Student-Teacher Feature Pyramid Matching for Anomaly Detection

Guodong Wang*[1,2]
wanggd@buaa.edu.cn
Shumin Han*[3]
hanshumin@baidu.com
Errui Ding[3]
dingerrui@baidu.com
Di Huang†[1,2]
dhuang@buaa.edu.cn

[1] State Key Laboratory of Software Development Environment
Beihang University
Beijing, China

[2] School of Computer Science and Engineering
Beihang University
Beijing, China

[3] Department of Computer Vision Technology
Baidu, Inc.
Beijing, China

Abstract

Anomaly detection is a challenging task and usually formulated as an one-class learning problem for the unexpectedness of anomalies. This paper proposes a simple yet powerful approach to this issue, which is implemented in the student-teacher framework for its advantages but substantially extends it in terms of both accuracy and efficiency. Given a strong model pre-trained on image classification as the teacher, we distill the knowledge into a single student network with the identical architecture to learn the distribution of anomaly-free images and this one-step transfer preserves the crucial clues as much as possible. Moreover, we integrate the multi-scale feature matching strategy into the framework, and this hierarchical feature matching enables the student network to receive a mixture of multi-level knowledge from the feature pyramid under better supervision, thus allowing to detect anomalies of various sizes. The difference between feature pyramids generated by the two networks serves as a scoring function indicating the probability of anomaly occurring. Due to such operations, our approach achieves accurate and fast pixel-level anomaly detection. Very competitive results are delivered on the MVTec anomaly detection dataset, superior to the state of the art ones.

1 Introduction

Anomaly detection is generally referred to as identifying samples that are atypical with respect to regular patterns in the data set and has shown great potential in various real-world applications such as video surveillance [1, 31], product quality control [7, 8, 27] and medical

* Equal contribution.
† Corresponding author.

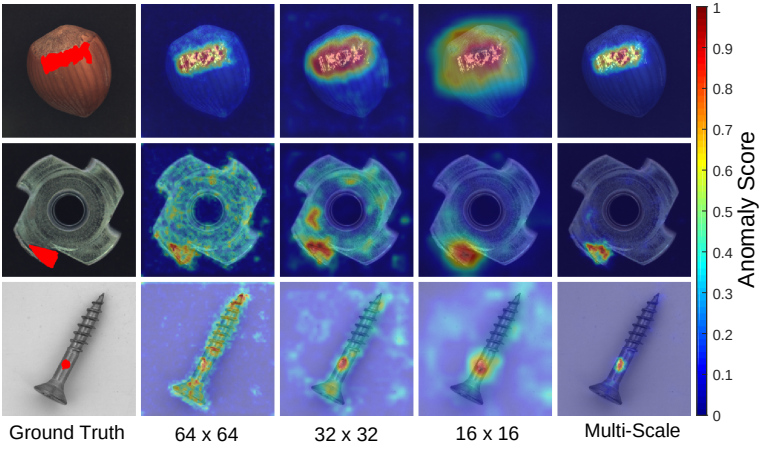


Figure 1: Visual results of our method on three defective images from the MVTec AD dataset. ResNet-18 is used as backbone and the three bottom blocks (*i.e.*, conv2_x, conv3_x, conv4_x) are selected as feature extractors. Columns from left to right correspond to input images with defects (ground truth regions in red), anomaly maps of the three blocks, and the resulting anomaly maps respectively.

diagnosis [34, 35, 40]. Its key challenge lies in the unexpectedness of anomalies which is very difficult to deal with in a supervised way, as labeling all types of anomalous instances seems unrealistic.

Previous studies address this challenge in the form of one-class learning paradigm [25]. They approximate the decision boundary for a binary classification problem by searching a feature space where the distribution of normal data is accurately modeled. Deep learning, in particular convolutional neural networks (CNNs) [20] and residual networks (ResNets) [16], provides a powerful alternative to automatically build comprehensive representations at multiple levels. Such deep features prove very effective in capturing the intrinsic characteristics of the normal data manifold [3, 10, 24, 32, 47]. Despite the promising results in their respective fields, all these methods simply predict anomalies at the image-level without spatial localization.

The pixel-level methods advance anomaly detection by means of pixel-wise comparison of image patches and their reconstructions [6, 34, 35] or per-pixel estimation of probability density on entire images [1, 37], among which Auto-encoders, Generative Adversarial Networks (GANs), and their variants are dominating models. However, their performance is prone to serious degradation when images are poorly reconstructed [30] or likelihoods are inaccurately calibrated [26].

Some recent attempts transfer the knowledge from other well-studied computer vision tasks. They directly apply the networks pre-trained on image classification and show that they are sufficiently generic to image-level detection [4, 9, 14]. Cohen and Hoshen [11] investigate this idea in pixel-level detection and delivers performance gain; unfortunately, it has the time bottleneck due to per-pixel comparison. Bergmann *et al.* [3] utilize the pre-trained model in a more efficient way by implicitly learning the distribution of normal features with a student-teacher framework and reach decent results. The difference between the outputs of the students and teacher along with the uncertainty among students' predictions serves as

the anomaly scoring function. Nevertheless, two major drawbacks still remain: *i.e.*, the incompleteness of transferred knowledge and complexity of handling scaling. For the former, since knowledge is distilled from a ResNet-18 [16] into a lightweight teacher network, the big gap between their model capacities [42] tends to incur loss of important information. For the latter, multiple student-teacher ensemble pairs are required to be separately trained, each for a specific respective field, to achieve scale invariance, which leads to the inconvenience in computation. Both the facts leave much room for improvement.

In this paper, we propose a simple yet powerful approach to anomaly detection, which follows the student-teacher framework for the advantages but substantially extends it in terms of both accuracy and efficiency. Specifically, given a powerful network pre-trained on image classification as the teacher, we distill the knowledge into a single student network with the identical architecture. In this case, the student network learns the distribution of anomaly-free images by matching their features with the counterparts of the pre-trained network, and this one-step transfer preserves the crucial information as much as possible. Furthermore, to enhance the scale robustness, we embed multi-scale feature matching into the network, and this hierarchical feature matching strategy enables the student network to receive a mixture of multi-level knowledge from the feature pyramid under a stronger supervision and thus allows to detect anomalies of various sizes (see Figure 1 for visualization). The feature pyramids from the teacher and student networks are compared for prediction, where a larger difference indicates a higher probability of anomaly occurrence.

Compared to the previous work, especially the preliminary student-teacher model, the benefits of our approach are two-fold. First, useful knowledge is well transferred from the pre-trained network to the student network within one-step distillation, as they share the same structure. Second, thanks to the hierarchical structure of the network, multi-scale anomaly detection is conveniently reached by the proposed feature pyramid matching scheme. Due to such strengths, our approach conducts accurate and fast pixel-level anomaly detection. It reports very competitive results on the MVTec anomaly detection dataset, and more results on ShanghaiTech Campus (STC) [23] and CIFAR-10 [18] are presented in the supplementary material.

2 Related Work

2.1 Image-level Anomaly Detection

Image-level techniques manifest anomalies in images of unseen categories. They can be coarsely divided into: reconstruction-based, distribution-based and classification-based.

The first group of approaches reconstruct the training images to capture the normal data manifold. An anomalous image is very likely to possess a high reconstruction error during inference, as it is drawn from a different distribution. The main weakness of these approaches comes from the excellent generalization ability of the deep models, including variational autoencoder [3], robust autoencoder [47], conditional GAN [2], and bi-directional GAN [46], which probably allows anomalous images to be faithfully reconstructed.

Distribution-based approaches model the probabilistic distribution of the normal images. The images that have low probability density values are designated as anomalous. Recent algorithms such as anomaly detection GAN (ADGAN) [12] and deep autoencoding Gaussian mixture model (DAGMM) [48] learn a deep projection that maps high-dimensional images into a low-dimensional latent space. Nevertheless, these methods have high sample com-

plexity and demand large training data.

Classification-based approaches have dominated anomaly detection in the last decade. One useful paradigm is to feed the deep features extracted by deep generative models [9] or transferred from pre-trained networks [4, 14] into a separate shallow classification model like one-class support vector machine (OC-SVM) [36]. Another line of research depends on self-supervised learning. Geom [15] creates a dataset by applying dozens of geometric transformations to the normal images and trains a multi-class neural network over the self-labeled dataset to discriminate such transformations. At test time, anomalies are expected to be assigned with less confidence in discriminating the transformations.

2.2 Pixel-level Anomaly Detection

Pixel-level techniques are particularly designed for anomaly localization. They aim to precisely segment anomalous regions in images, which is more complicated than binary classification.

The expressive power of deep neural networks inspires a series of studies that explore how to transfer the benefits of the networks pre-trained on image classification tasks to anomaly detection. Napoletano *et al.* [27] exploit a pre-trained ResNet-18 to embed cropped training image patches into a feature space, reduce the dimension of feature vectors by PCA, and model their distribution using K-means clustering. This method requires a large number of overlapping patches to obtain a spatial anomaly map at inference time, which results in coarse-grained maps and may become a performance bottleneck.

To avoid cropping image patches and accelerate feature extraction, Sabokrou *et al.* [33] build descriptors from early feature maps of a pre-trained fully convolutional network (FCN) and adopt a unimodal Gaussian distribution to fit feature vectors of the anomaly-free images. However, the unimodel Gaussian distribution fails to characterize the training feature distribution as the problem complexity increases. More recently, a convolutional adversarial variational autoencoder with guided attention (CAVGA) [41] incorporates Grad-CAM [38] into a variational autoencoder with an attention expansion loss to encourage the deep model itself to focus on all normal regions in the image. Simliar to typical autoencoders (AE) [2, 30] and variational autoencoders (VAE) [22], CAVGA also suffers from the strong generalization ability which allows good reconstruction for anomalous images.

3 Method

3.1 Framework

We make use of the student-teacher learning framework to implicitly model the feature distribution of the normal training images. The teacher is a powerful network pre-trained on the image classification task (*e.g.*, a ResNet-18 pre-trained on ImageNet). To reduce information loss, the student shares the same architecture with the teacher. This is in essence one case of feature-based knowledge distillation [42].

Here, we need to consider a key factor, *i.e.*, position of distillation. Deep neural networks generate a pyramid of features for each input image. Bottom layers result in higher-resolution features encoding low-level information such as textures, edges and colors. By contrast, top layers yield low-resolution features that contain context information. The features created by bottom layers are often generic enough and they can be shared by various vision tasks [29,

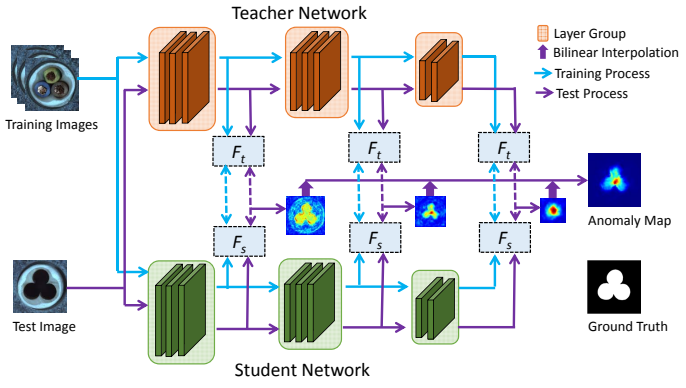


Figure 2: Schematic overview of our method. The feature pyramid of a student network is trained to match with the counterpart of a pre-trained teacher network. A test image (or pixel) has a high anomaly score if its features from the two models differ significantly. The feature pyramid matching enables our method to detect anomalies of various sizes with a single forward pass.

45]. This motivates us to integrate low-level and high-level features in a complementary way. As different layers in deep neural networks correspond to distinct receptive fields, we select the features extracted by a few successive bottom layer groups (*e.g.*, blocks in ResNet-18) of the teacher to guide the student's learning. This hierarchical feature matching allows our method to detect anomalies of various sizes.

Figure 2 gives a sketch of our method with the images from the MVTec AD dataset [8] as examples. The training and test processes are formally provided as follows.

3.2 Training Process

The training phase aims to obtain a good student which can perfectly imitate the outputs of a fixed teacher on normal images. Formally, given a training dataset of anomaly-free images $\mathcal{D} = \{I_1, I_2, \ldots, I_n\}$, our goal is to capture the normal data manifold by matching the features extracted by the L bottom layer groups of the teacher with the counterparts of the student. For an input image $I_k \in^{w \times h \times c}$, where h is the height, w is the width and c is the number of the color channels, the lth bottom layer group of the teacher and student outputs a feature map $F_t^l(I_k) \in^{w_l \times h_l \times d_l}$ and $F_s^l(I_k) \in^{w_l \times h_l \times d_l}$, where w_l, h_l and d_l denote the width, height and channel number of the feature map, respectively. Since there is no prior knowledge regarding the appearances and locations of objects, we simply assume that all image regions are anomaly-free in the training set. Note that $F_t^l(I_k)_{ij} \in^{d_l}$ and $F_s^l(I_k)_{ij} \in^{d_l}$ are feature vectors at position (i, j) in the feature maps from the teacher and student, respectively. We define the loss at position (i, j) as ℓ_2-distance between the ℓ_2-normalized feature vectors, namely,

$$\ell^l(I_k)_{ij} = \frac{1}{2} \left\| \hat{F}_t^l(I_k)_{ij} - \hat{F}_s^l(I_k)_{ij} \right\|_{\ell_2}^2, \tag{1}$$

$$\hat{F}_t^l(I_k)_{ij} = \frac{F_t^l(I_k)_{ij}}{\|F_t^l(I_k)_{ij}\|_{\ell_2}}, \; \hat{F}_s^l(I_k)_{ij} = \frac{F_s^l(I_k)_{ij}}{\|F_s^l(I_k)_{ij}\|_{\ell_2}}.$$

It is worth noting that the ℓ_2 distance used in (Eq. 1) is proportional to the cosine distance as $F_t^l(I_k)$ and $F_s^l(I_k)$ are ℓ_2-normalized vectors. Thus the loss $\ell^l(I_k)_{ij} \in (0,1)$. The loss for the entire image I_k is given as an average of the loss at each position,

$$\ell^l(I_k) = \frac{1}{w_l h_l} \sum_{i=1}^{w_l} \sum_{j=1}^{h_l} \ell^l(I_k)_{ij}, \tag{2}$$

and the total loss is the weighted average of the loss at different pyramid scales,

$$\ell(I_k) = \sum_{l=1}^{L} \alpha_l \ell^l(I_k), \quad \text{s.t. } \alpha_l \geq 0, \tag{3}$$

where α_l depicts the impact of the lth feature scale on anomaly detection. We simply set $\alpha_l = 1, l = 1, \ldots, L$ in all our experiments. Given a minibatch $\mathcal{B}$ sampled from the training dataset $\mathcal{D}$, we update the student by minimizing the loss $\ell_{\mathcal{B}} = \frac{1}{|\mathcal{B}|} \sum_{k \in \mathcal{B}} \ell(I_k)$. Note that we only update the student while keeping the teacher fixed throughout the training phase.

3.3 Test Process

In the test phase, we aim to obtain an anomaly map Ω of size $w \times h$ regarding a test image $J \in^{w \times h \times c}$. The score $\Omega_{ij} \in [0,1]$ indicates how much the pixel at position (i, j) deviates from the training data manifold. We forward the test image J into the teacher and the student. Let $F_t^l(J)$ and $F_s^l(J)$ denote the feature maps generated by the lth bottom layer group of the teacher and the student, respectively. We can compute an anomaly map $\Omega^l(J)$ of size $w_l \times h_l$, whose element $\Omega_{ij}^l(J)$ is the loss (Eq. 1) at position (i, j). The anomaly map $\Omega^l(J)$ is upsampled to size $w \times h$ by bilinear interpolation. The resulting anomaly map is defined as the element-wise product of L equal-sized upsampled anomaly maps,

$$\Omega(J) = \prod_{l=1}^{L} \Omega^l(J). \tag{4}$$

A test image is designated as anomaly if any pixel in the image is anomalous. As a result, we simply choose the maximum value in the anomaly map, *i.e.*, $\max(\Omega(J))$ as the anomaly score for the test image J.

4 Experiments

4.1 Dataset

We conduct experiments on the MVTec Anomaly Detection (MVTec AD) [7] dataset, with both the image-level and pixel-level anomaly detection tasks considered. The dataset is specifically created to benchmark algorithms for anomaly localization. It collects more than 5,000 high-resolution images of industrial products covering 15 different categories. For each category, the training set only includes defect-free images and the test set comprises both defect-free images and defective images of different types. The performance is measured by two popular metrics: AUC-ROC and Per-Region-Overlap (PRO) [8]. Supplementary material provides more results on ShanghaiTech Campus (STC) [23] and CIFAR-10 [18].

4.2 Implementation Details

For all the experiments, we choose the first three blocks (*i.e.*, conv2_x, conv3_x, conv4_x) of ResNet-18 as the pyramid feature extractors for both the teacher and student networks. The parameters of the teacher network are copied from the ResNet-18 pre-trained on ImageNet, while those of the student network are initialized randomly. We train the network using stochastic gradient descent (SGD) with a learning rate of 0.4 for 100 epochs. The batch size is 32. All the images in the training and test sets are resized to 256×256. For each category, we use 80% of training images to build the student, keeping the remaining 20% for validation. We select the checkpoint with the lowest validation error (Eq. 1) to perform anomaly detection.

4.3 Results

We begin with the task of finding anomalous images. As defective regions usually occupy a small proportion of the whole image, the test anomalies differ in a subtle way from the training images. This makes the MVTec AD dataset more challenging than those previously used in the literature (*e.g.*, MNIST and CIFAR-10) where the images from the other categories are regarded as anomalous to the selected one. Table 2 compares our method to state-of-the-art approaches: Geom [15], GANomaly [2], ł_2-AE [5], ITAE [17], Cut-Paste [21] Patch-SVDD [43], PaDiM [13] and SPADE [11]. We clearly see that our approach outperforms all the other methods. In particular, the performance is improved up to 11.7% compared with SPADE [11], which also leverages multi-scale features from a pre-trained model. It validates the superiority of the student-teacher learning framework.

We then consider the task of pixel-level anomaly detection and compare our method with the counterparts including Patch-SVDD [43], PaMiD [13], *etc*. Table 1 reports the performance in terms of the AUC-ROC and PRO metrics. We notice two trends to achieve performance gains: (1) by pre-trained models, with a Wide-ResNet50×2 network [44], SPADE reports very competitive scores; (2) by self-training techniques, Cut-Paste [21] and Patch-SVDD [43] show this potential through designing proper pretext tasks for feature learning. As our approach assumes that anomaly detection is fulfilled via the heterogeneity of the student and teacher networks, *i.e.* different network parameters learned from individual data, we employ a pre-trained model built on generic images rather than self-supervised learning on the small scale anomaly detection dataset. As Table 1 displays, our approach delivers better performance than the others. It should be noted although STAD [8] adopts the student-teacher learning framework, its performance is always inferior to that of our method. This gap can be attributed to the information loss in its two-step and single-scale knowledge transfer process. This validates our improvement in feature learning. When equipped with the same backbone as SPADE [11], our method further boosts the results, *i.e.* 0.973 and 0.923 in AUC-ROC and PRO, respectively.

5 Ablation Studies and Discussions

We first perform feature visualization to investigate what the student learns from its teacher and also conduct ablation studies on the MVTec AD dataset to answer the following three questions. Is feature pyramid matching superior to single feature matching? Is the teacher pre-trained on other datasets still useful? Is our method applicable to small training dataset?

	Category	SSIM-AE	AnoGAN	CNN-Dict*	STAD*	Cut-Paste	Patch-SVDD	PaDiM-R18*	SPADE*	Ours*
Textures	Carpet	0.65	0.20	0.47	0.695	-	-	**0.960**	0.947	0.958
		0.87	0.54	0.72	-	0.983	0.926	**0.989**	0.975	0.988
	Grid	0.85	0.23	0.18	0.819	-	-	0.909	0.867	**0.966**
		0.94	0.58	0.59	-	0.975	0.962	0.949	0.937	**0.990**
	Leather	0.56	0.38	0.64	0.819	-	-	0.979	0.972	**0.980**
		0.78	0.64	0.87	-	**0.995**	0.974	0.991	0.976	0.993
	Tile	0.18	0.18	0.80	0.912	-	-	0.816	0.759	**0.921**
		0.59	0.50	0.93	-	0.905	0.914	0.912	0.874	**0.974**
	Wood	0.61	0.39	0.62	0.725	-	-	0.903	0.874	**0.936**
		0.73	0.62	0.91	-	0.955	0.908	0.936	0.885	**0.972**
Objects	Bottle	0.83	0.62	0.74	0.918	-	-	0.939	**0.955**	0.951
		0.93	0.86	0.78	-	0.976	0.981	0.981	0.984	**0.988**
	Cable	0.48	0.38	0.56	0.865	-	-	0.862	**0.909**	0.877
		0.82	0.78	0.79	-	0.900	0.968	0.958	**0.972**	0.955
	Capsule	0.86	0.31	0.31	0.916	-	-	0.919	**0.937**	0.922
		0.94	0.84	0.84	-	0.974	0.958	0.983	**0.990**	0.983
	Hazelnut	0.92	0.70	0.84	0.937	-	-	0.914	**0.954**	0.943
		0.97	0.87	0.72	-	0.973	0.975	0.977	**0.991**	0.985
	Metal nut	0.60	0.32	0.36	0.895	-	-	0.819	0.944	**0.945**
		0.89	0.76	0.82	-	0.931	0.980	0.967	**0.981**	0.976
	Pill	0.83	0.78	0.46	0.935	-	-	0.906	0.946	**0.965**
		0.91	0.87	0.68	-	0.957	0.951	0.947	0.965	**0.978**
	Screw	0.89	0.47	0.28	0.928	-	-	0.913	**0.960**	0.930
		0.96	0.80	0.87	-	0.967	0.957	0.974	**0.989**	0.983
	Toothbrush	0.78	0.75	0.15	0.863	-	-	0.923	**0.935**	0.922
		0.92	0.93	0.90	-	0.981	0.981	0.987	0.979	**0.989**
	Transistor	0.73	0.55	0.63	0.701	-	-	0.802	**0.874**	0.695
		0.90	0.86	0.66	-	0.930	0.970	**0.972**	0.941	0.825
	Zipper	0.67	0.47	0.70	0.933	-	-	0.947	0.926	**0.952**
		0.88	0.78	0.76	-	**0.993**	0.951	0.982	0.965	0.985
	Mean	0.69	0.44	0.52	0.857	-	-	0.901	0.917	**0.921**
		0.87	0.74	0.78	-	0.960	0.957	0.967	0.965	**0.970**

* denotes extra dataset pre-trained model used.

Table 1: Pixel-level anomaly detection. For each dataset category, PRO (top row) and AUC-ROC (bottom row) scores are given.

Geom	GANomaly	ł_2-AE	ITAE	Cut-Paste	Patch-SVDD	PaDiM-WR50*	SPADE*	Ours
0.672	0.762	0.754	0.839	0.952	0.921	0.953	0.855	**0.955**

* denotes extra dataset pre-trained model used.

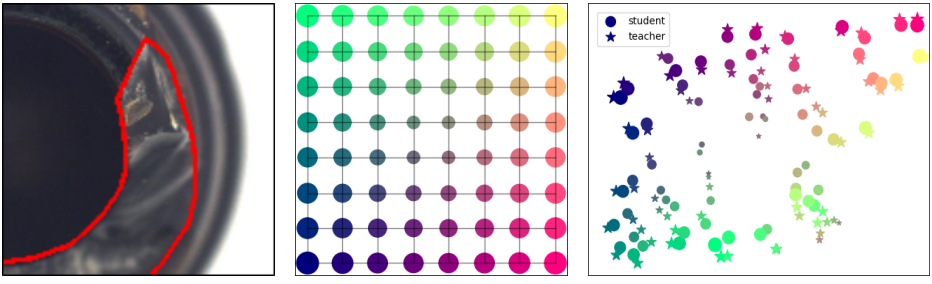
Table 2: Image-level anomaly detection. The performance is measured by average AUC-ROC across 15 categories.

5.1 Feature Visualization

Figure 3 shows t-SNE visualization [39] of learned features from the student and teacher. Obviously, the features from the student and teacher on normal regions distribute closer (even overlapped) than the ones on anomalous regions. It suggests that the student learns to match the teacher's output on normal images. It also shows that the student well captures the distribution of normal patterns under the supervision of a good teacher.

5.2 Feature Matching

We first minutely investigate the effectiveness of feature extraction by each individual block of ResNet-18. Considering that the first block is a simple convolutional layer, we exclude it from comparison. We train the student by matching features extracted by its second, third, fourth and fifth blocks with the counterparts of the teacher respectively. As shown in Table 3, feature matching conducted at the end of the third and fourth blocks can achieve better performance. This is in good agreement with the previous discovery that the middle-level features play a more important role in knowledge transfer [29].



(a) Bottle (broken_large) (b) Points at various positions (c) t-SNE

Figure 3: t-SNE visualization [39] of learned features from the student and teacher. (a) an example test image with defects contoured by a red line. (b) point map in which different positions are encoded by different sizes and colors. (c) t-SNE visualizations for features from the student (circle) and the teacher (star) with (a) as input. Zoomed in for better display.

Metric \ # Block	2	3	4	5	[2, 3]	[2, 3, 4]	[2, 3, 4, 5]
AR_I	0.808	0.917	0.934	0.819	0.849	**0.955**	0.949
AR_P	0.915	0.953	0.957	0.860	0.950	**0.970**	0.969
PRO	0.815	0.897	0.835	0.504	0.886	**0.921**	0.886

Table 3: Ablation studies for feature matching. The performance is measured by the average image-level AUC-ROC (AR_I), average pixel-level AUC-ROC (AR_P) and average PRO across 15 categories.

Metric \ Dataset	ImageNet	MNIST	CIFAR-10	CIFAR-100	SVHN
AR_I	**0.955**	0.619	0.826	0.835	0.796
AR_P	**0.970**	0.759	0.931	0.937	0.902
PRO	**0.921**	0.528	0.863	0.842	0.742

Table 4: Ablation studies for pre-trained datasets. The performance is measured by the average image-level AUC-ROC (AR_I), average pixel-level AUC-ROC (AR_P) and average PRO across 15 categories.

We then test three different combinations of the consecutive blocks of ResNet-18. Likewise, we match the features extracted from the corresponding compound blocks of the teacher and the student. Table 3 shows that the mixture of the second, third and fourth blocks outperforms other combinations as well as the single components. It implies that feature pyramid matching is a better way for feature learning. This finding is also validated in Figure 1. Anomaly maps generated by low-level features are more suitable for precise anomaly localization, but they are likely to include background noise. By contrast, anomaly maps generated by high-level features are able to segment big anomalous regions. The aggregation of anomaly maps at different scales contributes to accurate detection of anomalies of various sizes.

5.3 Pre-trained Datasets

To answer the second question, we pre-train the teacher on a couple of image classification benchmarks, including MNIST [19], CIFAR-10 [18], CIFAR-100 [18], and SVHN [28].

	5%		10%	
Metric	Ours	SPADE	Ours	SPADE
AR_I	**0.871**	0.782	**0.907**	0.797
AR_P	**0.961**	0.932	**0.967**	0.955
PRO	**0.892**	0.842	**0.913**	0.890

Table 5: Performance in terms of the number of training samples. The performance is measured by the average image-level AUC-ROC (AR_I), average pixel-level AUC-ROC (AR_P) and average PRO across 15 categories.

These pre-trained teachers are individually exploited to guide the student training. The MNIST and SVHN datasets simply contain digital numbers from 0 to 9. We see from Table 4 that the teacher networks pre-trained on these two datasets yield worse results. It indicates that the features learned from these two pre-trained models generalize poorly on the MVTec AD dataset. By contrast, the features extracted from the teacher networks pre-trained on CIFAR-10 and CIFAR-100 exhibit better generalization, as they contain more natural images. Note that the performance of these two pre-trained teachers is still inferior to that of the teacher pre-trained on ImageNet. This is because that the ImageNet dataset consists of a huge number of high-resolution natural images, which is crucial to learning more discriminating features.

5.4 Number of Training Samples

We investigate the effect of the training set size in this experiment. Only 5% and 10% anomaly-free images are used to train our model. It can be seen in Table 5 that our model still reaches a satisfactory level even if only a few training images are available. By contrast, SPADE suffers a serious performance degradation. This is caused by the missing of the tailored feature learning. Our model profits from this strategy and can capture the feature distribution of anomaly-free images in the few-shot scenario. Furthermore, our method uses only 10% training samples to outperform the preliminary student-teacher framework [8]. It validates the effectiveness of our feature pyramid matching technique.

6 Conclusion

We present a new feature pyramid matching technique and incorporate it into the student-teacher anomaly detection framework. Given a powerful network pre-trained on image classification as the teacher, we use its different levels of features to guide a student network with the same structure to learn the distribution of anomaly-free images. On account of the hierarchical feature matching, our method is capable of detecting anomalies of various sizes with only a single forward pass. Experimental results on the MVTec AD dataset show that our method achieves superior performance to the state-of-the-art.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (62022011), the Research Program of State Key Laboratory of Software Development Environment (SKLSDE-2021ZX-04), and the Fundamental Research Funds for the Central Universities.

References

[1] Davide Abati, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Latent space autoregression for novelty detection. In *CVPR*, 2019.

[2] Samet Akcay, Amir Atapour-Abarghouei, and Toby P. Breckon. GANomaly: Semi-supervised anomaly detection via adversarial training. In *ACCV*, 2018.

[3] Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probabiliy. Technical report, SNU Data Mining Center, 2015.

[4] Jerone T. A. Andrews, Thomas Tanay, Edward J. Morton, and Lewis D. Griffin. Transfer representation-learning for anomaly detection. In *ICML Workshops*, 2016.

[5] Caglar Aytekin, Xingyang Ni, Francesco Cricri, and Emre Aksu. Clustering and unsupervised anomaly detection with l_2 normalized deep auto-encoder representations. In *IJCNN*, 2018.

[6] Christoph Baur, Benedikt Wiestler, Shadi Albarqouni, and Nassir Navab. Deep autoencoding models for unsupervised anomaly segmentation in brain mr images. In *MICCAI Workshops*, 2018.

[7] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec AD - A comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*, 2019.

[8] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *CVPR*, 2020.

[9] Philippe Burlina, Neil Joshi, and I-Jeng Wang. Where's wally now? deep generative and discriminative embeddings for novelty detection. In *CVPR*, 2019.

[10] Raghavendra Chalapathy, Aditya Krishna Menon, and Sanjay Chawla. Anomaly detection using one-class neural networks. *arXiv:1802.06360*, 2018.

[11] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. *arXiv:2005.02357*, 2020.

[12] Lucas Deecke, Robert Vandermeulen, Lukas RuffStephan Mandt, and Marius Kloft. Image anomaly detection with generative adversarial networks. In *ECML-PKDD*, pages 3–17, 2018.

[13] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: A patch distribution modeling framework for anomaly detection and localization. In *ICPR*, 2021.

[14] Sarah M. Erfani, Sutharshan Rajasegarar, Shanika Karunasekera, and Christopher Leckie. High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning. *Pattern Recognit.*, 58:121–134, 2016.

[15] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In *NeurIPS*, 2018.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[17] Chaoqin Huang, Fei Ye, Jinkun Cao, Maosen Li, Ya Zhang, and Cewu Lu. Attribute restoration framework for anomaly detection. *arXiv:1911.10676*, 2020.

[18] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

[19] Yann LeCun and Corinna Cortes. Mnist handwritten digit database. 2010.

[20] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

[21] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *CVPR*, 2021.

[22] Wenqian Liu, Runze Li, Meng Zheng, Srikrishna Karanam, Ziyan Wu, Bir Bhanu, Richard J Radke, and Octavia Camps. Towards visually explaining variational autoencoders. In *CVPR*, 2020.

[23] Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked rnn framework. In *ICCV*, 2017.

[24] Marc Masana, Idoia Ruiz, Joan Serrat, Van De Weijer Joost, and Antonio M Lopez. Metric learning for novelty and anomaly detection. In *BMVC*, 2018.

[25] M. M. Moya, M. W. Koch, and L. D. Hostetler. One-class classifier networks for target recognition applications. In *WCCI*, 1993.

[26] Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *ICLR*, 2019.

[27] Paolo Napoletano, Flavio Piccoli, and Raimondo Schettini. Anomaly detection in nanofibrous materials by CNN-based self-similarity. *Sensors*, 18(2):209, 2018.

[28] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NeurIPS Workshops*, 2011.

[29] Maxime Oquab, Léon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, 2014.

[30] Michael Fauser David Sattlegger Paul Bergmann, Sindy Löwe and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In *VISIGRAPP*, 2019.

[31] Alina Roitberg, Ziad Al-Halah, and Rainer Stiefelhagen. Informed democracy: Voting-based novelty detection for action recognition. In *BMVC*, 2018.

[32] Lukas Ruff, Robert A. Vandermeulen, Nico Görnitz, Lucas Deecke, Shoaib A. Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *ICML*, 2018.

[33] Mohammad Sabokrou, Mohsen Fayyaz, Mahmood Fathy, Zahra Moayed, and Reinhard Klette. Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes. *CVIU*, 172, 2018.

[34] Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *IPMI*, 2017.

[35] Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks. *MED IMAGE ANAL*, 54:30–44, 2019.

[36] Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a high-dimensional distribution. *NEURAL COMPUT*, 13(7), 2001.

[37] Philipp Seeböck, Sebastian Waldstein, Sophie Klimscha, Bianca S. Gerendas René Donner, Thomas Schlegl, Ursula Schmidt-Erfurth, and Georg Langs. Identifying and categorizing anomalies in retinal imaging data. *arXiv:1612.00686*, 2016.

[38] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, and Devi Parikh. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017.

[39] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9 (11), 2008.

[40] Aleksei Vasilev, Vladimir Golkov, Ilona Lipp, Eleonora Sgarlata, Valentina Tomassini, Derek K. Jones, and Daniel Cremers. q-Space novelty detection with variational autoencoders. *arXiv:1806.02997*, 2018.

[41] Shashanka Venkataramanan, Kuan-Chuan Peng, Rajat Vikram Singh, and Abhijit Mahalanobis. Attention guided anomaly localization in images. In *ECCV*, 2020.

[42] Lin Wang and Kuk-Jin Yoon. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *arXiv preprint arXiv:2004.05937*, 2020.

[43] Jihun Yi and Sungroh Yoon. Patch svdd: Patch-level svdd for anomaly detection and segmentation. In *ACCV*, 2020.

[44] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[45] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014.

[46] Houssam Zenati, Manon Romain, Chuan-Sheng Foo, Bruno Lecouat, and Vijay Chandrasekhar. Adversarially learned anomaly detection. In *ICDM*, 2018.

[47] Chong Zhou and Randy C. Paffenroth. Anomaly detection with robust deep autoencoders. In *KDD*, 2017.

[48] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *ICLR*, 2018.