstruction-based methods, called restoration-based methods [5,8]. This kind of method applies some argumentations to images, like or random erasing [5], graying [8] or random rotation [8], to erase some attributes related to semantic information. Then they train AE to restore the information. Their assumption is similar to reconstruction-based methods. They think it is hard to restore the attributes related to semantic information in anomalous images. It is a way to force the autoencoder to learn semantic information.

Another important trend of anomaly detection is to use pretrained models. This kind of method uses neural networks pretrained on large-scale dataset to extract features and model the distribution of normal features. The most popular pretrained model is convolution neural networks [23] pretrained on ImageNet [13] classification task. The anomaly detection methods using pretrained model are much better than those who purely train on normal dataset. The typical methods include Uninformed Students [7], Patch SVDD [4] and PaDiM [6]. These methods usually divided images into patches, extract features of patches and build a classification model with machine learning algorithms. The key to those methods is to have a good feature representation of images or patches. In other words, you need to choose a suitable feature space that can distinguish between normal and abnormal samples. CNN pretrained on large datasets for classification is useful because it can produce enough discriminative features which contains high-level semantic information. So it is reasonable that the methods using pretrained model perform much stronger than those who solely trained on small datasets which cannot extract rich features. From this point of view, it is essential for future anomaly detection methods to use pretrained models.

Reconstruction in feature space combines the advantages of reconstruction based methods and pretrained model based methods. This framework is proposed by [3], called deep feature reconstruction (DFR). DFR first uses pretrained CNN to extract features of the image. Then it aligns and aggregates the output feature maps of different convolution layers to get a multi-scale dense regional feature representation of the image. Next, it uses a deep convolutional autoencoder with one by one kernel to reproduce the dense feature maps. Finally, it compares the feature maps and reconstructed feature maps to get the anomaly score of the feature maps and upsample it to the spatial size of the image to get pixel-wise anomaly segmentation. It leverages pretrained model to enhance the reconstruction-based method. It is also a multiscale method which can detect different size of abnormal regions

 doi:10.20944/preprints202108.0570.v1

because the receptive field of convolution kernels increase from shallow layer to deep layer. The anomaly segmentation can be got from the anomaly score of the feature maps because convolution keeps the spatial relationship and each point on the feature maps corresponds to a spatial region of the image. Based on these advantages, so it is better than reconstruction in image space is much better than reconstruction in image space. This framework is pretty good, but the component still needs to be discussed. The upper limit of this framework is related to the feature extractor and the autoencoder. Convolutional autoencoder has strong reconstruction ability,but in many cases, the anomalous region can also be reconstructed well, which is not good for anomaly detection. Compared to DFR, our approach mainly has two improvements: constructing a better feature space and design a self-attention autoencoder that is suitable for anomaly detection and segmentation.

2.2. *Visual Transformers*

We introduce transformer into the anomaly detection task. Transformer [10] is initially designed for natural language processing (NLP) tasks and makes a great success. Transformer based models like BERT [11], GPT [14] dominate many aspects of NLP. Recently, transformer has been extended into vision tasks which shows competitive and even better performance on plenty of tasks compared with CNN. Now it has reached the state of art result in image classification, object detection, semantic segmentation, and so on. DETR [21] is a simple end-to-end object detection framework. It is a hybrid structure of CNN and transformer. It is an early attempt for visual transformer and attracts great attention. ViT is a pure transformer model proposed by [9]. It has reached state-of-the-art performance in image classification. IPT is proposed by [16] for multiple low-level vision tasks, such as denoising and super solution. TransGAN [24] is a pure transformer version of GAN, which produces a comparable result to state-of-the-art CNN-based GAN in image generation. In computer vision, the dominant architecture of neural networks is CNN. Now transformer has the potential to be an alternative to CNN with a larger receptive field and less inductive bias. Inspired by the success of visual transformers, we try to apply the transformer to anomaly detection.

Self-attention is the basic operation of the transformer. It is an attention mechanism that relates different positions of a sequence to compute a representation of the sequence. In contrast to the locality of convolution, self-attention use global information to extract features, which can handle long-distance dependency well. In computer vision tasks, this characteristic will capture the relation of distant pixels in an image and enhance high-level semantic features. In anomaly detection, we speculate that self-attention based autoencoder could reconstruct normal images well and reconstruct the anomalous regions like the normal part due to larger receptive and the interaction of global pixels which is mostly normal ones.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2021 doi:10.20944/preprints202108.0570.v1

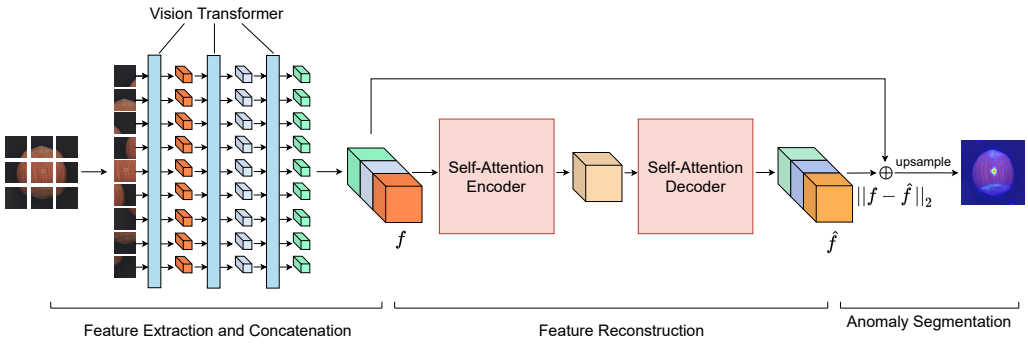

Figure 1. This is an overview of our anomaly segmentation method. Firstly, we use a vision transformer (ViT) to extract the feature of the input image. Then we reshape the output features of each ViT layer from 1D sequence into 2D feature map and concatenate the feature maps from different layers to get a dense hierarchical feature representation f. After that, we use a self-attention autoencoder to compress the feature representation and reconstruct it. Finally, we compute the L_2 loss between the feature maps f and the reconstruction $\hat{f}$ to get the anomaly score map and upsample it to the size of the input image. We can define a threshold T to binarize the anomaly map to get the anomaly segmentation of the whole image.

3. Method

Fig 1 shows the overview structure of our method. It can be divided into three steps. The first step is using pretrained ViT to extract features and stack feature maps together to become a dense hierarchical feature representation. The second step is using self-attention to reconstruct the dense feature representation. The third step is comparing the dense feature representation with its reconstruction to get anomaly score map, further upsample the anomaly map to the size of the input image and get anomaly segmentation of the whole image with a threshold. The details of the three steps will be described in the following paragraphs.

3.1. ViT Feature Extraction and Concatenation

We use pretrained ViT [9] as a feature extractor. ViT is pure transformer model designed for image classification. It follows the structure of the original transformer [10] as closely as possible. Since transformer receives a sequence as input, they split an image into fixed-size patches and linearly embed each of them as the input of the transformer. ViT consists of several encoders. Each encoder is composed of a multi-headed self-attention (MSA) layer and a multi-layer perceptron (MLP) layer. The output of each encoder is the feature representation of the input image. We reshape them from 1D sequence into 2D feature map and concatnate them together to get a dense hierarchical feature representation of the input image, as shown in Fig 1.

Suppose the input image has height H, width W and C channels. ViT receives 1D sequence as input, so the image $x \in \mathbb{R}^{H\times W\times C}$ is reshaped into a sequence of flattened 2D patches $x_p \in \mathbb{R}^{N\times P^2C}$, where the patch size is $P \times P$ and $N = HW/P^2$. Now we have a sequence of N patches and each patch has dimension P^2C. Then a linear projection layer will map each patch into D dimensions. After that, we get a sequence of N patches with dimension D, called patch embedding, which is the input of the transformer.

The ViT consists of M encoders. The output of each encoder is size of $N \times D$. The N patches are flattened from $2D$ images and each patches are corresponding to a region of the input image. So we could reshape the output of each encoder into 2D patches as feature maps $\{z_1(x), z_2(x), \ldots, z_M(x)\}$ where $z_i(x) \in \mathbb{R}^{H_0\times W_0\times D}$, $H_0 = H/P$ and $W_0 = W/P$. Then we concatenate all the feature maps together to get stacked feature maps

$$f_{1:M}(x) = cat(z_1(x), z_2(x), \ldots, z_M(x)) \tag{1}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2021 doi:10.20944/preprints202108.0570.v1

where $f_{1:M}(x)$ denotes that the 1th to Mth feature maps are concatnated together. The stacked feature maps are size of $W_0 \times H_0 \times U$, where $U = MD$.

With self-attention, each point on the featue map of a ViT layer comes from a weighted average of all points on the output of the last layer. This allows ViT to have less inductive bias and more freedom compare to CNN, which leads to a good representation of the input image. The feature map contains enough discriminative semantic information due to pretraining on large-scale dataset for classification task. Besides, each point on the feature map corresponds to a spatial region of the input image, so the anomaly segmentation on the feature map can be mapped to the segmentation of the whole image. In addition, the feature maps of CNN have various sizes, which needs to align manually, whereas the feature maps of ViT are naturally aligned. So feature maps can be aggregated directly and become a dense feature representation, which will avoid the inaccuracy caused by manual padding and interpolation existing in generating the dense feature representation of CNN. Finally, [15] shows that ViT is able to learn about higher level relationships very early as it uses global attention instead of local. It means we do not need to concatenate a lot of layers like CNN, which is more time-efficient.

3.2. Self-Attention Autoencoder

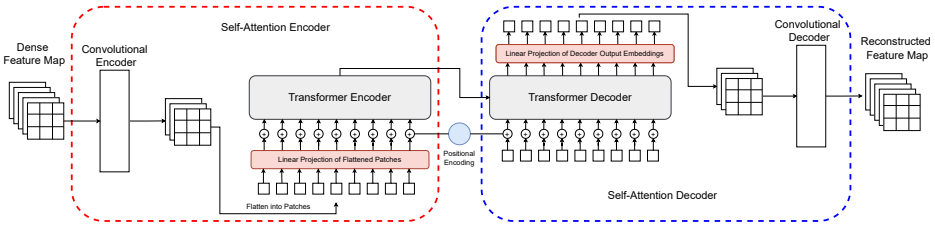

Figure 2. This is the detailed structure of the self-attention autoencoder. The input feature maps pass through a convolutional encoder with solely 1×1 kernels and is compressed spatial separately in channels. Then it is flattened into 2D patches and entered into a transformer structure. Learnable position encoding is added into the input embedding of both transformer encoder and transformer decoder. Finally, we use a convolutional decoder symmetric to the convolutional encoder to reproduce the input feature maps.

The detailed structure of the self-attention autoencoder is shown in Fig 2. In our method, only the self-attention autoencoder needs to be trained and it is trained solely on the representation of the anomaly free images. The input of the self-attention autoencoder is a dense feature representation $f(x)$ and the aim of it is to reconstruct the input. We call the reconstructed feature maps as $\hat{f}(x)$.

Self-attention encoder (SAE) consists of a convolutional encoder (CE) and a transformer encoder (TE). The input representation $f(x)$ is size of $W_0 \times H_0 \times U$. It firstly passes through the CE which consists of several convolution units. Each convolution unit consists of a convolution layer with 1×1 kernel, a batch norm layer [18] and a Rectified Linear Unit(ReLU) activation layer [20]. The output of the CE $g(x)$ is size of $W_0 \times H_0 \times V$, where $V << U$. The CE mainly compresses each point on the feature maps separately to a lower dimensions. Then the TE structure is almost the same as ViT [9]. In order to handle 2D images, we reshape the input image into a sequence of flattened 2D patches. Then these patches need to pass a trainable linear projection and be mapped to fixed E dimensions which is the constant latent vector size of the TE. The output of this projection is called the patch embeddings. Follow the configuration in [9], we add standard learnable 1D positional encodings are to the patch embeddings in order to retain positional information. The sum of patch embeddings and the positional encodings are as the input of the TE. The components of TE is the same as the standard transformer[10]. It is composed of multi-head self-attention layer (MSA) and feed forward network (FFN). The self-attention operation contacts the information between each point of the feature maps and use global

 doi:10.20944/preprints202108.0570.v1

context information to produce the output. The FFN consists of two multi-headed self-attention (MLP) layers, the input and output have the same dimension V, the inner layer has dimension L, called MLP size. In conclusion, the self-attention encoder expands ViT and add several convolution layers to do dimension reduction. Only use linear projection to do dimension reduction is also feasible, but it is too weak to capture enough information for reconstruction.

Self-attention decoder (SAD) consists of transformer decoder (TD) and convolutional decoder (CD). The TD is composed of two multi-head attention layers and an FFN. The first attention layer is self-attention. For the second attention layer, the query is from the output of the first self-attention layer, the key and value are from the output of the TE. The TD accepts three inputs: decoder input embedding, positional encoding and the output of the TE. The decoder input embedding is learnable parameter that is similar to the object query described in [21] and contains position-related information. We directly use the positional encoding of TE as the positional encoding of TD. The decoding process is parallel which is different from the original transformer because the number of output patches is fixed and the information of each decoder input embedding can be transmitted in the self-attention layer. So we use parallel decoding which is much faster than serial decoding. The main operation of the TD happens in the second attention layer. It queries necessary information from the output of the TE to reconstruct the input feature maps. We add a linear projection layer at the tail of the TD. Finally, we use a CD which has several convolutional units with one by one kernel and is symmetrical to the CE, to ascend the channels of the output of self-attention autoencoder to the same as the input feature maps. The detailed structure of the CE and CD is listed in Table 1.

Table 1. Configuration of Convolutional Encoder and Convolutional Decoder

Layer	Channels	Kernel	Stride	BN	Activation
Conv1	(U + 2 V) // 2	1×1	1	True	ReLU
Conv2	2V	1×1	1	True	ReLU
Conv3	V	1×1	1	False	/
Conv4	2V	1×1	1	True	ReLU
Conv5	(U + 2V) // 2	1×1	1	True	ReLU
Conv6	U	1×1	1	False	/

BN refers to batch normalization.

We use L_2 distance between input representation $f(x)$ and its reconstruction $\hat{f}(x)$ as loss function:

$$L(x) = \sum_{i=1}^{H_0} \sum_{j=1}^{W_0} ||f_{ij}(x) - \hat{f}_{ij}(x)||_2 \tag{2}$$

Intuitively, the CE compress each component of the feature maps and the CD rebuild them again. The transformer autoencoder make connections between each components of the feature maps. The SAD considers more about the global information due to the global receptive field of self-attention, which brings strong inpainting ability. The self-attention autoencoder is both an effective and efficient autoencoder for feature reconstruction based anomaly detection.

3.3. Anomaly Segmentation

The last step is to calculate the anomaly score map and further upsample the anomaly map to get the anomaly score map to the size of the input image. We finally need to choose a threshold to binarize the anomaly score map for pixel-wise anomaly segmentation. We first compare the difference between the image representation $f(x)$ and its reconstruction

 doi:10.20944/preprints202108.0570.v1

$\hat{f}(x)$ by calculating pair-wise reconstruction error to get the anomaly score on the feature maps:

$$A_{ij}(x) = ||f_{ij}(x) - \hat{f}_{ij}(x)||_2 \tag{3}$$

Where A(x) is the anomaly score map of the feature maps which is size of $W_0 \times H_0$. Then we bilinearly upsample the anomaly score map to the size of $W \times H$, which is the same as the input image, to get the pixel-wise anomaly score map for the input image. We assume that the self-attention autoencoder solely trained on the representation of normal images is unable to reconstruct the regional features of anomaly images. So the anomaly regions will correspond to large reconstruction errors.

We use the same method as described in [3] to decide a threshold T for anomaly segmentation, which uses the acceptable false positive rate (FPR) on the normal images to estimate the threshold. If the FPR is 0.005, that means 0.5 percent of pixels in the normal images will be misclassified as anomalous with the threshold.

4. Experiments

4.1. Experimental Setup

4.1.1. Dataset

We evaluate our method on the MVTec Anomaly Detection (MVTec AD) dataset [1]. It mimics real-world industrial inspection scenarios and consists of 5354 high-resolution images of 10 objects classes and 5 texture classes from different domains. The MVTec AD dataset provides 3629 images for training and 1725 images for testing. The training set contains only defect-free images. The test set contains both defect-free images and anomalous images, including 73 different types of defects, such as contamination, hole, misplaced and bent. It provides pixel-wise ground truth regions for all anomalies. The resolutions of images are in the range between 700×700 and 1024×1024 pixels. Now it has been a commonly used benchmark for anomaly detection and segmentation task.

4.1.2. Training and Testing Configuration

In the training stage, we only use normal images. We choose ViT-B/16 [9] pretrained on ImageNet as a feature extractor. ViT-B/16 has 12 layers totally and we choose the first four layers to aggregate to dense feature maps $f_{1:4}(x)$ as the represent[ation of the input image. Additionally, we concatenate the output of the linear projection layer with the dense feature maps. The weights of the pretained ViT are frozen during training and testing. We choose Adam [19] as optimizer and the learning rate is set to 1e-4. All the input images are resized to 384×384. The size of the input of the SAAE is $24 \times 24 \times 3840$. The patch size of the SAAE is 1×1. We use Principal Component Analysis (PCA) to compute latent dim V, as described in [3], which retains 80% information. The MLP size is V for texture classes and $2V$ for object classes. The depth of the transformer encoder and decoder is 1 and the heads are 3. We train our model on a single GTX 1080Ti for 700 epochs, and the batch size is 32.

4.1.3. Evaluation

As the measurements reported in the existing literature, we choose two threshold-independent metrics to evaluate the performance of anomaly segmentation. The first one is the area under the receiver operating characteristic curve (ROC-AUC). It calculates the accuracy of pixel-level segmentation. We regard abnormal pixels as positive and normal pixel as negative, so the true positive rate (TPR) is the percentage of the pixels correctly classified as abnormal and the false positive rate (FPR) is the percentage of the pixels incorrectly classified as abnormal. ROC-AUC is simple and widely used, but it has a problem that a large area that is segmented correctly can offset many incorrectly segmented small areas. So we choose the second metric called the area under the per-region-overlap curve (PRO-AUC), which is proposed in [7]. In contrast to per-pixel ROC-AUC, it measures the performance of the anomaly segmentation at region level. It gives equal weights to all the connected components within the ground truth anomalous region, which means

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2021 doi:10.20944/preprints202108.0570.v1

it treats equally to the big or small anomalous regions. We compute the PRO-AUC as described in [7] and record the normalized PRO-AUC up to an average per-pixel FPR of 30%.

Table 2. Anomaly segmentation accuracy on MVTec AD (ROC-AUC %)

Class	AE SSIM	AE L2	VAE	CNN Dict	Patch SVDD	DFR	PaDiM	Ours
Carpet	87	59	59.7	72	92.6	97	**99.1**	97.9
Grid	94	90	61.2	59	96.2	98	97.3	**98.6**
Leather	78	75	67.1	87	97.4	98	99.2	**99.6**
Tile	59	51	51.3	93	91.4	87	94.1	**97.3**
Wood	73	73	66.6	91	90.8	93	94.9	**97.6**
Mean texture classes	78	70	61.2	80	93.7	95	96.9	**98.2**
Bottle	93	86	83.1	78	98.1	97	**98.3**	97.9
Cable	82	86	83.1	79	96.8	92	96.7	**96.8**
Capsule	94	88	81.7	84	95.8	**99**	98.5	98.2
Hazelnut	97	95	87.7	72	97.5	**99**	98.2	**98.5**
Metal Nut	89	86	78.7	82	**98**	93	97.2	97.6
Pill	91	85	81.3	68	95.1	97	95.7	**98.1**
Screw	96	96	75.3	87	95.7	**99**	98.5	**98.9**
Toothbrush	92	93	91.9	77	98.1	**99**	98.8	98.7
Transistor	90	86	75.4	66	97	80	**97.5**	96.0
Zipper	88	77	71.6	76	95.1	96	**98.5**	96.9
Mean object classes	91	88	81.0	77	96.7	95	97.8	**97.8**
Mean	87	82	74.4	78	95.7	95	97.5	**97.9**

Mean refers to the average value of all classes. Mean texture classes and mean object classes refer to the average value of all texture classes and all object classes. The maximum value of each row is bolded.

Table 3. Anomaly segmentation accuracy on MVTec AD (PRO-AUC %)

Class	AE SSIM	AE L2	VAE	CNN Dict	ST p=33	DFR	PaDiM	Ours
Carpet	64.7	45.6	61.9	46.9	89.3	93	**96.2**	93.1
Grid	84.9	58.2	40.8	18.3	94.9	93	94.6	**96.4**
Leather	56.1	81.9	64.9	64.1	95.6	97	97.8	**98.7**
Tile	17.5	89.7	24.2	79.7	**95.0**	79	86.0	92.7
Wood	60.5	72.7	57.8	62.1	92.9	91	91.1	**95.4**
Mean texture classes	56.7	69.6	49.9	54.2	93.5	91	93.1	**95.3**
Bottle	83.4	91.0	70.5	74.2	89.0	93	**94.8**	94.3
Cable	47.8	82.5	77.9	55.8	76.4	81	88.8	**89.0**
Capsule	86.0	86.2	77.9	30.6	**96.3**	97	93.5	92.9
Hazelnut	91.6	91.7	77.0	84.4	96.5	97	92.6	**96.6**
Metal Nut	60.3	83.0	57.6	35.8	**92.8**	90	85.6	91.7
Pill	83.0	89.3	79.3	46.0	95.9	96	92.7	**97.1**
Screw	88.7	75.4	66.4	27.7	93.7	**96**	94.4	94.6
Toothbrush	78.4	82.2	85.4	15.1	**94.4**	93	93.1	93.1
Transistor	72.5	72.8	61.0	62.8	61.1	79	84.5	**88.2**
Zipper	66.5	83.9	60.8	70.3	94.2	90	**95.9**	90.3
Mean object classes	75.8	83.8	71.4	50.3	89.0	91	91.6	**92.8**
Mean	69.5	79.1	64.2	51.6	90.5	91	92.1	**93.6**

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2021 doi:10.20944/preprints202108.0570.v1

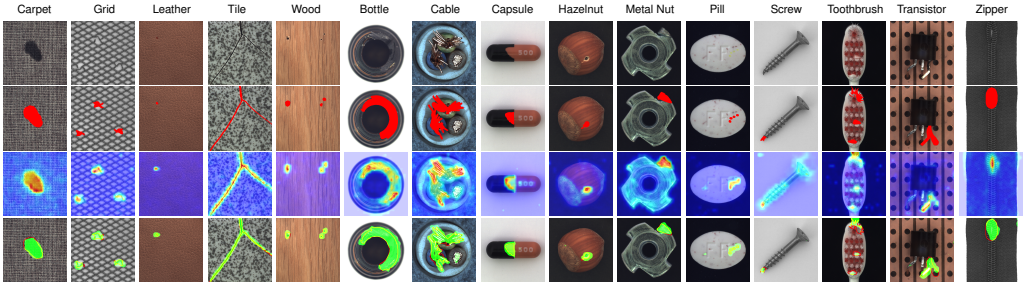

Figure 3. This is the qualitative result of our model. The first row is input image, the second row is ground truth anomaly segmentation and the third is the anomaly score map produced by our model. The last row is the segmentation result when FPR of 0 for texture classes and FPR of 0.001 for object classes on corresponding training set are given.

4.2. Result

We evaluate our method on the MVTec AD dataset for anomaly segmentation. The quantitative results is shown in Table 2 and Table 3. We compare the performance of our approach with the current state-of-the-art methods. We quote the ROC-AUC and PRO-AUC results from [1,3,5–7] directly because of the same evaluation metrics.

Table 2 shows the ROC-AUC results. In general, our method far exceeds other methods in textures and produces similar performance in objects with the PaDiM which is the current strongest model. On average, our method becomes a new state-of-the-art result with 97.9% ROC-AUC. Table 3 shows the PRO-AUC results. Our method performs well both in textures and objects, and improves the state-of-the-art result by 1.5% on average. In general, our method has better performance on the MVtecAD dataset compared with all of the other methods. In addition, it is noticeable that our model has balanced performance on each category in texture and objects which means more general and stable.

Compared to DFR [3] which is a basic feature reconstruction based method, we get similar or better results in all the classes, which suggest our feature extractor and autoencoder are quite useful. We improve a lot in tile and transistor for over 10% ROC-AUC.

We visualize some anomaly segmentation results of our method in Fig 3. The first row is the input image, the second row is ground truth segmentation and the third row is the anomaly score map produced by our model. The last row is the segmentation result when FPR of 0 for texture classes and FPR of 0.001 for object classes on corresponding training set are given. We show the result of all the 15 categories in the MVTec AD dataset. We could see that the segmentation result produced by our method is pretty good and very close to the ground truth result.

Table 4. Ablation Study (ROC-AUC %)

Class	w/o ViT	w/o SAAE	Ours
Mean texture classes	95.7	97.5	98.2
Mean object classes	97.0	96.5	97.8
Mean	96.5	96.9	97.9

4.3. Ablation Study

Our method uses ViT as feature extractor and self-attention autoencoder to reconstruct the feature maps. To show their effectiveness, we design two ablation studies and report the result in Table 4.

In the first ablation study, we use a convolutional autoencoder (CAE) to replace our self-attention autoencoder. The structure of CAE is the same as described in [7], which is equal to our self-attention autoencoder removed the transformer module. The result shows a significant drop both in ROC-AUC and PRO-AUC when using CAE, which verifies that global self-attention is effective for anomaly segmentation.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2021 doi:10.20944/preprints202108.0570.v1

The second ablation study uses a VGG19 [19] to replace the ViT as feature extractor. We concatenate all the first 12 feature maps together to generate a dense feature representation with a regional feature generator proposed in [3]. The input image is resized to 256×256. The input feature representation of SAAE is size of $64 \times 64 \times 3456$. The patch size of the SAAE is 8. Other configurations are the same as described in IV.A. The result shows that ViT as feature extractor is much stronger than VGG19. It suggests that global context information and better performance on the ImageNet classification task yield a good feature representation for feature reconstruction based anomaly detection method.

Table 5. Anomaly segmentation accuracy on MVTec AD with different ViT layers (ROC-AUC %)

Class	$f_{1:4}$	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}
Carpet	97.9	86.2	93.4	96.8	98.2	98.5	98.5	93.5	96.7	98.0	97.1	82.9	94.8
Grid	98.6	97.3	98.7	98.3	97.5	96.8	91.1	89.3	88.8	91.6	91.8	86.6	85.7
Leather	99.6	98.0	99.3	99.5	99.6	99.6	98.8	99.4	99.5	99.4	99.5	98.9	97.5
Tile	97.3	92.2	94.3	96.6	96.9	96.4	89.5	92.8	93.1	94.2	95.3	92.6	86.5
Wood	97.6	91.1	95.9	97.2	97.7	75.5	95.6	95.0	92.6	91.0	94.2	92.8	85.6
Mean texture classes	98.2	93.0	96.3	97.7	98.0	93.4	94.7	94.0	94.1	94.8	95.6	90.8	90.0
Bottle	97.9	89.5	95.1	97.2	97.8	97.4	93.6	80.6	86.4	95.0	94.7	91.1	88.9
Cable	96.8	64.4	93.7	95.5	95.8	96.2	93.8	91.9	95.1	95.5	95.6	91.5	92.0
Capsule	98.2	95.9	97.0	97.2	96.7	95.8	93.3	90.9	93.5	93.8	92.9	90.0	82.9
Hazelnut	98.5	97.3	98.2	98.2	98.1	98.1	94.2	93.4	97.5	97.0	97.2	92.8	95.6
Metal Nut	97.6	92.6	95.1	95.4	93.4	95.1	82.8	71.2	79.3	81.4	87.4	83.5	93.1
Pill	98.1	92.7	92.8	89.9	96.8	79.9	84.5	88.0	92.8	90.8	92.2	88.2	92.6
Screw	98.9	97.6	97.8	97.2	98.2	79.0	93.0	92.0	92.5	92.1	92.3	89.7	84.9
Toothbrush	98.7	98.6	98.5	98.3	96.0	85.7	63.3	62.7	60.5	61.7	66.9	63.9	67.2
Transistor	96.0	88.4	94.2	95.7	95.4	93.8	87.1	88.6	89.4	88.4	83.4	85.2	89.9
Zipper	96.9	94.7	95.6	94.5	94.9	88.5	76.0	66.7	75.0	74.3	76.1	75.9	65.2
Mean object classes	97.8	91.2	95.8	95.9	96.3	93.5	86.2	82.6	86.2	87.0	87.9	85.2	85.2
Mean	97.9	91.8	96.0	96.5	96.9	89.3	89.0	86.4	88.8	89.6	90.4	87.0	86.8

4.4. Analysis

4.4.1. Impact of Different ViT Layers

We choose feature maps from different layers of ViT as the representation of the original image to test the effect. We evaluate the performance of the feature maps of all 12 layers on the MVTec Dataset. Additionally, we concatenate the output of the linear projection layer with each feature map. The latent dimension V is hard to choose and even PCA is not suitable for every feature map. We use PCA to compute latent dim V for the feature map of layer 4 to retain 80% information. For the feature maps of other layers, their latent dim is set to be the same as layer 4, which is simple and good in practice.

The result is shown in Table 5. Layer 4 produces the best result and the shallower or deeper layers are slightly worse than it. We notice that in most cases, the shallow layers produce better results than deep layers. The layers deeper than layer 6 yield poor results on the toothbrush and zipper. The result of $f_4(x)$ is similar to our final result $f_{1:4}(x)$. So you can only use $f_4(x)$ for lower memory cost.

4.4.2. Time Efficiency

We test the running time of our model on each class of the MVTec AD dataset. We use the configuration described in IV.A. The inference time is related to the categories, so we record the average value of all categories. The average running time of our model is 86 frames-per-second (FPS), which proves our method support real-time anomaly detection and thus can be used in practice. We also test the DFR which is a baseline feature reconstruction based method in the same GPU environment. The inference speed of it is 31 FPS in the configuration of $f_{1:12}$ [7]. It shows that our method is pretty time-efficient.

5. Conclusions

In this work, we propose a new anomaly segmentation method based on feature reconstruction. We use ViT as a feature extractor and design an efficient self-attention based autoencoder to reconstruct the feature representation. Our model outperforms the state-of-the-art methods on the MVTec AD dataset. This result suggests that global self-attention has the ability to reconstruct the normal pixels well and reconstruct the abnormal pixels to look like normal ones because self-attention has global receptive field to see other regions which are mostly normal. This work is a first step to discover the ability of self-attention in anomaly detection and segmentation task. Further study is needed to tap the potential of self-attention autoencoder in this aspect.

Funding: This research received no external funding.

Data Availability Statement: The dataset used in this article is MVTec AD. For details, please refer to [1].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bergmann, P.; Fauser, M.; Sattlegger, D.; Steger, C. MVTec AD — A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CA, USA, 16-20 June 2019; pp. 9584–9592.
2. Bergmann, P.; Löwe, S.; Fauser, M.; Sattlegger, D.; Steger, C. Improving Unsupervised Defect Segmentation by Applying Structural Similarity to Autoencoders. In Proceedings of the 14th International Joint Conference on Computer Vision, Prague, Czech Republic, 25-27 February 2019; pp. 372–380.
3. Shi, Y.;Yang, J.; Qi, Z. Unsupervised anomaly segmentation via deep feature reconstruction. *Neurocomputing* **2021**, *424*, 9–22.
4. Yi, J.; Yoon, S. Patch SVDD: Patch-Level SVDD for Anomaly Detection and Segmentation. In Proceedings of the Asian Conference on Computer Vision, Kyoto, Japan, 30 November-4 December 2020; pp. 375–390.
5. Zavrtanik, V.;Kristan, M.; Skočaj, D. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition* **2021**, *112*, 107706.
6. Defard, T.; Setkov, A.; Loesch, A.; Audigier, R. PaDiM: a Patch Distribution Modeling Framework for Anomaly Detection and Localization. *arXiv:2011.08785 [cs]* **2020** [Online]. Available: http://arxiv.org/abs/2011.08785
7. Bergmann, P.; Fauser, M.; Sattlegger, D.; Steger, C. Uninformed Students: Student-Teacher Anomaly Detection With Discriminative Latent Embeddings. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition , Seattle Washington, USA, 14-19 June 2020; pp. 4182–4191.
8. Fei, Y.; Huang, C.; Jinkun, C.; Li, M; Zhang Y.; Lu, C. Attribute Restoration Framework for Anomaly Detection. *IEEE Transactions on Multimedia* **2020**.
9. Dosovitskiy, A. et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv:2010.11929 [cs]* **2020** [Online]. Available: http://arxiv.org/abs/2010.11929
10. Vaswani, A. et al. Attention Is All You Need. *arXiv:1706.03762 [cs]* **2017** [Online]. Available: http://arxiv.org/abs/1706.03762
11. Devlin, J.; Chang, M.W.; Lee K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]* **2019** [Online]. Available: http://arxiv.org/abs/1810.04805
12. Wang, Z.; Bovik, A.C.; Sheikh H.R.; Simoncelli, E.P. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. on Image Process* **2004**, *13*, 600–612.
13. Deng, J.; Dong, W.; Socher, R.; Li, L.; Kai, L.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition , Miami Florida, USA, 20-26 June 2009; pp. 248-255.
14. Radford, A.; Narasimhan, K.; Salimans T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. **2018**.
15. Han, K. et al. A Survey on Visual Transformer. *arXiv:2012.12556 [cs]* **2021** [Online]. Available: http://arxiv.org/abs/2012.12556
16. Chen, H. et al. Pre-Trained Image Processing Transformer. *arXiv:2012.00364 [cs]* **2020** [Online]. Available: http://arxiv.org/abs/2012.00364
19. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556 [cs]* **2015** [Online]. Available: http://arxiv.org/abs/1409.155
18. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 6-11 July 2015; pp. 448-456.
19. Simonyan, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]* **2017** [Online]. Available: http://arxiv.org/abs/1412.6980
20. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, 21-24 June 2010.
21. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the European Conference on Computer Vision, online, 23-28 August 2020; pp. 213–229.

 doi:10.20944/preprints202108.0570.v1

22. Akcay, S.; Atapour-Abarghouei, A.; Breckon, T.P. GANomaly: Semi-Supervised Anomaly Detection via Adversarial Training. *arXiv:1805.06725 [cs]* **2018** [Online]. Available: http://arxiv.org/abs/1805.06725
23. LeCun, Y.; Boser, B.; Denker J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation applied to handwritten zip code recognition. *Neural computation* **1989**, *1*, 541–551.
24. Jiang, Y.; Chang, S.; Wang Z. TransGAN: Two Transformers Can Make One Strong GAN. *arXiv:2102.07074 [cs]* **2021** [Online]. Available: http://arxiv.org/abs/2102.07074