Multiresolution Knowledge Distillation for Anomaly Detection

Mohammadreza Salehi, Niousha Sadjadi*, Soroosh Baselizadeh*, Mohammad H. Rohban, Hamid R. Rabiee
Department of Computer Engineering, Sharif University of Technology
(smrsalehi, nsadjadi, baselizadeh)@ce.sharif.edu, (rohban, rabiee)@sharif.edu

Abstract

Unsupervised representation learning has proved to be a critical component of anomaly detection/localization in images. The challenges to learn such a representation are two-fold. Firstly, the sample size is not often large enough to learn a rich generalizable representation through conventional techniques. Secondly, while only normal samples are available at training, the learned features should be discriminative of normal and anomalous samples. Here, we propose to use the "distillation" of features at various layers of an expert network, which is pre-trained on ImageNet, into a simpler cloner network to tackle both issues. We detect and localize anomalies using the discrepancy between the expert and cloner networks' intermediate activation values given an input sample. We show that considering multiple intermediate hints in distillation leads to better exploitation of the expert's knowledge and a more distinctive discrepancy between the two networks, compared to utilizing only the last layer activation values. Notably, previous methods either fail in precise anomaly localization or need expensive region-based training. In contrast, with no need for any special or intensive training procedure, we incorporate interpretability algorithms in our novel framework to localize anomalous regions. Despite the striking difference between some test datasets and ImageNet, we achieve competitive or significantly superior results compared to SOTA on MNIST, F-MNIST, CIFAR-10, MVTecAD, Retinal-OCT, and two other medical datasets on both anomaly detection and localization.

1. Introduction

Anomaly detection (AD) aims for recognizing test-time inputs that look abnormal or novel to the model according to previously seen normal samples during training. AD has been a vital demanding task in computer vision with various applications, like in industrial image-based product quality control [27, 7], or health monitoring processes [26]. These tasks also require pixel-precise localization of anomalous regions, called defects. This is pivotal for comprehending the dynamics of monitored procedures, triggering the apt antidotes, and providing pertinent data for downstream models in industrial settings.

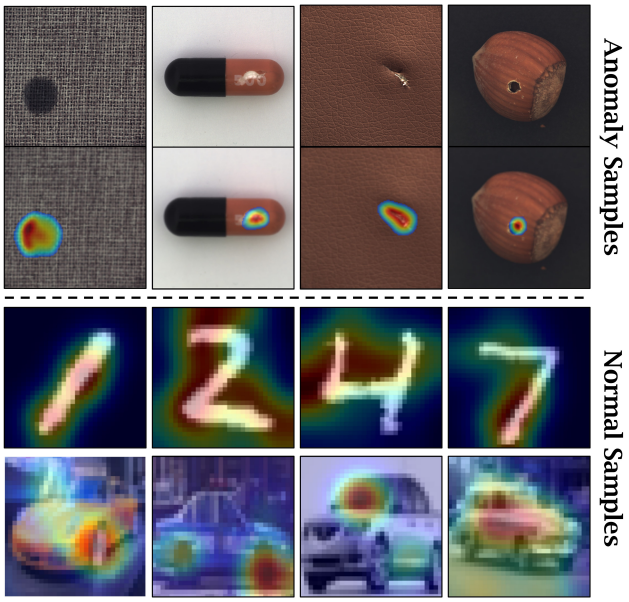

Figure 1: Our precise heatmaps are localizing anomalous features in MVTecAD (top two rows) and normal features in MNIST and CIFAR-10 (two bottom rows).

Traditionally, the AD problem has been approached in a one-class setting, where anomalies represent a broadly different class from normal samples. Recently, considering subtle anomalies has attracted attention. This new setting further necessitates precise anomaly localization. However, performing excellently in both settings on various datasets is highly appreciated but is not fully achieved.

Due to the unsupervised nature of the AD problem and the restricted data access, only having anomaly-free data in training, the majority of AD methods [36, 31, 40, 18, 34] model the normal data abstraction by extracting semantically meaningful latent features. These methods perform

* Denotes equal contribution.

well solely on either of the two mentioned cases. This problem, called the *generality* problem [39], highly declines trust in them on unseen future datasets. Moreover, anomaly localization is either impossible inadequate in most of them [36, 31, 33] and leads to intensive computations that hurt their real-time performance. Additionally, many earlier works [33, 31] suffer from unstable training, requiring unprincipled early stopping to achieve acceptable results.

Though not fully explored in the AD context, using pre-trained networks could potentially be an alternative track. This is especially helpful when the sample size is small and the normal class shows significant variations. Some earlier studies [4, 12, 28, 29] try to train their model based on pre-trained features of the normal data. These methods either miss anomaly localization [4, 12], or tackle the problem in a region-based fashion [28, 53], i.e., splitting images into smaller patches to determine the sub-regional abnormality. This is computationally expensive and often leads to inaccurate localization. To evade this issue, Bergmann *et al*. [8] train an ensemble of student networks to mimic the *last layer* of a teacher network on the anomaly-free data. However, performing a region-based approach in this work makes it heavily rely on the size of the cropped patches and hence susceptible to the changes in this size, and intensifies the training cost severely. Furthermore, imitating only the last layer misses fully exploiting the knowledge of the teacher network [32]. This makes them complicate their model and employ other complementary techniques, such as self-supervised learning in parallel.

Lately, Zhang *et al*. [52] have demonstrated that activation values of intermediate layers of neural networks are a solid perceptual representation of their input images. By this premise, we propose a novel knowledge distillation method for AD that is designed to *distill* the *comprehensive* knowledge of an ImageNet pre-trained *source* network, *solely* on the normal training data, into a simpler *cloner* network. This happens by forcing the cloner's *intermediate* embedding of normal training data at *several critical layers* to conform to those of the source. Consequently, the cloner learns the normal data manifold thoroughly and yet earns no knowledge from the source about other possible input data. Hence, the cloner will behave differently from the source when fed with the anomalous data. Furthermore, a simpler cloner architecture enables avoiding distraction by non-distinguishing features and enhances the discrepancy in the behavior of the two networks on anomalies.

Moreover, we derive precise anomaly localization heat maps without using region-based expensive training and testing through exploiting the concept of gradient. We evaluate our method on a comprehensive set of datasets on various anomaly detection/localization tasks, where we exceed SOTA in both localization and detection. Our training is highly stable and needs no dataset-dependent fine-tuning. As we only train the cloner's parameters, we require just *one* more *forward* pass of inputs through the source compared to a standard network training on the normal data. We also investigate our method through exhaustive ablation studies. Our main contributions are summarized as follows:

1. Enabling a more comprehensive transfer of the knowledge of the pre-trained expert network to the cloner one. Distilling the knowledge into a *more compact* network also helps to concentrate solely on the features that are distinguishing normal vs. anomalous.

2. Our method has a computationally inexpensive and stable training process compared to the earlier work.

3. Our method allows a real-time and precise anomaly localization based on computing gradients of the discrepancy loss concerning the input.

4. Conducting a considerable number of diverse experiments and outperforming previous SOTA models by a *large* margin on many datasets and yet staying competitive on the rest.

2. Related Work

Previous Methods: Autoencoder (AE)-based methods use the idea that abnormal inputs are not reconstructed as precisely as normal ones by learning normal latent features. Hence, anomalous samples will have higher reconstruction errors than normal ones. To better learn these normal latent features, LSA [1] trains an autoregressive model at its latent space, and OC-GAN [31] attempts to force abnormal inputs to be reconstructed as normal ones. These methods fail on industrial or complex datasets [38]. SSIM-AE [10] trains an AE with the SSIM loss [54] instead of MSE, causing it to perform just better on defect segmentation. Gradient-based VAE [15] introduces an energy criterion, which is minimized at test-time by an iterative procedure. Both of the latter methods do not perform well on one-class settings, such as CIFAR-10 [23].

GAN-based approaches, like AnoGan [41], f-AnoGan [40], and GANomaly [3], attempt to find a specific latent space where the generator's reconstructions, obtained from samplings of this space, are analogous to normal samples. f-AnoGan and GANomaly add an extra encoder to the generator to reduce the inference time of AnoGan. Despite their acceptable performance in localization and detection of subtle anomalies, they fail in one-class settings.

Methods like uninformed-students [9], GT[18], and DSVDD [33] keep only the valuable information of normal data by building a compact latent feature space, in contrast to AE-based ones that try to miss the least amount of normal

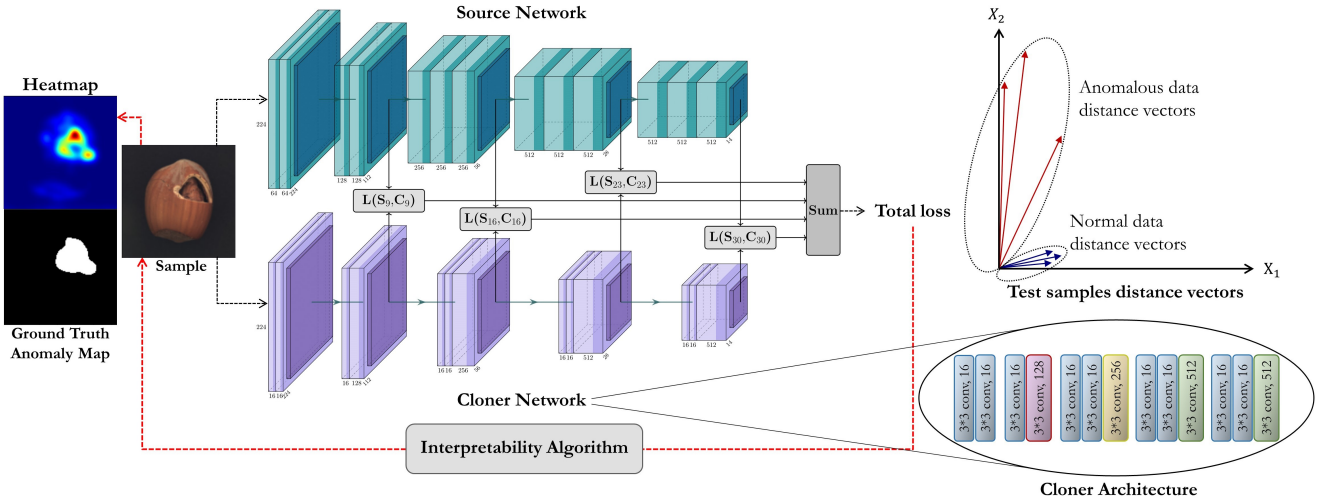

Figure 2: Visualized summary of our proposed framework. A smaller cloner network, C, is trained to imitate the *whole* behavior of a source network, S (VGG-16), on normal data. The discrepancy of their intermediate behavior is formulated by a total loss function and is used to detect anomalies at the test time. A hypothetical example of distance vectors between the activations of C and S on anomalous and normal data is also depicted. Interpretability algorithms are employed to yield pixel-precise anomaly localization maps.

data information. To achieve this, they use self-supervised learning methods or one-class techniques. However, since we only have access to normal samples in an unsupervised setting, the optimization here is more challenging than in AE-based methods and usually converges to trivial solutions. Unprincipled early stopping is used to solve this issue that lowers the trust in these models on unseen future datasets. For example, GT fails on subtle anomaly datasets like MVTecAD while performs well on one-class settings.

Using Pre-trained Features: Some previous methods use pre-trained VGG last layer to solve the representation problem [14, 35]. However, [14] sticks in bad local minima as it uses only the last layer. [35] attempts to solve this by extracting lots of different patches from normal images. Then, it fits a Gaussian distribution on the VGG extracted embeddings of the patches. Although this might alleviate the problem, they fail to provide sound localization or detection on diverse datasets because of using the unimodal Gaussian distribution and hand-engineered size of patches.

Interpretability: Interpretability methods inspect the contribution of input elements to a deep network. Gradient-based ones compute pixels importance using gradients as a proxy. While Gradients [42] uses rough gradients, GuidedBackprop (GBP) [45] filters out negative backpropagated gradients to only consider positively contributing elements. As Gradients' maps can be noisy, SmoothGrad [44] adds noises to the input and averages the maps obtained for each noisy input by Gradients. [2, 30] reveal GBP's flaws by showing that it reconstructs the image instead of explaining the outcome function.

3. Method

3.1. Our Approach

Given a training dataset $D_{train} = \{x_1, ..., x_n\}$ consisting only of normal images (i.e., no anomalies in them), we aim to train a *cloner* network, C, that detects anomalous images in the test set, D_{test}, and localizes anomalies in those images with the help of a pre-trained network. As C needs to predict each sample's deviation from the manifold of normal data, it needs to know the manifold quite well. Therefore, it is trained to mimic the *comprehensive* behavior of an expert network, called the *source* network S. Earlier works in knowledge distillation have conducted tremendous efforts to transfer one network knowledge to another smaller one to save computational cost and memory usage. Many of them strive to teach just the output of S to C. We, however, aim to transfer the intermediate knowledge of S on the normal training data to C as well.

In [32], it is shown that by using a single intermediate-level hint from the source, a thinner but deeper cloner even outperforms the source on classification tasks. In this work, we provide C with multiple intermediate hints from S by encouraging C to learn S's knowledge on normal samples through conforming its intermediate representations in several *critical layers* to S's representations. It is known that layers of neural networks correspond to features at various abstraction levels. For instance, first layer filters act as simple edge detectors. They represent more semantic features when considering later layers. Therefore, mimicking different layers educates C in various abstraction levels, which

leads to a more thorough final understanding of the normal data. In contrast, using only the last layer shares a small portion of S's knowledge with C. Besides, this causes the optimization to stuck in irrelevant local minima, especially when dealing with subtle anomalies that share almost all semantic concepts. On the contrary, using several intermediate hints turns the ill-posed problem into a more well-posed one. The effect of considering different layers in our method is more investigated in Sec. 3.3.1.

In what follows, we refer to the i-th critical layer in the networks as CP_i (CP_0 stands for the raw input) and the source activation values of that critical layer as $a_s^{CP_i}$, and the cloner's ones as $a_c^{CP_i}$. As discussed in the knowledge distillation literature [32, 50], the notion of knowledge can be seen as the value of activation functions. We define the notion of knowledge as both the value and direction of all a^{CP_i}s to intensify the full knowledge transfer from S to C. Hence, we define two losses, $\mathcal{L}_{val}$ and $\mathcal{L}_{dir}$ to represent each aspect. The first, $\mathcal{L}_{val}$, aims to minimize the Euclidean distance between C's and S's activation values at each CP_i. Thus, $\mathcal{L}_{val}$ is formulated as

$$\mathcal{L}_{val} = \sum_{i=1}^{N_{CP}} \frac{1}{N_i} \sum_{j=1}^{N_i} (a_s^{CP_i}(j) - a_c^{CP_i}(j))^2, \quad (1)$$

where N_i indicates the number of neurons in layer CP_i and $a^{CP_i}(j)$ is the value of j-th activation in layer CP_i. N_{CP} represents the total number of critical layers.

Additionally, we use $\mathcal{L}_{dir}$ to increase the directional similarity between the activation vectors. This is more vital in ReLU networks whose neurons are activated only after exceeding a zero value threshold. This indicates that two activation vectors with the same Euclidean distance from the target vector, may have contrasting behaviors in activating a following neuron. For instance, for e being a positive number, let $a_1 = (0, 0, e, 0, \dots, 0) \in \mathbb{R}^k, a_2 = (0, (\sqrt{2}+1)e, 0, 0, \dots, 0) \in \mathbb{R}^k$ be activation vectors of two disparate cloner networks both trying to mimic the activation vector of a source network, a^*, defined as $a^* = (0, e, 0, 0, \dots, 0) \in \mathbb{R}^k$. It is clear a_1 and a_2 have the same Euclidean distance from a^*. However, assuming $W = (0, 1, \dots, 0, 0)$ as the weight vector of a neuron in the next layer of the network, we have

$$\begin{aligned} W^T a_1 &= 0 \leq 0, \\ W^T a_2 &= (\sqrt{2}+1)e > 0, \\ W^T a^* &= e > 0. \end{aligned} \quad (2)$$

This means that the corresponding ReLU neuron would be activated by a_2, similar to a^*, while deactivated by a_1. To address this, using the cosine similarity metric, we define the $\mathcal{L}_{dir}$ as

$$\mathcal{L}_{dir} = \sum_i 1 - \frac{vec(a_s^{CP_i})^T \cdot vec(a_c^{CP_i})}{\left\| vec(a_s^{CP_i}) \right\| \left\| vec(a_c^{CP_i}) \right\|}, \quad (3)$$

where $vec(x)$ is a vectorization function transforming a matrix x with arbitrary dimensions into a 1-D vector. This encourages the activation vector of C be not only close to the S's one in terms of Euclidean distance but also be in the same direction. Note that $\mathcal{L}_{dir}$ is 1 for a_1, and is 0 for a_2. The role of $\mathcal{L}_{dir}$ and $\mathcal{L}_{val}$ is more elaborated in Sec. 3.3.3. Using the two aforementioned losses, $\mathcal{L}_{total}$ is formulated as

$$\mathcal{L}_{total} = \mathcal{L}_{val} + \lambda \mathcal{L}_{dir}, \quad (4)$$

where λ is set to make the scale of both constituent terms the same. For this, we find the initial amount of error for each term on the untrained network and set λ with respect to it. Training using $\mathcal{L}_{total}$, unlike many other methods [18, 6], continues to fully converge, which is the only accessible criterion to measure when to stop training epochs.

Moreover, the architecture of C is designed to be simpler than S to enable knowledge "distillation." This compression of the network facilitates the concentration on normal main features. While the source needs to be a very deep and wide model to learn all necessary features to perform well on a large-scale domain dataset, like ImageNet [16], the goal of the cloner is simply acquiring the source's knowledge of the normal data. Hence, superfluous filters are only detrimental by focusing on non-distinguishing features, present in normal and anomalous data. Compressing the source prevents such distractions for the model. This is more effective when the normal class' boundary and abnormal samples are extremely close, e.g., MVTecAD screw class. We investigate this effect more in Sec. 3.3.2.

Anomaly Detection: To detect anomalous samples, each test input is fed to both S and C. As S has only taught the normal point of view to C, anomalies, inputs out of the normal manifold, are a potential surprise for C. In contrast, S has insights about images out of the normal manifold as well. All this leads to a potential discrepancy in their behavior for anomalous inputs, which could be detected by thresholding the loss in Eq. 4.

Anomaly Localization: [15, 58] have shown that derivative of the loss function with respect to the input has meaningful information about the significance of each pixel for the loss value. Hence, we employ the gradients of $\mathcal{L}_{total}$ with respect to the input to find the most impactful pixels on the loss function, i.e., anomalous regions in anomalous samples. To obtain our localization map for an input x, we first acquire the attribution map, Λ, by

$$\Lambda = \frac{\partial \mathcal{L}_{total}}{\partial x}. \quad (5)$$

To reduce the natural noises in these maps, we induce Gaussian blur and opening morphological filter on Λ. Hence, the localization map, L_{map}, is achieved by

$$\begin{aligned} M &= g_\sigma(\Lambda), \\ L_{map} &= (M \ominus B) \oplus B, \end{aligned} \quad (6)$$

where g denotes the Gaussian filtering with the standard deviation of σ. $\ominus$ and $\oplus$ represent morphological erosion and dilation by a structuring element B, respectively. Together, called opening, these operations remove small sporadic noises and yield clean maps. The structuring element, B, is a simple binary map usually in the shape of an ellipse or disk. Instead of using simple gradients as in Eq. 5, some other gradient-based interpretability methods can be employed to further illuminate the role of each pixel on loss value. We discuss different methods more in Sec. 3.3.4. Our proposed framework is illustrated in Fig. 2. Note that we need only two forward passes for detection and one backward pass through C for localization at the test time.

3.2. Settings

VGG [43] features have shown remarkable performance in classification and transfer learning [46, 48]. This highlights the practicality of its filters in different domains. By transferring the knowledge of an ImageNet VGG-16 to a simple cloner, we exploit the discrepancy of features between C and S to find anomalies. In our VGG-16 source network, we choose the four final layers of each convolutional block, i.e., max-pooling layers, to be the critical points (CP_is). Selecting critical points is explored more in Sec. 3.3.1. For the cloner network, we use the architecture described in Fig. 2, which is smaller than the source for all experiments and datasets. As a result, it can benefit from the advantages of compression discussed in Sec. 3. The role of cloner architecture is discussed more in Sec. 3.3.2. Note that, similar to [33], we avoid using bias terms in our cloner's network. As proven by [33], networks with bias in any layer can easily learn constant functions, independent of the input. In our work, though it can be negligible on datasets with diverse normal data, it can be detrimental when normal images are roughly the same. To be more specific, for some layers l and $l+1$ that are between any i-th and $(i-1)$-th CP, the cloner can generate a specific constant activation vector, $a_C^{CP_i}$, regardless of the input, only by setting the l-th layer weights to zero and adjusting the $l+1$-th layer bias. As the normal training images are much alike, the intermediate source activations are also highly similar for them. Therefore, those constant $a_c^{CP_i}$s can be arbitrarily close to the source correlated intermediate activations for any training input, which is the goal of the training phase, while harming the test procedure since they are constant outputs indeed. To avoid this, we use a bias-less network for C. In all experiments, we use Adam optimizer [21] with learning rate $= 0.001$ and batch size $= 64$ for optimization.

3.3. Ablation Studies

3.3.1 Intermediate Knowledge

In this experiment, we examine the effect of involving the last, the last two, and the last four max-pooling layers as CP_is on MVTecAD and MNIST. We report the average AUROC of all classes in Fig. 3.3.1. Clearly, a consistent growth trend exists that shows the effectiveness of considering more layers. Notice that some MVTecAD classes (e.g. "screw") have near-random AUROC in "just the last layer" setting. This suggests that using just the last layer makes the problem ill-posed and hard to optimize.

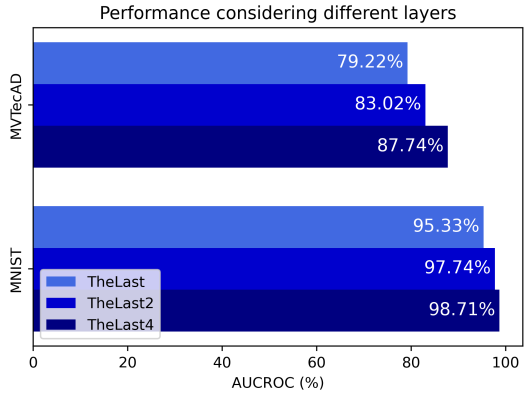

Figure 3: The performance of our proposed method using various layers for distillation. More intermediate layers lead to a performance boost on anomaly detection.

3.3.2 Distillation Effect (Compact C)

As motivated initially in the knowledge distillation field, smaller C plays an essential role in our approach by eliminating non-distinguishing filters causing various distractions. It is especially more important when performing on normal data, where the scope is dramatically limited. Here, we probe the effect of the cloner architecture. As in Fig. 4, anomaly detection, on MVTecAD, using a compact C outperforms a C with equal size to S. This is especially noticeable in the classes where anomalies are partial (like in "toothbrush" or "screw"). Overall, the smaller network performs better with a margin of $\sim 3\%$.

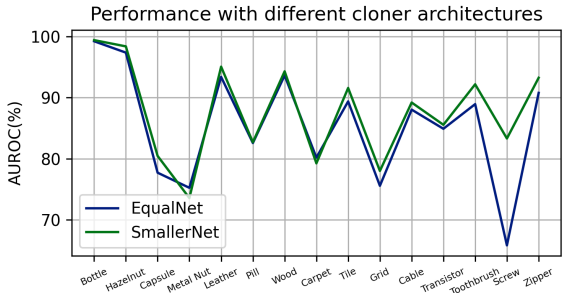

Figure 4: The performance of our proposed method using different equal/smaller cloner architectures compared to the source. The smaller network performs better in general.

3.3.3 $\mathcal{L}_{dir}$ and $\mathcal{L}_{val}$

In this part, we discuss each loss component effect to show the insufficiency of solely considering the Euclidean distance or directional loss in practice. The high impact of using $\mathcal{L}_{total}$ can be seen in Fig. 5. We report the mean AUROC over all classes in the datasets. Refer to the supplementary for a class-detailed report. Using only $\mathcal{L}_{dir}$ shows top results in cases where anomalies are essentially different from normal cases and are more diverse, like in CIFAR-10. In contrast, in cases with subtle anomalies, as in MVTecAD, MSE loss performs better. While $\mathcal{L}_{dir}$ and $\mathcal{L}_{val}$ fail noticeably in either of the cases, our proposed $\mathcal{L}_{total}$, which is a combination of the two losses, achieves the highest performance considering both classification-based and defect detection settings at the same time. These results highlight the positive impact of considering a direction-wise notion of knowledge in addition to an MSE approach.

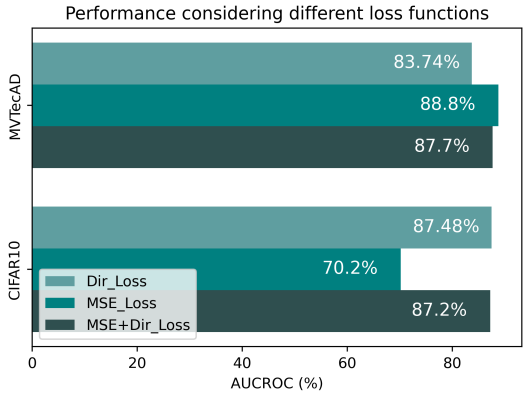

Figure 5: The performance of our proposed method using different loss functions. $\mathcal{L}_{total}$ performs well on both cases while individual directional or Euclidean losses fail in one.

3.3.4 Localization using Interpretability Methods

Here, in addition to simple Gradients explained in Eq. 6, we use other interpretability methods for anomaly localization based on our framework. In Table 1, we report the results on MVTecAD with and without applying the Gaussian filter. As expected, SmoothGrad highlights the anomalous parts better as it discards Gradients' wrongly highlighted pixels by calculating an average over the gradients of noisy inputs. GBP, however, performs the worst since it tends more to reconstruct the image instead of staying faithful to the function [2, 30]. Anyway, after applying the noise-removing filters, the methods perform almost the same. Hence, we use simple Gradients in the rest of our experiments instead of SmoothGrad, requiring severe additional computations.

Table 1: Pixel-wise (AUROC) of anomaly localization on MVTecAD using different interpretability methods with and without Gaussian filtering.

Method	Gradients	SmoothGrad	GBP
Without Gaussian Filter	86.16%	86.97%	84.38%
With Gaussian Filter	90.51%	90.54%	90.08%

4. Experiments

In this section, extensive experiments have been done to demonstrate the effectiveness of our method. As explored in [39], some methods' performance are harmed if trained for more than their hard-coded number of epochs. We report our results on average of the 10 last epochs for 10 different seeds plus the variances to show our stability. We report our method's running time in the supplementary. We stress that S is pre-trained on ImageNet and has not seen any data from the test datasets. Hence, the comparison is fair.

4.1. Experimental Setup

Datasets: We test our method on 7 datasets as follows: **MNIST [24]:** 60k training and 10k test 28×28 gray-scale handwritten digit images. **Fashion-MNIST [49]:** similar to MNIST (with 10k more training images) made up of 10 fashion product categories. **CIFAR-10 [23]** 50k training and 10k test 32×32 color images in 10 equally-sized natural entity classes. **MVTecAD [7]:** an industrial dataset with over 5k high-resolution images in 15 categories of objects and textures. Each category has both normal images and anomalous images having various kinds of defects (only for testing). All images have been down-scaled to the size 128×128. **Retinal OCT Images (optical coherence tomography) [17]:** consisting of 84,495 X-Ray images and 4 categories. **HeadCT [22]:** a medical dataset containing 100 128×128 normal head CT images and 100 with hemorrhage. Each image comes from a different person. **BrainMRI for brain tumor detection [13]:** consisting of 98 256×256 normal MRI images and 155 with tumors.

Evaluation Protocol: Medical datasets: 10 random normal images + all anomalous ones for test, the rest normal ones for training. **MVTecAD & Retinal-OCT**: datasets train and test sets are used. **Others:** one class as normal and others as anomaly, at testing: the whole test set is used.

4.2. Results

4.2.1 MNIST & Fashion-MNIST & CIFAR10

First, we evaluate our method on the conventional AD task on MNIST, Fashion-MNIST, and CIFAR-10 as described in Sec. 4.1. This targets detecting anomalies disparate from

Code to reproduce the results is provided at https://github.com/rohban-lab/Knowledge_Distillation_AD

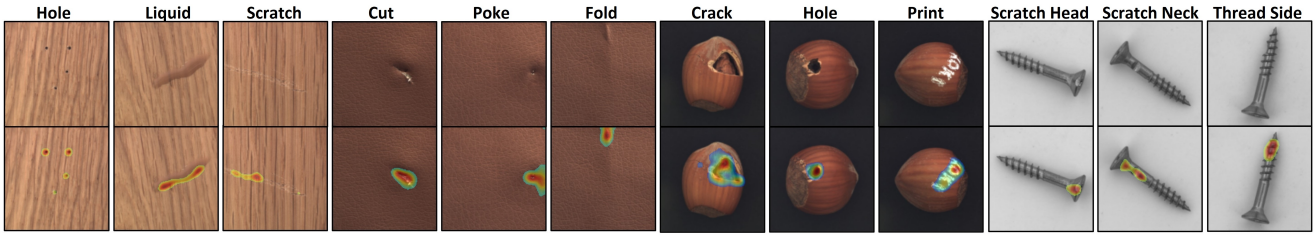

Figure 6: Anomaly localization maps on different types of anomalies in MVTecAD dataset sample classes. Pixels with low score are omitted from the heatmap. This indicates our method precise maps, no matter the defections variety.

Table 2: AUROC in % for anomaly **detection** on **several** datasets. As shown, our model shows SOTA results on MNIST [24] and Fashion-MNIST [49]. On CIFAR-10 [23] dataset our result is 13% higher than SOTA.

Dataset	Method	0	1	2	3	4	5	6	7	8	9	Mean
MNIST[24]	ARAE[38]	99.8	99.9	96.0	97.2	97.0	97.4	99.5	96.9	92.4	98.5	97.5
	OCSVM[14]	99.5	99.9	92.6	93.6	96.7	95.5	98.7	96.6	90.3	96.2	96.0
	AnoGAN[41]	96.6	99.2	85.0	88.7	89.4	88.3	94.7	93.5	84.9	92.4	91.3
	DSVDD[33]	98.0	99.7	91.7	91.9	94.9	88.5	98.3	94.6	93.9	96.5	94.8
	CapsNet$_{PP}$ [25]	99.8	99.0	98.4	97.6	93.5	97.0	94.2	98.7	99.3	99.0	97.7
	OCGAN[31]	99.8	99.9	94.2	96.3	97.5	98.0	99.1	98.1	93.9	98.1	97.5
	LSA[1]	99.3	99.9	95.9	96.6	95.6	96.4	99.4	98.0	95.3	98.1	97.5
	CAVGA-D_u[47]	99.4	99.7	98.9	98.3	97.7	96.8	98.8	98.6	98.8	99.1	98.6
	U-Std[9]	99.9	99.9	99	99.3	99.2	99.3	99.7	99.5	98.6	99.1	**99.35**
	OURS	99.82 ± 0.023	99.82 ± 0.017	97.79 ± 0.272	98.75 ± 0.098	98.43 ± 0.096	98.16 ± 0.182	99.43 ± 0.038	98.38 ± 0.178	98.41 ± 0.157	98.1 ± 0.152	98.71
Fashion-MNIST[49]	ARAE[38]	93.7	99.1	91.1	94.4	92.3	91.4	83.6	98.9	93.9	97.9	93.6
	OCSVM[14]	91.9	99.0	89.4	94.2	90.7	91.8	83.4	98.8	90.3	98.2	92.8
	DAGMM[59]	30.3	31.1	47.5	48.1	49.9	41.3	42.0	37.4	51.8	37.8	41.7
	DSEBM[51]	89.1	56.0	86.1	90.3	88.4	85.9	78.2	98.1	86.5	96.7	85.5
	DSVDD[33]	98.2	90.3	90.7	94.2	89.4	91.8	83.4	98.8	91.9	99.0	92.8
	LSA[1]	91.6	98.3	87.8	92.3	89.7	90.7	84.1	97.7	91.0	98.4	92.2
	OURS	92.5 ± 0.298	99.21 ± 0.064	92.48 ± 0.255	93.8 ± 0.095	92.95 ± 0.159	98.21 ± 0.157	84.87 ± 0.126	99.02 ± 0.331	94.33 ± 0.164	97.51 ± 0.055	**94.49**
CIFAR-10[23]	ARAE[38]	72.2	43.1	69.0	55.0	75.2	54.7	70.1	51.0	72.2	40.0	60.23
	OCSVM[14]	63.0	44.0	64.9	48.7	73.5	50.0	72.5	53.3	64.9	50.8	58.56
	AnoGAN[41]	67.1	54.7	52.9	54.5	65.1	60.3	58.5	62.5	75.8	66.5	61.79
	DSVDD[33]	61.7	65.9	50.8	59.1	60.9	65.7	67.7	67.3	75.9	73.1	64.81
	CapsNet$_{PP}$[25]	62.2	45.5	67.1	67.5	68.3	63.5	72.7	67.3	71.0	46.6	61.2
	OCGAN[31]	75.7	53.1	64.0	62.0	72.3	62.0	72.3	57.5	82.0	55.4	65.66
	LSA[1]	73.5	58.0	69.0	54.2	76.1	54.6	75.1	53.5	71.7	54.8	64.1
	DROCC[19]	81.66	76.74	66.66	67.13	73.62	74.43	74.43	71.39	80.02	76.21	74.23
	CAVGA-D_u[47]	65.3	78.4	76.1	74.7	77.5	55.2	81.3	74.5	80.1	74.1	73.7
	GT[18]	76.2	84.8	77.1	73.2	82.8	84.8	82	88.7	89.5	83.4	82.3
	U-Std[9]	78.9	84.9	73.4	74.8	85.1	79.3	89.2	83	86.2	84.8	81.96
	OURS	90.53 ± 0.158	90.35 ± 0.797	79.66 ± 0.415	77.02 ± 0.51	86.71 ± 0.346	91.4 ± 0.279	88.98 ± 0.2	86.78 ± 0.595	91.45 ± 0.148	88.91 ± 0.349	**87.18**

the normal samples in essence and not only slightly. As CIFAR-10 images are natural images, they have been resized and normalized according to the ImageNet properties. No normalization and resizing are done for other datasets. For evaluation, like previous works, we use the area under the receiver operating characteristic curve (AUROC). We compare our method with an exhaustive set of state-of-the-art approaches, including generative, self-supervised, and autoencoder-based methods, in Table 2. We outperform all other methods on F-MNIST and CIFAR-10 while staying comparatively well on MNIST, though avoiding complicated training procedures. Note that some methods, like U-Std, apply dataset-dependent fine-tunings. We, however, avoid such fine-tunings.

4.2.2 MVTecAD

Detection: In this part, we report the results of our method performance on AD using MVTecAD. As shown in Table 3, our method outperforms all others with a large margin of $\sim 10\%$. This is remarkable since other methods fail to perform well in both one-class settings and defect detection simultaneously. In contrast, we achieve SOTA in both cases.

Localization: We not only accomplish SOTA in AD but we also outperform previous SOTA methods in anomaly localization. As stated in 3.3.4, we use simple gradients to obtain maps. We use the Gaussian filter with $\sigma = 4$ and a 3×3 ellipse structural element kernel. We compare our method against others in Table 4. We use AUROC, based on each pixel anomaly score, to measure how well anomalies are localized. Vividly, we outperform previous methods. Fig. 6 shows our localization maps on different defect types in MVTecAD.

4.2.3 Medical Datasets

To further evaluate our method in various domains, we use three medical datasets and compare our method with others.

Table 3: AUROC in % for anomaly **detection** on MVTecAD [7]. We surpass the SOTA by $\sim 10\%$

Method	Bottle	Hazelnut	Capsule	Metal Nut	Leather	Pill	Wood	Carpet	Tile	Grid	Cable	Transistor	Toothbrush	Screw	Zipper	Mean
AVID [37]	88	86	85	63	58	86	83	70	66	59	64	58	73	66	84	73
AE_{SSIM} [11]	88	54	61	54	46	60	83	67	52	69	61	52	74	51	80	63
AE_{L2} [11]	80	88	62	73	44	62	74	50	77	78	56	71	98	69	80	71
AnoGAN[41]	69	50	58	50	52	62	68	49	51	51	53	67	57	35.0	59	55
LSA[1]	86	80	71	67	70	85	75	74	70	54	61	50	89	75	88	73
CAVGA-D_0[47]	89	84	83	67	71	88	85	73	70	75	63	73	91	77	87	78
DSVDD[33]	86	71	69	75	73	77	87	54	81	59	71	65	70	64	74	72
VAE-grad[15]	86	74	86	78	71	80	89	67	81	83	56	70	89	71	67	77
GT*[18]	74.29	33.32	67.79	82.37	82.51	65.16	48.24	45.9	53.86	61.91	84.7	79.79	94	44.58	87.44	67.06
OURS	99.39 ± 0.032	98.37 ± 0.285	80.46 ± 0.19	73.58 ± 0.376	95.05 ± 0.208	82.7 ± 0.646	94.29 ± 0.226	79.25 ± 0.8	91.57 ± 0.66	78.01 ± 0.621	89.19 ± 0.378	85.55 ± 0.212	92.17 ± 0.323	83.31 ± 0.643	93.24 ± 0.247	**87.74**

Table 4: AUROC in % for anomaly **localization** on MVTecAD [7].

Method	Bottle	Hazelnut	Capsule	Metal Nut	Leather	Pill	Wood	Carpet	Tile	Grid	Cable	Transistor	Toothbrush	Screw	Zipper	Mean
AE_{SSIM}[11]	93	97	94	89	78	91	73	87	59	94	82	90	92	96	88	87
AE_{L2}[11]	86	95	88	86	75	85	73	59	51	90	86	86	93	96	77	82
AnoGAN[41]	86	87	84	76	64	87	62	54	50	58	78	80	90	80	78	74
CNN-Dict[28]	78	72	84	82	87	68	91	72	93	59	79	66	77	87	76	78
VAE-grad[15]	92.2	97.6	91.7	90.7	92.5	93	83.8	73.5	65.4	96.1	91	91.9	98.5	94.5	86.9	89.3
OURS	96.32	94.62	95.86	86.38	98.05	89.63	84.8	95.64	82.77	91.78	82.4	76.45	96.12	95.96	93.9	**90.71**

Table 5: AUROC in % on Retinal-OCT [56]. We outperform all other SOTA methods.

	DSVDD[33]	Auto-Encoder[55]	AnoGan[41]	VAE-GAN[5]	Pix2Pix[20]	GANomaly[3]	Cycle-GAN[57]	P-Net[56]	GT[18]	OURS
RESC (OCT)[17]	74.40	82.07	84.81	90.64	79.34	91.96	87.39	92.88	60.13	**97.01** $\pm$ **0.426**

First, we use the Retinal-OCT dataset, a recent dataset for detecting abnormalities in retinal optical coherence tomography (OCT) images. According to Table 5, our method outplays all the SOTA methods by a considerable margin. This shows that the knowledge of the pre-trained network, S, has been precious to the cloner, C, even in an entirely different domain of medical retinal OCT inputs. Furthermore, the unawareness of C about the outside of the normal data manifold intensifies the discrepancy between them. This expresses the generality of our method to even future unseen datasets, something missed in many methods.

Moreover, we validate our performance on brain tumor detection using brain MRIs. In this dataset, images with tumors are assumed as anomalies while healthy ones are considered normal. In Table 6, our method achieves SOTA results alongside LSA. While slightly ($\sim 0.5\%$) less than LSA, our method shows a significantly lower variance, magnifying its stability compared to the others. It is also noteworthy that LSA fails substantially on other tasks such as CIFAR-10 and MVTecAD anomaly detection with AUROCs $\sim 23\%$ and $\sim 25\%$ below our method's, respectively. Lastly, using HeadCT (hemorrhage) dataset, we discuss an important aspect of our model. Performing on head computed tomography (CT) images for AD, we outperform OCGAN and GT by a considerable margin and perform $\sim 3\%$ below LSA. Since the training data is dramatically limited, our method may face difficulties transferring the S's knowledge to C. However, this can be addressed by using simple data augmentations. We use 20-degree rotation in addition to scaling in the range of $[0.9, 1.05]$ to augment the images. These augmentations are generic non-tuned ones aiming solely to increase the amount of data with no dependency on the dataset. Table 6 shows that by utilizing augmentations, the proposed method achieves similar results to LSA while outperforming it on other tasks significantly.

Table 6: AUROC in % medical datasets. The top two methods are in bold.

	BrainMRI	HeadCT
LSA*[1]	**95.61** $\pm$ **1.433**	**81.67** $\pm$ **0.358**
OCGAN*[31]	91.74 ± 3.050	51.22 ± 3.626
GT*[18]	80.82 ± 1.996	49.85 ± 3.873
OURS	**95.01** $\pm$ **0.229**	78.04 ± 0.225
OURS+AUG	-	**80.42** $\pm$ **0.006**

5. Conclusion

We show that "distilling" the intermediate knowledge of an ImageNet pre-trained expert network on anomaly-free data into a more compact cloner network, and then using their different behavior with different samples, sets a new direction for finding specific criteria to detect and localize anomalies. Without using intensive region-based training and testing, we leverage interpretability methods in our novel framework for obtaining localization maps. We achieve superior results in various tasks and on many datasets, even with domains far from the ImageNet domain.

References

[1] Davide Abati, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Latent space autoregression for novelty detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 481–490, 2019.

[2] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, pages 9505–9515, 2018.

[3] Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian Conference on Computer Vision*, pages 622–637. Springer, 2018.

[4] Jerone Andrews, Thomas Tanay, Edward J Morton, and Lewis D Griffin. Transfer representation-learning for anomaly detection. JMLR, 2016.

[5] Christoph Baur, Benedikt Wiestler, Shadi Albarqouni, and Nassir Navab. Deep autoencoding models for unsupervised anomaly segmentation in brain mr images. pages 161–169, 04 2018.

[6] Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. *arXiv preprint arXiv:2005.02359*, 2020.

[7] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9592–9600, 2019.

[8] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4183–4192, 2020.

[9] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. 03 2020.

[10] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. *arXiv preprint arXiv:1807.02011*, 2018.

[11] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, 2019.

[12] Philippe Burlina, Neil Joshi, I Wang, et al. Where's wally now? deep generative and discriminative embeddings for novelty detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11507–11516, 2019.

[13] Navoneel Chakrabarty. Brain mri images for brain tumor detection. https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection, 2019.

[14] Yunqiang Chen, Xiang Sean Zhou, and Thomas S Huang. One-class svm for learning in image retrieval. In *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, volume 1, pages 34–37. IEEE, 2001.

[15] David Dehaene, Oriel Frigo, Sébastien Combrexelle, and Pierre Eline. Iterative energy-based projection on a normal data manifold for anomaly localization. 02 2020.

[16] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[17] Peyman Gholami, Priyanka Roy, Mohana Kuppuswamy Parthasarathy, and Vasudevan Lakshminarayanan. Octid: Optical coherence tomography image database. *Computers & Electrical Engineering*, 81:106532, 2020.

[18] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In *Advances in Neural Information Processing Systems*, pages 9758–9769, 2018.

[19] Sachin Goyal, Aditi Raghunathan, Moksh Jain, Harsha Simhadri, and Prateek Jain. Drocc: Deep robust one-class classification. 02 2020.

[20] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei Efros. Image-to-image translation with conditional adversarial networks. pages 5967–5976, 07 2017.

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[22] Felipe Kitamura. Head ct - hemorrhage. https://www.kaggle.com/felipekitamura/head-ct-hemorrhage, 2018.

[23] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). 2009.

[24] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. 2010.

[25] Xiaoyan Li, Iluju Kiringa, Tet Yeap, Xiaodan Zhu, and Yifeng Li. Exploring deep anomaly detection methods based on capsule net. In *ICML 2019 Workshop on Uncertainty and Robustness in Deep Learning, At Long Beach*, 2019.

[26] Zhe Li, Chong Wang, Mei Han, Yuan Xue, Wei Wei, Li-Jia Li, and Li Fei-Fei. Thoracic disease identification and localization with limited supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8290–8299, 2018.

[27] Shuang Mei, Yudan Wang, and Guojun Wen. Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model. *Sensors*, 18(4):1064, 2018.

[28] Paolo Napoletano, Flavio Piccoli, and Raimondo Schettini. Anomaly detection in nanofibrous materials by cnn-based self-similarity. *Sensors (Basel, Switzerland)*, 18, 01 2018.

[29] 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.

[30] Weili Nie, Yang Zhang, and Ankit Patel. A theoretical explanation for perplexing behaviors of backpropagation-based visualizations. *arXiv preprint arXiv:1805.07039*, 2018.

[31] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. Ocgan: One-class novelty detection using gans with constrained latent representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2898–2906, 2019.

[32] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[33] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pages 4393–4402, 2018.

[34] Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. *arXiv preprint arXiv:1906.02694*, 2019.

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

[36] Mohammad Sabokrou, Mohammad Khalooei, Mahmood Fathy, and Ehsan Adeli. Adversarially learned one-class classifier for novelty detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3379–3388, 2018.

[37] Mohammad Sabokrou, Masoud Pourreza, Mohsen Fayyaz, Rahim Entezari, Mahmood Fathy, Jürgen Gall, and Ehsan Adeli. Avid: Adversarial visual irregularity detection. In *Asian Conference on Computer Vision*, pages 488–505. Springer, 2018.

[38] Mohammadreza Salehi, Atrin Arya, Barbod Pajoum, Mohammad Otoofi, Amirreza Shaeiri, Mohammad Hossein Rohban, and Hamid R Rabiee. Arae: Adversarially robust training of autoencoders improves novelty detection. *arXiv preprint arXiv:2003.05669*, 2020.

[39] Mohammadreza Salehi, Ainaz Eftekhar, Niousha Sadjadi, Mohammad Hossein Rohban, and Hamid R Rabiee. Puzzle-ae: Novelty detection in images through solving puzzles. *arXiv preprint arXiv:2008.12959*, 2020.

[40] 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.

[41] 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.

[42] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[43] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[44] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[45] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[46] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *International conference on artificial neural networks*, pages 270–279. Springer, 2018.

[47] Shashanka Venkataramanan, Kuan-Chuan Peng, Rajat Vikram Singh, and Abhijit Mahalanobis. Attention guided anomaly localization in images. 11 2019.

[48] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):9, 2016.

[49] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[50] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4133–4141, 2017.

[51] Shuangfei Zhai, Yu Cheng, Weining Lu, and Zhongfei Zhang. Deep structured energy based models for anomaly detection. *arXiv preprint arXiv:1605.07717*, 2016.

[52] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[53] Teng Zhang, Arnold Wiliem, and Brian C Lovell. Region-based anomaly localisation in crowded scenes via trajectory analysis and path prediction. In *2013 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–7. IEEE, 2013.

[54] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for neural networks for image processing. *arXiv preprint arXiv:1511.08861*, 2015.

[55] Chong Zhou and Randy Paffenroth. Anomaly detection with robust deep autoencoders. pages 665–674, 08 2017.

[56] Kang Zhou, Yuting Xiao, Jianlong Yang, Jun Cheng, Wen Liu, Weixin Luo, Zaiwang Gu, Jiang Liu, and Shenghua Gao. Encoding structure-texture relation with p-net for anomaly detection in retinal images. 08 2020.

[57] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. pages 2242–2251, 10 2017.

[58] David Zimmerer, Jens Petersen, Simon AA Kohl, and Klaus H Maier-Hein. A case for the score: Identifying image anomalies using variational autoencoder gradients. *arXiv preprint arXiv:1912.00003*, 2019.

[59] 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. 2018.