Uninformed Students: Student–Teacher Anomaly Detection with Discriminative Latent Embeddings

Paul Bergmann   Michael Fauser   David Sattlegger   Carsten Steger

MVTec Software GmbH

www.mvtec.com

{paul.bergmann, fauser, sattlegger, steger}@mvtec.com

Abstract—We introduce a powerful student–teacher framework for the challenging problem of unsupervised anomaly detection and pixel-precise anomaly segmentation in high-resolution images. Student networks are trained to regress the output of a descriptive teacher network that was pretrained on a large dataset of patches from natural images. This circumvents the need for prior data annotation. Anomalies are detected when the outputs of the student networks differ from that of the teacher network. This happens when they fail to generalize outside the manifold of anomaly-free training data. The intrinsic uncertainty in the student networks is used as an additional scoring function that indicates anomalies. We compare our method to a large number of existing deep learning based methods for unsupervised anomaly detection. Our experiments demonstrate improvements over state-of-the-art methods on a number of real-world datasets, including the recently introduced MVTec Anomaly Detection dataset that was specifically designed to benchmark anomaly segmentation algorithms.

1. Introduction

Unsupervised pixel-precise segmentation of regions that appear anomalous or novel to a machine learning model is an important and challenging task in many domains of computer vision. In automated industrial inspection scenarios, it is often desirable to train models solely on a single class of anomaly-free images to segment defective regions during inference. In an active learning setting, regions that are detected as previously unknown by the current model can be included in the training set to improve the model's performance.

Recently, efforts have been made to improve anomaly detection for one-class or multi-class classification [2, 3, 10, 11, 21, 28, 29]. However, these algorithms assume that anomalies manifest themselves in the form of images of an entirely different class and a simple binary image-level decision whether an image is anomalous or not must be made. Little work has been directed towards the development of methods that can segment anomalous regions that only differ in a very subtle way from the training data. Bergmann et al. [7] provide benchmarks for several state-of-the-art algorithms and identify a large room for improvement.

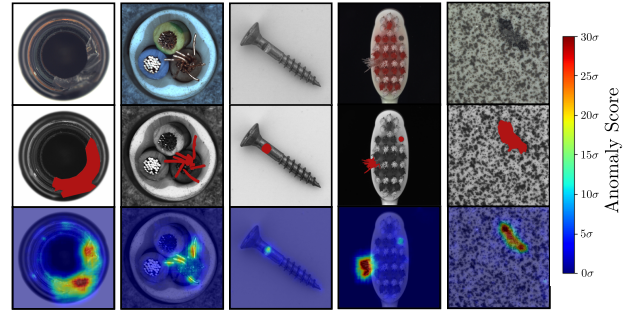Existing work predominantly focuses on generative algorithms such as Generative Adversarial Networks

Figure 1: Qualitative results of our anomaly detection method on the MVTec Anomaly Detection dataset. **Top row:** Input images containing defects. **Center row:** Ground truth regions of defects in red. **Bottom row:** Anomaly scores for each image pixel predicted by our algorithm.

(GANs) [31, 32] or Variational Autoencoders (VAEs) [5, 36]. These detect anomalies using per-pixel reconstruction errors or by evaluating the density obtained from the model's probability distribution. This has been shown to be problematic due to inaccurate reconstructions or poorly calibrated likelihoods [8, 22].

The performance of many supervised computer vision algorithms [16, 34] is improved by transfer learning, i.e. by using discriminative embeddings from pretrained networks. For unsupervised anomaly detection, such approaches have not been thoroughly explored so far. Recent work suggests that these feature spaces generalize well for anomaly detection and even simple baselines outperform generative deep learning approaches [10, 25]. However, the performance of existing methods on large high-resolution image datasets is hampered by the use of shallow machine learning pipelines that require a dimensionality reduction of the used feature space. Moreover, they rely on heavy training data subsampling since their capacity does not suffice to model highly complex data distributions with a large number of training samples.

We propose to circumvent these limitations of shallow models by implicitly modeling the distribution of training features with a student–teacher approach. This leverages the high capacity of deep neural networks and frames anomaly detection as a feature regression problem. Given a descriptive feature extractor pretrained on a large dataset of patches from natural images (the teacher), we train an ensemble of student networks on anomaly-free training
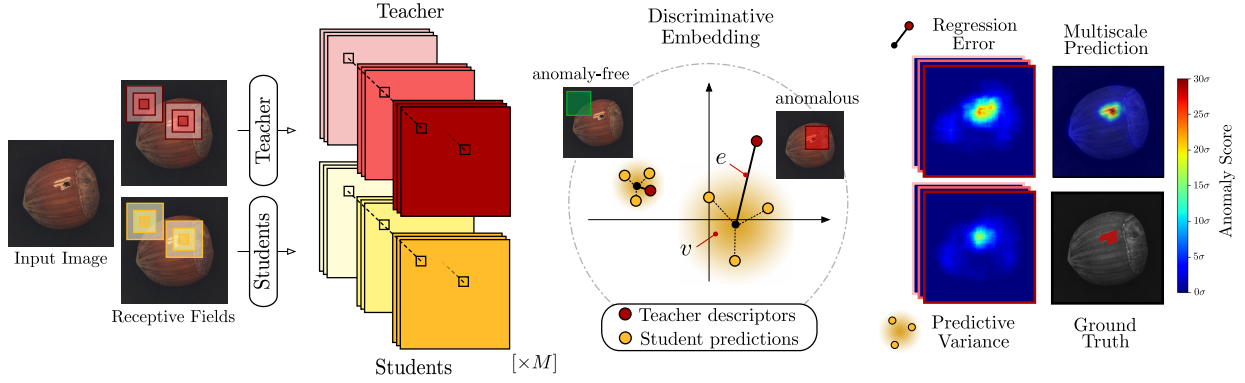
1

Figure 2: Schematic overview of our approach. Input images are fed through a teacher network that densely extracts features for local image regions. An ensemble of M student networks is trained to regress the output of the teacher on anomaly-free data. During inference, the students will yield increased regression errors e and predictive uncertainties v in pixels for which the receptive field covers anomalous regions. Anomaly maps generated with different receptive fields can be combined for anomaly segmentation at multiple scales.

data to mimic the teacher's output. During inference, the students' predictive uncertainty together with their regression error with respect to the teacher are combined to yield dense anomaly scores for each input pixel. Our intuition is that students will generalize poorly outside the manifold of anomaly-free training data and start to make wrong predictions. Figure 1 shows qualitative results of our method when applied to images selected from the MVTec Anomaly Detection dataset [7]. A schematic overview of the entire anomaly detection process is given in Figure 2. Our main contributions are:

- We propose a novel framework for unsupervised anomaly detection based on student–teacher learning. Local descriptors from a pretrained teacher network serve as surrogate labels for an ensemble of students. Our models can be trained end-to-end on large unlabeled image datasets and make use of all available training data.

- We introduce scoring functions based on the students' predictive variance and regression error to obtain dense anomaly maps for the segmentation of anomalous regions in natural images. We describe how to extend our approach to segment anomalies at multiple scales by adapting the students' and teacher's receptive fields.

- We demonstrate state-of-the-art performance on three real-world computer vision datasets. We compare our method to a number of shallow machine learning classifiers and deep generative models that are fitted directly to the teacher's feature distribution. We also compare it to recently introduced deep learning based methods for unsupervised anomaly segmentation.

2. Related Work

There exists an abundance of literature on anomaly detection [27]. Deep learning based methods for the segmentation of anomalies strongly focus on generative models such as autoencoders [1, 8] or GANs [31]. These attempt to learn representations from scratch, leveraging no prior knowledge about the nature of natural images, and segment anomalies by comparing the input image to a reconstruction in pixel space. This can result in poor anomaly detection performance due to simple per-pixel comparisons or imperfect reconstructions [8].

2.1. Anomaly Detection with Pretrained Networks

Promising results have been achieved by transferring discriminative embedding vectors of pretrained networks to the task of anomaly detection by fitting shallow machine learning models to the features of anomaly-free training data. Andrews et al. [3] use activations from different layers of a pretrained VGG network and model the anomaly-free training distribution with a ν-SVM. However, they only apply their method to image classification and do not consider the segmentation of anomalous regions. Similar experiments have been performed by Burlina et al. [10]. They report superior performance of discriminative embeddings compared to feature spaces obtained from generative models.

Nazare et al. [24] investigate the performance of different off-the-shelf feature extractors pretrained on an image classification task for the segmentation of anomalies in surveillance videos. Their approach trains a 1-Nearest-Neighbor (1-NN) classifier on embedding vectors extracted from a large number of anomaly-free training patches. Prior to the training of the shallow classifier, the dimensionality of the network's activations is reduced using Principal Component Analysis (PCA). To obtain a spatial anomaly map during inference, the classifier must be evaluated for a large number of overlapping patches, which quickly becomes a performance bottleneck and results in rather coarse anomaly maps. Similarly, Napoletano et al. [23] extract activations from a pretrained ResNet-18 for a large number of cropped training patches and model their distribution using K-Means clustering after prior dimensionality reduction with PCA. They also perform strided evaluation of test images during inference. Both approaches sample training patches from the input images and therefore do not make use of all possible training features. This is necessary since, in their frame-
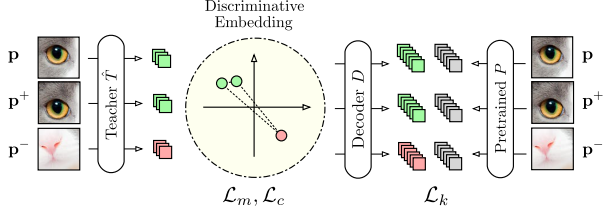
Figure 3: Pretraining of the teacher network $\hat{T}$ to output descriptive embedding vectors for patch-sized inputs. The knowledge of a powerful but computationally inefficient network P is distilled into $\hat{T}$ by decoding the latent vectors to match the descriptors of P. We also experiment with embeddings obtained using self-supervised metric learning techniques based on triplet learning. Information within each feature dimension is maximized by decorrelating the feature dimensions within a minibatch.

Figure 4: Embedding vectors visualized for ten samples of the MNIST dataset. Larger circles around the students' mean predictions indicate increased predictive variance. Being only trained on a single class of training images, the students manage to accurately regress the features solely for this class (green). They yield large regression errors and predictive uncertainties for images of other classes (red). Anomaly scores for the entire dataset are displayed in the bottom histogram.

work, feature extraction is computationally expensive due to the use of very deep networks that output only a single descriptor per patch. Furthermore, since shallow models are employed for learning the feature distribution of anomaly-free patches, the available training information must be strongly reduced.

To circumvent the need for cropping patches and to speed up feature extraction, Sabokrou et al. [30] extract descriptors from early feature maps of a pretrained AlexNet in a fully convolutional fashion and fit a unimodal Gaussian distribution to all available training vectors of anomaly-free images. Even though feature extraction is achieved more efficiently in their framework, pooling layers lead to a downsampling of the input image. This strongly decreases the resolution of the final anomaly map, especially when using descriptive features of deeper network layers with larger receptive fields. In addition, unimodal Gaussian distributions will fail to model the training feature distribution as soon as the problem complexity rises.

2.2. Open-Set Recognition with Uncertainty Estimates

Our work draws some inspiration from the recent success of open-set recognition in supervised settings such as image classification or semantic segmentation, where uncertainty estimates of deep neural networks have been exploited to detect out-of-distribution inputs using MC Dropout [14] or deep ensembles [19]. Seeboeck et al. [33] demonstrate that uncertainties from segmentation networks trained with MC Dropout can be used to detect anomalies in retinal OCT images. Beluch et al. [6] show that the variance of network ensembles trained on an image classification task serves as an effective acquisition function for active learning. Inputs that appear anomalous to the current model are added to the training set to quickly enhance its performance.

Such algorithms, however, demand prior labeling of images by domain experts for a supervised task, which is not always possible or desirable. In our work, we utilize feature vectors of pretrained networks as surrogate labels for the training of an ensemble of student networks. The predictive variance together with the regression error of the ensemble's output mixture distribution is then used as
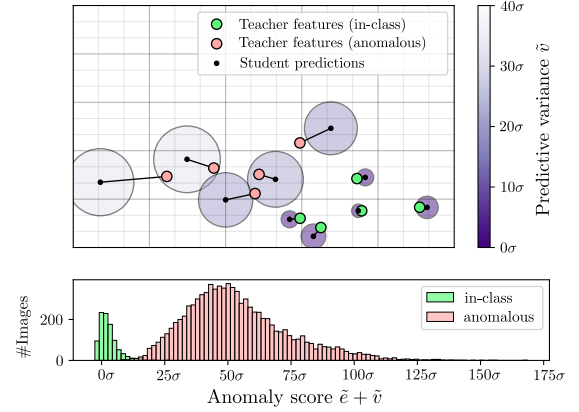
a scoring function to segment anomalous regions in test images.

3. Student–Teacher Anomaly Detection

This section describes the core principles of our proposed method. Given a training dataset $\mathcal{D} = \{\mathbf{I}_1, \mathbf{I}_2, \ldots, \mathbf{I}_N\}$ of anomaly-free images, our goal is to create an ensemble of *student networks* S_i that can later detect anomalies in test images $\mathbf{J}$. This means that they can assign a score to each pixel indicating how much it deviates from the training data manifold. For this, the student models are trained against regression targets obtained from a descriptive *teacher network* T pretrained on a large dataset of natural images. After the training, anomaly scores can be derived for each image pixel from the students' regression error and predictive variance. Given an input image $\mathbf{I} \in \mathbb{R}^{w \times h \times C}$ of width w, height h, and number of channels C, each student S_i in the ensemble outputs a feature map $S_i(\mathbf{I}) \in \mathbb{R}^{w \times h \times d}$. It contains descriptors $\mathbf{y}_{(r,c)} \in \mathbb{R}^d$ of dimension d for each input image pixel at row r and column c. By design, we limit the students' receptive field, such that $\mathbf{y}_{(r,c)}$ describes a square local image region $\mathbf{p}_{(r,c)}$ of $\mathbf{I}$ centered at (r, c) of side length p. The teacher T has the same network architecture as the student networks. However, it remains constant and extracts descriptive embedding vectors for each pixel of the input image $\mathbf{I}$ that serve as deterministic regression targets during student training.

3.1. Learning Local Patch Descriptors

We begin by describing how to efficiently construct a descriptive teacher network T using metric learning and knowledge distillation techniques. In existing work for anomaly detection with pretrained networks, feature extractors only output single feature vectors for patch-sized inputs or spatially heavily downsampled feature

maps [23, 30]. In contrast, our teacher network T efficiently outputs descriptors for every possible square of side length p within the input image. T is obtained by first training a network $\hat{T}$ to embed patch-sized images $\mathbf{p} \in \mathbb{R}^{p \times p \times C}$ into a metric space of dimension d using only convolution and max-pooling layers. Fast dense local feature extraction for an entire input image can then be achieved by a deterministic network transformation of $\hat{T}$ to T as described in [4]. This yields significant speedups compared to previously introduced methods that perform patch-based strided evaluations. To let $\hat{T}$ output semantically strong descriptors, we investigate both self-supervised metric learning techniques as well as distilling knowledge from a descriptive but computationally inefficient pretrained network. A large number of training patches $\mathbf{p}$ can be obtained by random crops from any image database. Here, we use ImageNet [18].

Knowledge Distillation. Patch descriptors obtained from deep layers of CNNs trained on image classification tasks perform well for anomaly detection when modeling their distribution with shallow machine learning models [23, 24]. However, the architectures of such CNNs are usually highly complex and computationally inefficient for the extraction of local patch descriptors. Therefore, we distill the knowledge of a powerful pretrained network P into $\hat{T}$ by matching the output of P with a decoded version of the descriptor obtained from $\hat{T}$:

$$\mathcal{L}_k(\hat{T}) = ||D(\hat{T}(\mathbf{p})) - P(\mathbf{p})||^2. \quad (1)$$

D denotes a fully connected network that decodes the d-dimensional output of $\hat{T}$ to the output dimension of the pretrained network's descriptor.

Metric Learning. If for some reason pretrained networks are unavailable, one can also learn local image descriptors in a fully self-supervised way [12]. Here, we investigate the performance of discriminative embeddings obtained using triplet learning. For every randomly cropped patch $\mathbf{p}$, a triplet of patches $(\mathbf{p}, \mathbf{p}^+, \mathbf{p}^-)$ is augmented. Positive patches $\mathbf{p}^+$ are obtained by small random translations around $\mathbf{p}$, changes in image luminance, and the addition of Gaussian noise. The negative patch $\mathbf{p}^-$ is created by a random crop from a randomly chosen different image. In-triplet hard negative mining with anchor swap [37] is used as a loss function for learning an embedding sensitive to the ℓ_2 metric

$$\mathcal{L}_m(\hat{T}) = \max\{0, \delta + \delta^+ - \delta^-\}, \quad (2)$$

where $\delta > 0$ denotes the margin parameter and in-triplet distances δ^+ and δ^- are defined as:

$$\delta^+ = ||\hat{T}(\mathbf{p}) - \hat{T}(\mathbf{p}^+)||^2 \quad (3)$$

$$\delta^- = \min\{||\hat{T}(\mathbf{p}) - \hat{T}(\mathbf{p}^-)||^2, ||\hat{T}(\mathbf{p}^+) - \hat{T}(\mathbf{p}^-)||^2\}. \quad (4)$$

Descriptor Compactness. As proposed by Vassileios et al. [35], we minimize the correlation between descriptors within one minibatch of inputs $\mathbf{p}$ to increase the descriptors' compactness and remove unnecessary redundancy:

$$\mathcal{L}_c(\hat{T}) = \sum_{i \neq j} c_{ij}, \quad (5)$$

where c_{ij} denotes the entries of the correlation matrix computed over all descriptors $\hat{T}(\mathbf{p})$ in the current minibatch.

The final training loss for $\hat{T}$ is then given as

$$\mathcal{L}(\hat{T}) = \lambda_k \mathcal{L}_k(\hat{T}) + \lambda_m \mathcal{L}_m(\hat{T}) + \lambda_c \mathcal{L}_c(\hat{T}), \quad (6)$$

where λ_k, λ_m, $\lambda_c \geq 0$ are weighting factors for the individual loss terms. Figure 3 summarizes the entire learning process for the teacher's discriminative embedding.

3.2. Ensemble of Student Networks for Deep Anomaly Detection

Next, we describe how to train student networks S_i to predict the teacher's output on anomaly-free training data. We then derive anomaly scores from the students' predictive uncertainty and regression error during inference. First, the vector of component-wise means $\boldsymbol{\mu} \in \mathbb{R}^d$ and standard deviations $\boldsymbol{\sigma} \in \mathbb{R}^d$ over all training descriptors is computed for data normalization. Descriptors are extracted by applying T to each image in the dataset $\mathcal{D}$. We then train an ensemble of $M \geq 1$ randomly initialized student networks S_i, $i \in \{1, \dots, M\}$ that possess the identical network architecture as the teacher T. For an input image $\mathbf{I}$, each student outputs its predictive distribution over the space of possible regression targets for each local image region $\mathbf{p}_{(r,c)}$ centered at row r and column c. Note that the students' architecture with limited receptive field of size p allows us to obtain dense predictions for each image pixel with only a single forward pass, without having to actually crop the patches $\mathbf{p}_{(r,c)}$. The students' output vectors are modeled as a Gaussian distribution $Pr(\mathbf{y}|\mathbf{p}_{(r,c)}) = \mathcal{N}(\mathbf{y}|\boldsymbol{\mu}_{(r,c)}^{S_i}, s)$ with constant covariance $s \in \mathbb{R}$, where $\boldsymbol{\mu}_{(r,c)}^{S_i}$ denotes the prediction made by S_i for the pixel at (r, c). Let $\mathbf{y}_{(r,c)}^T$ denote the teacher's respective descriptor that is to be predicted by the students. The log-likelihood training criterion $\mathcal{L}(S_i)$ for each student network then simplifies to the squared ℓ_2-distance in feature space:

$$\mathcal{L}(S_i) = \frac{1}{wh} \sum_{(r,c)} ||\boldsymbol{\mu}_{(r,c)}^{S_i} - (\mathbf{y}_{(r,c)}^T - \boldsymbol{\mu})\mathrm{diag}(\boldsymbol{\sigma})^{-1}||_2^2, \quad (7)$$

where $\mathrm{diag}(\boldsymbol{\sigma})^{-1}$ denotes the inverse of the diagonal matrix filled with the values in $\boldsymbol{\sigma}$.

Scoring Functions for Anomaly Detection. Having trained each student to convergence, a mixture of Gaussians can be obtained at each image pixel by equally weighting the ensemble's predictive distributions. From it, measures of anomaly can be obtained in two ways: First, we propose to compute the regression error of the mixture's mean $\boldsymbol{\mu}_{(r,c)}$ with respect to the teacher's surrogate label:

$$e_{(r,c)} = ||\boldsymbol{\mu}_{(r,c)} - (\mathbf{y}_{(r,c)}^T - \boldsymbol{\mu})\mathrm{diag}(\boldsymbol{\sigma})^{-1}||_2^2 \quad (8)$$

$$= \left|\left|\frac{1}{M}\sum_{i=1}^M \boldsymbol{\mu}_{(r,c)}^{S_i} - (\mathbf{y}_{(r,c)}^T - \boldsymbol{\mu})\mathrm{diag}(\boldsymbol{\sigma})^{-1}\right|\right|_2^2 \quad (9)$$

The intuition behind this score is that the student networks will fail to regress the teacher's output within anomalous

	Category	Ours $p=65$	1-NN	OC-SVM	K-Means	ℓ_2-AE	VAE	SSIM-AE	AnoGAN	CNN-Feature Dictionary
Textures	Carpet	**0.695**	0.512	0.355	0.253	0.456	0.501	0.647	0.204	0.469
	Grid	0.819	0.228	0.125	0.107	0.582	0.224	**0.849**	0.226	0.183
	Leather	**0.819**	0.446	0.306	0.308	**0.819**	0.635	0.561	0.378	0.641
	Tile	**0.912**	0.822	0.722	0.779	0.897	0.870	0.175	0.177	0.797
	Wood	0.725	0.502	0.336	0.411	**0.727**	0.628	0.605	0.386	0.621
Objects	Bottle	**0.918**	0.898	0.850	0.495	0.910	0.897	0.834	0.620	0.742
	Cable	**0.865**	0.806	0.431	0.513	0.825	0.654	0.478	0.383	0.558
	Capsule	**0.916**	0.631	0.554	0.387	0.862	0.526	0.860	0.306	0.306
	Hazelnut	**0.937**	0.861	0.616	0.698	0.917	0.878	0.916	0.698	0.844
	Metal nut	**0.895**	0.705	0.319	0.351	0.830	0.576	0.603	0.320	0.358
	Pill	**0.935**	0.725	0.544	0.514	0.893	0.769	0.830	0.776	0.460
	Screw	**0.928**	0.604	0.644	0.550	0.754	0.559	0.887	0.466	0.277
	Toothbrush	**0.863**	0.675	0.538	0.337	0.822	0.693	0.784	0.749	0.151
	Transistor	0.701	0.680	0.496	0.399	**0.728**	0.626	0.725	0.549	0.628
	Zipper	**0.933**	0.512	0.355	0.253	0.839	0.549	0.665	0.467	0.703
	Mean	**0.857**	0.640	0.479	0.423	0.790	0.639	0.694	0.443	0.515

Table 1: Results on the MVTec Anomaly Detection dataset. For each dataset category, the normalized area under the PRO-curve up to an average false positive rate per-pixel of 30% is given. It measures the average overlap of each ground-truth region with the predicted anomaly regions for multiple thresholds. The best-performing method for each dataset category is highlighted in boldface.

regions during inference since the corresponding descriptors have not been observed during training. Note that $e_{(r,c)}$ is non-constant even for $M = 1$, where only a single student is trained and anomaly scores can be efficiently obtained with only a single forward pass through the student and teacher network, respectively.

As a second measure of anomaly, we compute for each pixel the predictive uncertainty of the Gaussian mixture as defined by Kendall et al. [14], assuming that the student networks generalize similarly for anomaly-free regions and differently in regions that contain novel information unseen during training:

$$v_{(r,c)} = \frac{1}{M} \sum_{i=1}^{M} ||\boldsymbol{\mu}_{(r,c)}^{S_i}||_2^2 - ||\boldsymbol{\mu}_{(r,c)}||_2^2. \quad (10)$$

To combine the two scores, we compute the means e_μ, v_μ and standard deviations e_σ, v_σ of all $e_{(r,c)}$ and $v_{(r,c)}$, respectively, over a validation set of anomaly-free images. Summation of the normalized scores then yields the final anomaly score:

$$\tilde{e}_{(r,c)} + \tilde{v}_{(r,c)} = \frac{e_{(r,c)} - e_\mu}{e_\sigma} + \frac{v_{(r,c)} - v_\mu}{v_\sigma}. \quad (11)$$

Figure 4 illustrates the basic principles of our anomaly detection method on the MNIST dataset, where images with label 0 were treated as the normal class and all other classes were treated as anomalous. Since the images of this dataset are very small, we extracted a single feature vector for each image using $\hat{T}$ and trained an ensemble of $M = 5$ patch-sized students to regress the teacher's output. This results in a single anomaly score for each input image. Feature descriptors were embedded into 2D using multidimensional scaling [9] to preserve their relative distances.

3.3. Multi-Scale Anomaly Segmentation

If an anomaly only covers a small part of the teacher's receptive field of size p, the extracted feature vector predominantly describes anomaly-free traits of the local image region. Consequently, the descriptor can be predicted well by the students and anomaly detection performance will decrease. One could tackle this problem by downsampling the input image. This would, however, lead to an undesirable loss in resolution of the output anomaly map.

Our framework allows for explicit control over the size of the students' and teacher's receptive field p. Therefore, we can detect anomalies at various scales by training multiple student–teacher ensemble pairs with varying values of p. At each scale, an anomaly map with the same size as the input image is computed. Given L student–teacher ensemble pairs with different receptive fields, the normalized anomaly scores $\tilde{e}_{(r,c)}^{(l)}$ and $\tilde{v}_{(r,c)}^{(l)}$ of each scale l can be combined by simple averaging:

$$\frac{1}{L} \sum_{l=1}^{L} \left(\tilde{e}_{(r,c)}^{(l)} + \tilde{v}_{(r,c)}^{(l)} \right). \quad (12)$$

4. Experiments

To demonstrate the effectiveness of our approach, an extensive evaluation on a number of datasets is performed. We measure the performance of our student–teacher framework against existing pipelines that use shallow machine learning algorithms to model the feature distribution of pretrained networks. To do so, we compare to a K-Means classifier, a One-Class SVM (OC-SVM), and a 1-NN classifier. They are fitted to the distribution of the teacher's descriptors after prior dimensionality reduction using PCA. We also experiment with deterministic and variational autoencoders as deep distribution models over the teacher's discriminative embedding. The ℓ_2-reconstruction error [13] and reconstruction probability [2] are used as the anomaly score, respectively. We further compare our method to recently introduced generative and discriminative deep learning based anomaly detection models and report improved performance over the state of the art. We want to stress that the teacher has not observed images of the evaluated datasets during pretraining to avoid an unfair bias.
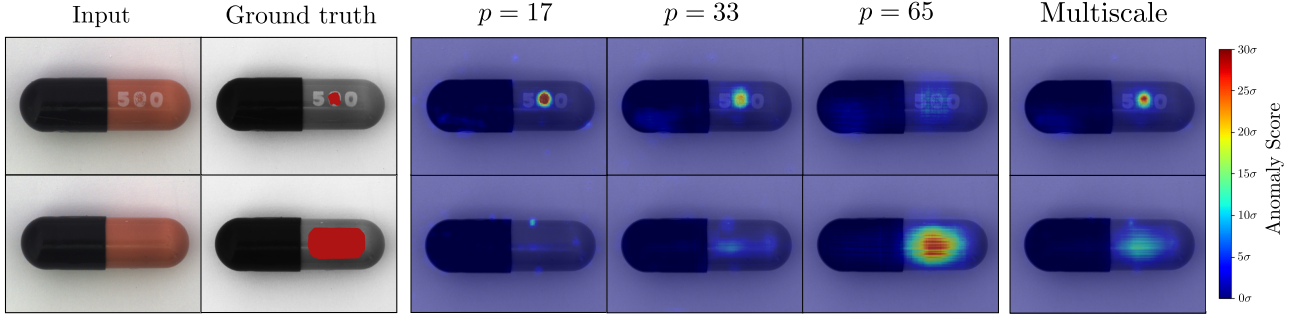
Figure 5: Anomaly detection at multiple scales: Architectures with receptive field of size $p = 17$ manage to accurately segment the small scratch on the capsule (top row). However, defects at a larger scale such as the missing imprint (bottom row) become problematic. For increasingly larger receptive fields, the segmentation performance for the larger anomaly increases while it decreases for the smaller one. Our multiscale architecture mitigates this problem by combining multiple receptive fields.

As a first experiment, we perform an ablation study to find suitable hyperparameters. Our algorithm is applied to a one-class classification setting on the MNIST [20] and CIFAR-10 [17] datasets. We then evaluate on the much more challenging MVTec Anomaly Detection (MVTec AD) dataset, which was specifically designed to benchmark algorithms for the segmentation of anomalous regions. It provides over 5000 high-resolution images divided into ten object and five texture categories. To highlight the benefit of our multi-scale approach, an additional ablation study is performed on MVTec AD, which investigates the impact of different receptive fields on the anomaly detection performance.

For our experiments, we use identical network architectures for the student and teacher networks, with receptive field sizes $p \in \{17, 33, 65\}$. All architectures are simple CNNs with only convolutional and max-pooling layers, using leaky rectified linear units with slope 5×10^{-3} as activation function. Table 4 shows the specific architecture used for $p = 65$. For $p = 17$ and $p = 33$, similar architectures are given in in Appendix A.

For the pretraining of the teacher networks $\hat{T}$, triplets augmented from the ImageNet dataset are used. Images are zoomed to equal width and height sampled from $\{4p, 4p + 1, \ldots, 16p\}$ and a patch of side length p is cropped at a random location. A positive patch $\mathbf{p}^+$ for each triplet is then constructed by randomly translating

the crop location within the interval $\{-\frac{p-1}{4}, \ldots, \frac{p-1}{4}\}$. Gaussian noise with standard deviation 0.1 is added to $\mathbf{p}^+$. All images within a triplet are randomly converted to grayscale with a probability of 0.1. For knowledge distillation, we extract 512-dimensional feature vectors from the fully connected layer of a ResNet-18 that was pretrained for classification on the ImageNet dataset. For network optimization, we use the Adam optimizer [15] with an initial learning rate of 2×10^{-4}, a weight decay of 10^{-5}, and a batch size of 64. Each teacher network outputs descriptors of dimension $d = 128$ and is trained for 5×10^4 iterations.

4.1. MNIST and CIFAR-10

Before considering the problem of anomaly segmentation, we evaluate our method on the MNIST and CIFAR-10 datasets, adapted for one-class classification. Five students are trained on only a single class of the dataset, while during inference images of the other classes must be detected as anomalous. Each image is zoomed to the students' and teacher's input size p and a single feature vector is extracted by passing it through the patch-sized networks $\hat{T}$ and $\hat{S}_i$. We examine different teacher networks by varying the weights $\lambda_k, \lambda_m, \lambda_c$ in the teacher's loss function $\mathcal{L}(\hat{T})$. The patch size for the experiments in this subsection is set to $p = 33$. As a measure of anomaly detection performance, the area under the ROC curve is evaluated. Shallow and deep distributions models are trained on the teacher's descriptors of all available in-distribution samples. We additionally report numbers for OCGAN [26], a recently proposed generative model directly trained on the input images. Detailed information on training parameters for all methods on this dataset is found in Appendix B.

Table 2 shows our results. Our approach outperforms the other methods for a variety of hyperparameter settings. Distilling the knowledge of the pretrained ResNet-18 into the teacher's descriptor yields slightly better performance than training the teacher in a fully self-supervised way using triplet learning. Reducing descriptor redundancy by minimizing the correlation matrix yields improved results. On average, shallow models and autoencoders fitted to our teacher's feature distribution outperform OCGAN but do not reach the performance of our approach. Since for 1-NN, every single training vector can be stored,

Method				MNIST	CIFAR-10
OCGAN [26]				0.9750	0.6566
1-NN				0.9753	0.8189
KMeans				0.9457	0.7592
OC-SVM				0.9463	0.7388
ℓ_2-AE				0.9832	0.7898
VAE				0.9535	0.7502
	L_k	L_m	L_c		
Ours	✓		✓	**0.9935**	**0.8196**
Ours	✓	✓	✓	0.9926	0.8035
Ours		✓	✓	**0.9935**	0.7940
Ours	✓			0.9917	0.8021

Table 2: Results on MNIST and CIFAR-10. For each method, the average area under the ROC curve is given, computed across each dataset category. For our algorithm, we evaluate teacher networks trained with different loss functions. ✓ corresponds to setting the respective loss weight to 1, otherwise it is set to 0.

	Category	$p = 17$	$p = 33$	$p = 65$	Multiscale
Textures	Carpet	0.795	**0.893**	0.695	0.879
	Grid	0.920	0.949	0.819	**0.952**
	Leather	0.935	**0.956**	0.819	0.945
	Tile	0.936	**0.950**	0.912	0.946
	Wood	**0.943**	0.929	0.725	0.911
Objects	Bottle	0.814	0.890	0.918	**0.931**
	Cable	0.671	0.764	**0.865**	0.818
	Capsule	0.935	0.963	0.916	**0.968**
	Hazelnut	0.971	**0.965**	0.937	**0.965**
	Metal nut	0.891	0.928	0.895	**0.942**
	Pill	0.931	0.959	0.935	**0.961**
	Screw	0.915	0.937	0.928	**0.942**
	Toothbrush	**0.946**	0.944	0.863	0.933
	Transistor	0.540	0.611	**0.701**	0.666
	Zipper	0.848	0.942	0.933	**0.951**
	Mean	0.866	0.900	0.857	**0.914**

Table 3: Performance of our algorithm on the MVTec AD dataset for different receptive field sizes p. Combining anomaly scores across multiple receptive fields shows increased performance for many of the dataset's categories. We report the normalized area under the PRO curve up to an average false-positive rate of 30%.

it performs exceptionally well on these small datasets. On average, however, our method still outperforms all evaluated approaches.

4.2. MVTec Anomaly Detection Dataset

For all our experiments on MVTec AD, input images are zoomed to $w = h = 256$ pixels. We train on anomaly-free images for 100 epochs with batch size 1. This is equivalent to training on a large number of patches per batch due to the limited size of the networks' receptive field. We use Adam with initial learning rate 10^{-4} and weight decay 10^{-5}. Teacher networks were trained with $\lambda_k = \lambda_c = 1$ and $\lambda_m = 0$, as this configuration performed best on MNIST and CIFAR-10. Ensembles contain $M = 3$ students.

To train shallow classifiers on the teacher's output descriptors, a subset of vectors is randomly sampled from the teacher's feature maps. Their dimension is then reduced by PCA, retaining 95% of the variance. The variational and deterministic autoencoders are implemented using a simple fully connected architecture and are trained on all available descriptors. In addition to fitting the models directly to the teacher's feature distribution, we benchmark our approach against the best performing deep learning based methods presented by Bergmann et al. [7] on this dataset. These methods include the CNN-Feature Dictionary [23], the SSIM-Autoencoder [8], and AnoGAN [31]. All hyperparameters are listed in detail in Appendix C.

We compute a threshold-independent evaluation metric based on the per-region-overlap (PRO), which weights ground-truth regions of different size equally. This is in contrast to simple per-pixel measures, such as ROC, for which a single large region that is segmented correctly can make up for many incorrectly segmented small ones. It was also used by Bergmann et al. in [7]. For computing the PRO metric, anomaly scores are first thresholded to make a binary decision for each pixel whether an anomaly

Layer	Output Size	Parameters	
		Kernel	Stride
Input	$65 \times 65 \times 3$		
Conv1	$61 \times 61 \times 128$	5×5	1
MaxPool	$30 \times 30 \times 128$	2×2	2
Conv2	$26 \times 26 \times 128$	5×5	1
MaxPool	$13 \times 13 \times 128$	2×2	2
Conv3	$9 \times 9 \times 128$	5×5	1
MaxPool	$4 \times 4 \times 256$	2×2	2
Conv4	$1 \times 1 \times 256$	4×4	1
Conv5	$1 \times 1 \times 128$	3×3	1
Decode	$1 \times 1 \times 512$	1×1	1

Table 4: General outline of our network architecture for training teachers $\hat{T}$ with receptive field size $p = 65$. Leaky rectified linear units with slope 5×10^{-3} are applied as activation functions after each convolution layer. Architectures for $p = 17$ and $p = 33$ are given in Appendix A.

is present or not. For each connected component within the ground truth, the relative overlap with the thresholded anomaly region is computed. We evaluate the PRO value for a large number of increasing thresholds until an average per-pixel false-positive rate of 30% for the entire dataset is reached and use the area under the PRO curve as a measure of anomaly detection performance. Note that for high false-positive rates, large parts of the input images would be wrongly labeled as anomalous and even perfect PRO values would no longer be meaningful. We normalize the integrated area to a maximum achievable value of 1.

Table 1 shows our results training each algorithm with a receptive field of $p = 65$ for comparability. Our method consistently outperforms all other evaluated algorithms for almost every dataset category. The shallow machine learning algorithms fitted directly to the teacher's descriptors after applying PCA do not manage to perform satisfactorily for most of the dataset categories. This shows that their capacity does not suffice to accurately model the large number of available training samples. The same can be observed for the CNN-Feature Dictionary. As it was the case in our previous experiment on MNIST and CIFAR-10, 1-NN yields the best results amongst the shallow models. Utilizing a large number of training features together with deterministic autoencoders increases the performance, but still does not match the performance of our approach. Current generative methods for anomaly segmentation such as Ano-GAN and the SSIM-autoencoder perform similar to the shallow methods fitted to the discriminative embedding of the teacher. This indicates that there is indeed a gap between methods that learn representations for anomaly detection from scratch and methods that leverage discriminative embeddings as prior knowledge.

Table 3 shows the performance of our algorithm for different receptive field sizes $p \in \{17, 33, 65\}$ and when combining multiple scales. For some objects, such as *bottle* and *cable*, larger receptive fields yield better results. For others, such as *wood* and *toothbrush*, the inverse behavior can be observed. Combining multiple scales enhances the performance for many of the dataset categories. A qualitative example highlighting the benefit of our multi-scale anomaly segmentation is visualized in Figure 5.

Layer	Output Size	Parameters	
		Kernel	Stride
Input	33×33×3		
Conv1	29×29×128	3×3	1
MaxPool	14×14×128	2×2	2
Conv2	10×10×256	5×5	1
MaxPool	5×5×256	2×2	2
Conv3	4×4×256	2×2	1
Conv4	1×1×128	4×4	1
Decode	1×1×512	1×1	1

(a) Architecture for $p = 33$.

Layer	Output Size	Parameters	
		Kernel	Stride
Input	17×17×3		
Conv1	12×12×128	5×5	1
Conv2	8×8×256	5×5	1
Conv3	4×4×256	5×5	1
Conv4	1×1×128	4×4	1
Decode	1×1×512	1×1	1

(b) Architecture for $p = 17$.

Table 5: Network architectures for teacher networks $\hat{T}$ with different receptive field sizes p.

5. Conclusion

We have proposed a novel framework for the challenging problem of unsupervised anomaly segmentation in natural images. Anomaly scores are derived from the predictive variance and regression error of an ensemble of student networks, trained against embedding vectors from a descriptive teacher network. Ensemble training can be performed end-to-end and purely on anomaly-free training data without requiring prior data annotation. Our approach can be easily extended to detect anomalies at multiple scales. We demonstrate improvements over current state-of-the-art methods on a number of real-world computer vision datasets for one-class classification and anomaly segmentation.

Appendix A.
Network Architectures

A description of the network architecture for a patch-sized teacher network $\hat{T}$ with receptive field of size $p = 65$ can be found in our main paper (Table 4). Architectures for teachers with receptive field sizes $p = 33$ and $p = 17$ are depicted in Tables 5a and 5b, respectively. Leaky rectified linear units with slope 5×10^{-3} are used as activation function after each convolution layer.

Appendix B.
Experiments on MNIST and CIFAR-10

Here, we give details about additional hyperparameters for our experiments on the MNIST and CIFAR-10 datasets. We additionally provide the per-class ROC-AUC values for the two datasets in Tables 6 and 7, respectively.

Hyperparameter Settings. For the deterministic ℓ_2-autoencoder (ℓ_2-AE) and the variational autoencoder (VAE), we use a fully connected encoder architecture of shape 128–64–32–10 with leaky rectified linear units of slope 5×10^{-3}. The decoder is constructed in a manner symmetric to the encoder. Both autoencoders are trained for 100 epochs at an initial learning rate of 10^{-2} using the Adam optimizer and a batch size of 64. A weight decay rate of 10^{-5} is applied for regularization. To evaluate the reconstruction probability of the VAE, five independent forward passes are performed for each feature vector. For the One-Class SVM (OC-SVM), a radial basis function kernel is used. K-Means is trained with 10 cluster centers and the distance to the single closest cluster center is

evaluated as the anomaly score for each input sample. For 1-NN, the feature vectors of all available training samples are stored and tested during inference.

Appendix C.
Experiments on MVTec AD

We give additional information on the hyperparameters used in our experiments on MVTec AD for both shallow machine learning models as well as deep learning methods.

Shallow Machine Learning Models. For the 1-NN classifier, we construct a dictionary of 5000 feature vectors and take the distance to the closest training sample as anomaly score. For the other shallow classifiers, we fit their parameters on 50 000 training samples, randomly chosen from the teacher's feature maps. The K-Means algorithm is run with 10 cluster centers and measures the distance to the nearest cluster center in the feature space during inference. The OC-SVM employs a radial basis function kernel.

Deep-Learning Based Models. For evaluation on MVTec AD, the architecture of the ℓ_2-AE and VAE are identical to the ones used on the MNIST and CIFAR-10 dataset. Each fully connected autoencoder is trained for 100 epochs. We use Adam with initial learning rate 10^{-4} and weight decay 10^{-5}. Batches are constructed from 512 randomly sampled vectors of the teacher's feature maps. The reconstruction probability of the VAE is computed by five individual forward passes through the network. For the evaluation of AnoGAN, the SSIM-Autoencoder, and the CNN-Feature Dictionary, we use the same hyperparameters as Bergmann et al. in the MVTec AD dataset paper [7]. Only a slight adaption is applied to the CNN-Feature Dictionary by cropping patches of size $p = 65$ and performing the evaluation by computing anomaly scores for overlapping patches with a stride of 4 pixels.

Qualitative Results. We provide additional qualitative results of our method on MVTec AD for three objects and three textures in Figure 6. For each category, anomaly maps for multiple defect classes are provided. Our method performs well across different defect types and sizes. The results are shown for an ensemble of 3 students and a multi-scale architecture of receptive field sizes in $\{17, 33, 65\}$ pixels.
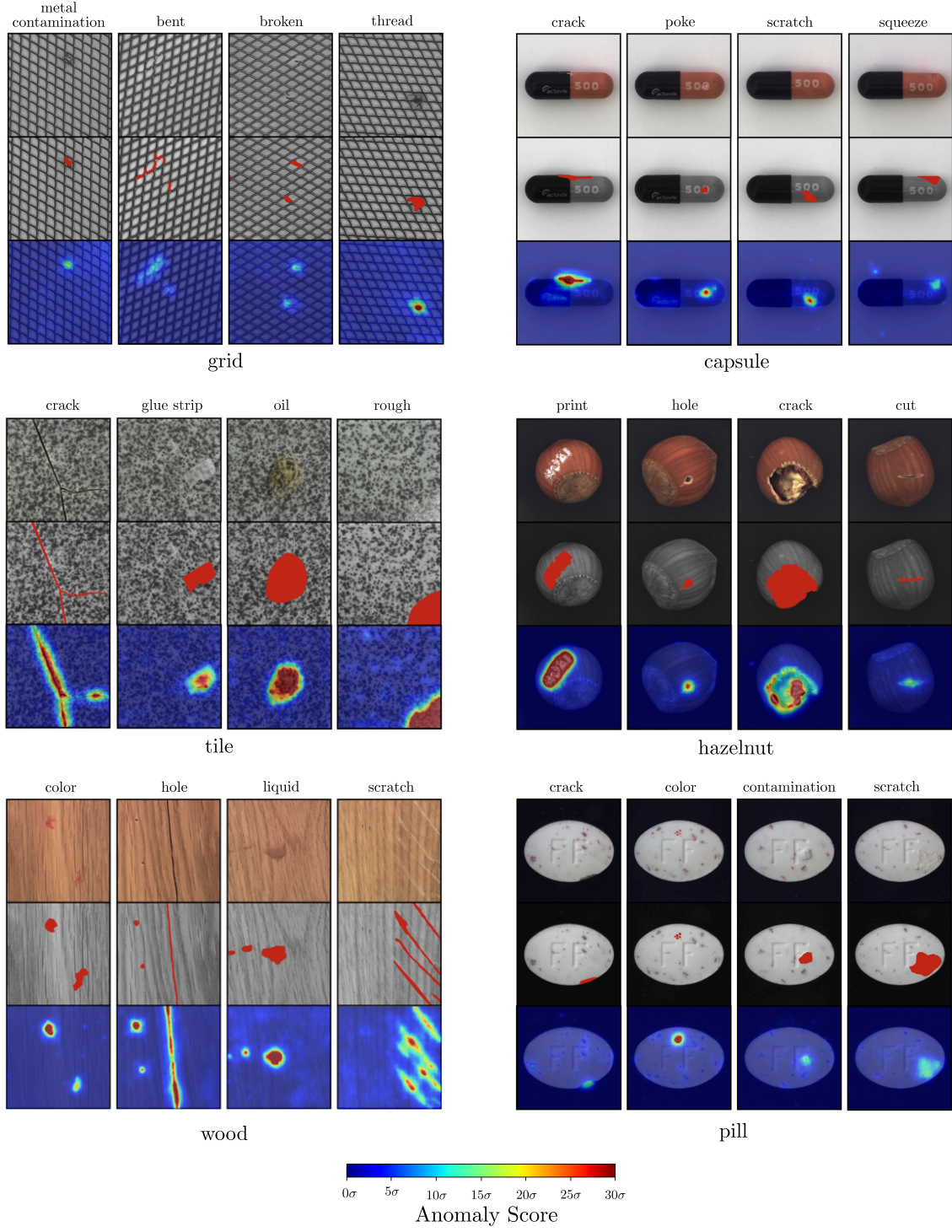
Figure 6: Qualitative results of our method on selected textures (left) and objects (right) of the MVTec Anomaly Detection dataset. Our algorithm performs robustly across various defect categories, such as color defects, contaminations, and structural anomalies. **Top row:** Input images containing defects. **Center row:** Ground truth regions of defects in red. **Bottom row:** Anomaly scores for each image pixel predicted by our algorithm.

Method	L_k	L_m	L_c	0	1	2	3	4	5	6	7	8	9	Mean
OCGAN				0.998	**0.999**	0.942	0.963	0.975	0.980	0.991	0.981	0.939	0.981	0.9750
1-NN				0.989	0.998	0.962	0.970	0.980	0.955	0.979	0.981	0.968	0.971	0.9753
KMeans				0.973	0.995	0.898	0.948	0.960	0.920	0.948	0.948	0.940	0.927	0.9457
OC-SVM				0.980	0.998	0.887	0.944	0.964	0.909	0.949	0.957	0.935	0.940	0.9463
ℓ_2-AE				0.992	**0.999**	0.967	0.980	0.988	0.970	0.988	0.987	0.978	0.983	0.9832
VAE				0.983	0.998	0.915	0.941	0.969	0.925	0.964	0.940	0.955	0.945	0.9535
Ours	✓		✓	**0.999**	**0.999**	0.990	**0.993**	**0.992**	**0.993**	**0.997**	**0.995**	0.986	0.991	**0.9935**
Ours	✓	✓	✓	**0.999**	**0.999**	0.988	0.992	0.988	**0.993**	**0.997**	**0.995**	0.984	0.991	0.9926
Ours		✓	✓	**0.999**	**0.999**	0.992	0.992	0.988	**0.993**	**0.997**	**0.995**	0.988	0.992	**0.9935**
Ours	✓			**0.999**	**0.999**	0.989	0.990	0.990	0.990	**0.997**	0.993	0.981	0.989	0.9917

Table 6: Results on the MNIST dataset. For each method and digit, the area under the ROC curve is given. For our algorithm, we evaluate teacher networks trained with different loss functions. ✓ corresponds to setting the respective loss weight to 1, otherwise it is set to 0.

Method	L_k	L_m	L_c	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck	Mean
OCGAN				0.757	0.531	0.640	0.620	0.723	0.620	0.723	0.575	0.820	0.554	0.6566
1-NN				0.792	**0.860**	**0.746**	0.729	0.815	**0.797**	0.876	**0.836**	0.856	**0.882**	0.8189
KMeans				0.673	0.822	0.665	0.676	0.742	0.746	0.828	0.780	0.817	0.843	0.7592
OC-SVM				0.651	0.785	0.618	0.679	0.733	0.730	0.797	0.760	0.799	0.836	0.7388
ℓ_2-AE				0.747	0.862	0.690	0.698	0.788	0.759	0.849	0.824	0.812	0.869	0.7898
VAE				0.705	0.819	0.605	0.700	0.734	0.731	0.797	0.751	0.801	0.859	0.7502
Ours	✓		✓	0.789	0.849	0.734	**0.748**	**0.851**	0.793	**0.892**	0.830	**0.862**	0.848	**0.8196**
Ours	✓	✓	✓	0.784	0.836	0.706	0.742	0.826	0.768	0.870	0.815	0.857	0.831	0.8035
Ours		✓	✓	**0.804**	0.855	0.706	0.709	0.798	0.738	0.860	0.797	0.849	0.824	0.7940
Ours	✓			0.766	0.817	0.715	0.736	0.855	0.763	0.885	0.819	0.838	0.827	0.8021

Table 7: Results on the CIFAR-10 dataset. For each method and class, the area under the ROC curve is given. For our algorithm, we evaluate teacher networks trained with different loss functions. ✓ corresponds to setting the respective loss weight to 1, otherwise it is set to 0.

References

[1] D. Abati, A. Porrello, S. Calderara, and R. Cucchiara. Latent space autoregression for novelty detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 481–490, June 2019. doi: 10.1109/CVPR.2019.00057.

[2] Jinwon An and Sungzoon Cho. Variational Autoencoder based Anomaly Detection using Reconstruction Probability. *SNU Data Mining Center, Tech. Rep.*, 2015.

[3] Jerone TA Andrews, Thomas Tanay, Edward J Morton, and Lewis D Griffin. Transfer Representation-Learning for Anomaly Detection. In *Anomaly Detection Workshop at ICML2016*, 2016.

[4] Christian Bailer, Tewodros A Habtegebrial, Kiran Varanasi, and Didier Stricker. Fast Dense Feature Extraction with CNNs that have Pooling or Striding Layers. In *British Machine Vision Conference (BMVC)*, 2017.

[5] Christoph Baur, Benedikt Wiestler, Shadi Albarqouni, and Nassir Navab. Deep Autoencoding Models for Unsupervised Anomaly Segmentation in Brain MR Images. *arXiv preprint arXiv:1804.04488*, 2018.

[6] William H. Beluch, Tim Genewein, Andreas Nürnberger, and Jan M. Köhler. The power of ensembles for active learning in image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[7] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. MVTec AD – A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9592–9600, 2019.

[8] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving Unsupervised Defect Segmentation by Applying Structural Similarity to Autoencoders. *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, February 2019.

[9] Ingwer Borg and Patrick Groenen. Modern multidimensional scaling: Theory and applications. *Journal of Educational Measurement*, 40(3):277–280, 2003.

[10] Philippe Burlina, Neil Joshi, and I-Jeng Wang. Where's Wally Now? Deep Generative and Discriminative Embeddings for Novelty Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

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

[12] Dov Danon, Hadar Averbuch-Elor, Ohad Fried, and Daniel Cohen-Or. Unsupervised natural image patch learning. *Computational Visual Media*, 5(3):229–237, Sep 2019.

[13] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 1735–1742. IEEE, 2006.

[14] Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer

Vision? In *Advances in Neural Information Processing Systems 30*, pages 5574–5584, 2017.

[15] Diederik P Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *3rd International Conference for Learning Representations*, 2015.

[16] Simon Kornblith, Jonathon Shlens, and Quoc V. Le. Do better imagenet models transfer better? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

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

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification With Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[19] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In *Advances in Neural Information Processing Systems 30*, pages 6402–6413, 2017.

[20] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

[21] Marc Masana, Idoia Ruiz, Joan Serrat, Joost van de Weijer, and Antonio M Lopez. Metric Learning for Novelty and Anomaly Detection. In *British Machine Vision Conference (BMVC)*, 2018.

[22] Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do Deep Generative Models Know What They Don't Know? *arXiv preprint arXiv:1810.09136*, 2018.

[23] Paolo Napoletano, Flavio Piccoli, and Raimondo Schettini. Anomaly Detection in Nanofibrous Materials by CNN-Based Self-Similarity. *Sensors*, 18(1):209, 2018.

[24] Tiago S Nazare, Rodrigo F de Mello, and Moacir A Ponti. Are pre-trained cnns good feature extractors for anomaly detection in surveillance videos? *arXiv preprint arXiv:1811.08495*, 2018.

[25] Pramuditha Perera and Vishal M. Patel. Deep transfer learning for multiple class novelty detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[26] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. OCGAN: One-class novelty detection using GANs with constrained latent representations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[27] Marco AF Pimentel, David A Clifton, Lei Clifton, and Lionel Tarassenko. A review of novelty detection. *Signal Processing*, 99:215–249, 2014.

[28] Alina Roitberg, Ziad Al-Halah, and Rainer Stiefelhagen. Informed democracy: Voting-based novelty detection for action recognition. In *29th British Machine Vision Conference: BMVC 2018, Northumbria University, Newcastle, UK, 3-6 September 2018*. BMVA Press, Durham, 2019. URL https://arxiv.org/pdf/1810.12819.pdf.

[29] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4393–4402, 2018.

[30] Mohammad Sabokrou, Mohsen Fayyaz, Mahmood Fathy, Zahra Moayed, and Reinhard Klette. Deepanomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes. *Computer Vision and Image Understanding*, 172:88–97, 2018.

[31] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. In *International Conference on Information Processing in Medical Imaging*, pages 146–157. Springer, 2017.

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

[33] Philipp Seebock, José Ignacio Orlando, Thomas Schlegl, Sebastian M Waldstein, Hrvoje Bogunovic, Sophie Klimscha, Georg Langs, and Ursula Schmidt-Erfurth. Exploiting Epistemic Uncertainty of Anatomy Segmentation for Anomaly Detection in Retinal OCT. *IEEE transactions on medical imaging*, 2019.

[34] Ruoqi Sun, Xinge Zhu, Chongruo Wu, Chen Huang, Jianping Shi, and Lizhuang Ma. Not all areas are equal: Transfer learning for semantic segmentation via hierarchical region selection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[35] Yurun Tian, Bin Fan, and Fuchao Wu. L2-Net: Deep Learning of Discriminative Patch Descriptor in Euclidean Space. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6128–6136, 2017.

[36] A. Vasilev, V. Golkov, M. Meissner, I. Lipp, E. Sgarlata, V. Tomassini, D.K. Jones, and D. Cremers. q-Space Novelty Detection with Variational Autoencoders. *MICCAI 2019 International Workshop on Computational Diffusion MRI*, 2019.

[37] Daniel Ponsa Vassileios Balntas, Edgar Riba and Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 119.1–119.11, September 2016.