MAEDAY: MAE for few and zero shot AnomalY-Detection

Anonymous CVPR submission

Paper ID 5117

Abstract

The goal of Anomaly-Detection (AD) is to identify outliers, or outlying regions, from some unknown distribution given only a set of positive (good) examples. Few-Shot AD (FSAD) aims to solve the same task with a minimal amount of normal examples. Recent embedding-based methods, that compare the embedding vectors of queries to a set of reference embeddings, have demonstrated impressive results for FSAD, where as little as one good example is provided. A different approach, image-reconstruction-based, has been historically used for AD. The idea is to train a model to recover normal images from corrupted observations, assuming that the model will fail to recover regions when encountered with an out-of-distribution image. However, image-reconstruction-based methods were not yet used in the low-shot regime as they need to be trained on a diverse set of normal images in order to properly perform. We suggest using Masked Auto-Encoder (MAE), a self-supervised transformer model trained for recovering missing image regions based on their surroundings for FSAD. We show that MAE performs well by pre-training on an arbitrary set of natural images (ImageNet) and only fine-tuning on a small set of normal images. We name this method MAEDAY. We further find that MAEDAY provides an orthogonal signal to the embedding-based methods and the ensemble of the two approaches achieves very strong SOTA results. We also present a novel task of Zero-Shot AD (ZSAD) where no normal samples are available at training time. We show that MAEDAY performs surprisingly well at this task. Finally, we provide a new dataset for detecting foreign objects on the ground and demonstrate superior results for this task as well.

1. Introduction

"All happy families are alike, but every unhappy family is unhappy in its own way" [24]. The challenge of Anomaly-Detection (AD) stems from the fact that good cases are similar and easy to model, while anomalies rarely happen, and when they do, they can take an unpredictable form. For this reason, classic supervised training is sometimes not feasible for AD. In AD only good images are provided during training, the goal is to model the distribution of the good images and thus detect outliers at inference time when they occur. There are two main approaches to AD, embedding-similarity based [3, 4, 19] and image-reconstruction based [5,9,12,21,27,29]. Embedding-similarity based methods utilize a pre-trained model to extract and aggregate representations of the normal images or patches. The representation of a query image is compared with those of the normal images to determine if it is anomalous. Image-reconstruction based methods use only normal images to train a model to reconstruct the images from from a corrupted observation, e.g. noisy image or partially masked-out.

Recently, there has been a great interest in Few-shot AD (FSAD) [11, 19, 20, 23]. The promise of FSAD is that a single model can be used for different objects and adapted based on only few good samples. Embedding-based methods have demonstrated high performance for FSAD since they mostly rely on pre-trained models and do not need a lot of training data. On the other hand, previous image-reconstruction-based methods trained the reconstruction model from scratch and therefore required larger training sets.

We suggest, for the first time, image-reconstruction based method that can be used for FSAD. We do that by pre-training the model for general natural-image reconstruction (pre-training on ImageNet). Our suggested method, MAEDAY, addresses FSAD by using Masked AutoEncoder (MAE) [8], a model trained for general image completion based on partial observations. MAE was introduced for a different purpose, trained on a self-supervised task (image inpainting) with the end goal of learning image representation. We re-purpose MAE for FSAD, unlike MAE where the decoder is discarded at inference time, we use both the encoder and the encoder to get a recovered image and not just an intermediate representations. We use the available few good images to further fine-tune the MAE. The idea is that normal regions will be easier to recover based on patterns observed in the few good examples and based on recurring patterns in the query image itself. As in

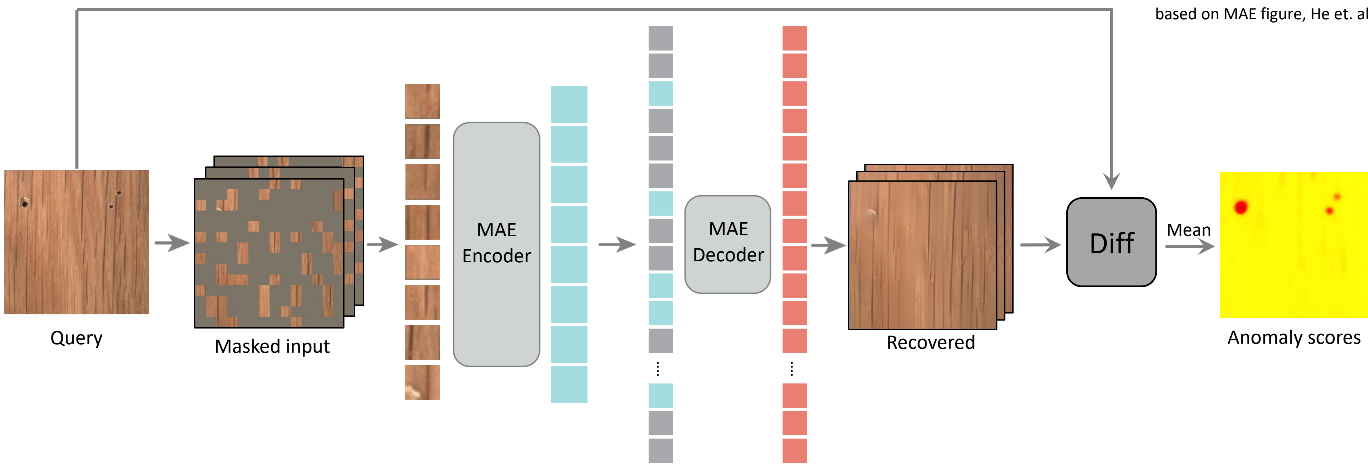


Figure 1. MAEDAY: We repurposed MAE for Zero and Few-Shot Anomaly-Detection. In the zero-shot setup, with no special training and no good images as a reference, ImageNet pre-trained MAE is used to reconstruct a mostly masked-out query image. Anomalous regions are detected in areas where the reconstruction fails, as these regions cannot be accurately inferred from neighboring regions. The anomaly scores are averaged across multiple reconstructions with different random masks. In the few-shot case, the pre-trained model is further finetuned on the reconstruction of the available normal images.

the many-shot case, image-reconstruction is underperforming compared to embedding-based. But we observed that an ensemble of embedding-based and MAEDAY performs extremely well and sets a new state-of-the-art result.

Following FSAD, we also suggest a new task, Zero-shot AD (ZSAD). A class-invariant model that takes as input a single query image (without any good reference) and detects anomalies or irregularities. Since the model should detect anomalies with no access to a reference image, it is relevant for textures, where patterns repeat and the query image acts as a self-reference. Such a model can be particularly useful in industrial settings, e.g. manufacturing of textured materials. We show that MAEDAY, without any training images, achieves high results for ZSAD and particularly compares favorably to the FSAD SOTA for the textures datasets in MVTec.

We also explore a new task of Zero-Shot Foreign Object Detection (ZSFOD). Most Foreign Object Detection works are using annotated images with bounding-boxes or segmentation masks to train an object-detector [13, 15, 17]. A common use-case is detecting foreign objects or debris on the pavement in airports' runways [16]. We focus on the zero-shot case, having a single model that can generalize to new use-cases, with no prior reference of either a free-of-objects surface or the objects to be detected. We treated this problem similarly to ZSAD where the objects are an anomaly in the surface texture. We release a new FOD dataset of wooden floor (indoor) and pavement (outdoor) with or without foreign object. We show that MAEDAY, without any training images, outperforms the SOTA one-shot results on this dataset.

To summarise, our contributions are (1) Suggesting MAEDAY, MAE-based model pre-trained for image reconstruction on arbitrary set of images and used for Few-Shot Anomaly-Detection (FSAD); (2) Suggesting the new task of Zero-Shot AD (ZSAD) and demonstrating strong results, particularly for textures (3) Suggesting the new task of Zero-Shot Foreign Object Detection (ZSFOD) and showing strong results; (4) Releasing a new FOD dataset.

2. Related Work

AD methods divide into two categories: embedding-similarity-based and image-reconstruction-based.

Embedding-similarity-based methods compare image or patch embedding with a distribution of normal image or patch embeddings (modeled by the training set), e.g. [3, 4, 11, 19]. Some methods performs registration of the images, i.e. spatial mapping of the image to some canonical form [2, 11]. Other approaches learn the negative distribution, too. That requires some assumptions on the anomaly distribution and achieved by artificially producing anomalies [14, 32]. The similarity-based methods are successful in cases where normal data is abundant, and even demonstrated success in the low data regime (FSDA) [19, 20, 23].

Image-reconstruction-based methods usually train a generative model on a set of normal images, e.g. an AutoEncoder [9, 12, 21] or GAN [6, 22, 26, 30]. The underline assumption is that only good images can be generated by the trained model. Another kind of generative-model is Normalizing Flows [18], by using an invertible mapping from a latent space with controlled distribution to images we also obtain the inverse mapping that allows verifying the likelihood of a query image [7, 28, 31]. Other methods apply some form of image degradation and again train a model to reconstruct the images, assuming only good images will be well-reconstructed [5, 25, 27, 29]. The closest to our approach is RIAD [29], it masks parts of the image and performs image inpainting. However, RIAD and other image-reconstruction methods rely on training a model from scratch on the normal images and are not intended for the low-data or no-data regime.

3. Method

We begin by describing our approach (MAEDAY) for ZSAD which is based on image reconstruction from partial observations. MAE [8] is trained on the self-supervised task of predicting an image from a partial observation. This makes MAE a great tool for our purpose. We use an ImageNet pretrained MAE as our backbone.

As commonly done in transformer-based architectures, the input image I is split into non-overlapping patches, each patch is flattened into a single token. The tokens go through a linear projection with the addition of a positional encoding and are then processed by a sequence of transformer blocks. For MAE most of the input tokens are masked out and discarded, therefore the encoder operates on a small number of tokens. The decoder receives the output tokens of the encoder and in addition 'empty' tokens with just the positional encoding replacing the masked-out tokens. Through a sequence of transformer blocks, the decoder 'fills' these empty tokens based on information from the encoder output tokens. The output of the decoder is the recovered image.

Usually, at inference time only the MAE encoder is used (for features extraction), while the decoder is discarded. In our case, we use both the encoder and decoder. Given a query image, a random small subset of its patches (25%) are fed to the MAE. The recovered image is then compared against the query image and mismatched pixels indicate an anomalous region. We repeat this process multiple times for each image, each time a different subset of the tokens is retained. With enough repetitions (we used $N = 32$) each token is likely to be masked out at least once, such that we can measure how well it is reconstructed. We found in our experiments that the reconstruction for retained tokens (not masked-out) is also somewhat indicative of them being normal vs. anomalous. Our intuition for that is that since the transformer mixes the information from all tokens, even when a token is visible it will be better reconstructed when it is in agreement with its surrounding tokens. Given this observation, we can simply run a query image N times with different random masks and compare the N reconstructed images (full images) against the query image. The method is illustrated in Figure 1.

Formally, given a query image $I \in \mathbb{R}^{H\times W\times 3}$ and a set of N random masks $\{M_1, ..., M_N\}$, we use MAE to get N reconstructed images $\{R_1, ..., R_N\}$, where $R_i = MAE(I \cdot M_i)$. Image resolution and patch size are the same as those used for pretraining MAE (224 and 16). hen we use R_i to compute N squared error maps. The squared error maps are channel-wise filtered with a Gaussian kernel g (kernel size 7, $\sigma = 1.4$) to remove noise and summed over the 3 color channels,

$$E_i = \sum_{c\in\{R,G,B\}} (I^c - R_i^c)^2 * g. \tag{1}$$

The N error maps are averaged to get a single error map, $E = \frac{1}{N}\sum_{i=1}^{N} E_i$. E is the pixel-level anomaly score. Finally, the image anomaly score is set by the max error $S = max(E)$.

For FSAD we first finetune the MAE model with the available normal images. Unlike MAE, where the loss is applied only on the recovered masked out patches, we apply the loss to all patches. We do that because we use all predicted patches (both masked and unmasked) for detecting anomalies. We use LoRA [10], a method originally introduced for finetuning large language models (transformers) without overfitting a small dataset. In LoRA additional low-rank weight matrix is introduced for each weight matrix in the original pre-trained model. The low rank is enforced by having a low-rank decomposition. During fine-tuning, only the low-rank weights are updated and the output of each multiplication is the sum of performing the multiplication with the original weights and the new low-rank weights. After finetuning is finished, the weights are updated to be the sum of the original weights and the new ones (to avoid additional compute and memory consumption at inference time).

We set the rank of the additional LoRA weights to 32 for all tensors in the model. The model is trained for 50 iterations using an SGD optimizer with a learning rate of $1e-2$ (LoRA requires a relatively high learning rate), a momentum of 0.9 and weight decay of 0.05. We train with random crop and random rotation augmentations. The batch size is set to 32, so the few available shots are used multiple times to fill the batch (but with different random masks each time).

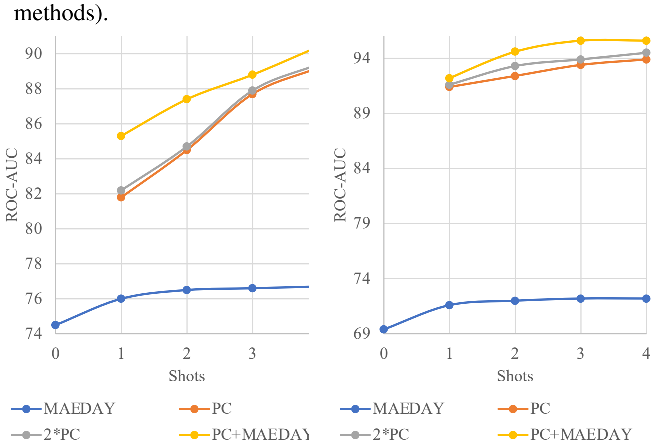
4. Results

We evaluated our method on all of the 15 datasets in MVTec-AD [1], the most popular and the main AD benchmark. It is focused on an industrial inspection use case and consists of 10 unique objects and 5 unique textures. For each object or texture a training set of defect-free images and a test set of both normal and anomalous instances are available. The anomalous images are provided with pixel-level annotation marking the anomaly location.

For the few-shot test, in each run we selected a few random training samples from the relevant dataset training set and tested on the full associated test set. Since the performance can be dependent on the selected samples we averaged all results over 3 different shots selection. When comparing to other methods we made sure the same exact shots

	0-Shot	1-Shot			
	Single-Model	Single-Model		Ensemble	
	MAEDAY	PC	MAEDAY	2*PC	MAEDAY+PC
Objects					
bottle	74.3	**96.1** ± 3.5	74.8 ± 0.1	**98.3** ± 1.8	93.7 ± 1.8
cable	53.0	**82.6** ± 0.8	50.1 ± 5.0	**83.6** ± 2.3	69.0 ± 4.6
capsule	64.0	**63.0** ± 1.8	59.9 ± 9.5	63.7 ± 1.8	**64.9** ± 1.9
hazelnut	97.1	84.9 ± 5.6	**97.0** ± 0.2	85.4 ± 5.1	**94.1** ± 0.2
metal-nut	43.6	**75.4** ± 3.4	53.1 ± 1.5	**77.0** ± 2.8	73.4 ± 1.8
pill	63.4	**77.5** ± 1.4	63.5 ± 0.5	79.1 ± 1.9	**81.7** ± 2.1
screw	69.9	46.0 ± 2.6	**78.1** ± 2.5	45.8 ± 2.6	**61.4** ± 2.2
toothbrush	77.5	**84.4** ± 1.6	81.7 ± 2.9	83.8 ± 1.4	**92.5** ± 1.0
transistor	48.3	**82.1** ± 3.8	56.3 ± 4.1	**80.1** ± 5.0	75.3 ± 2.7
zipper	82.0	**96.6** ± 1.4	79.0 ± 0.2	**96.9** ± 0.4	94.3 ± 1.1
Mean (Objects)	67.3	**78.9**	69.3	79.3	**80.1**
Textures					
carpet	74.6	**99.1** ± 0.1	72.3 ± 1.1	**99.2** ± 0.0	97.9 ± 0.2
grid	97.9	43.4 ± 6.1	**97.1** ± 0.3	43.2 ± 5.5	**83.9** ± 6.5
leather	92.9	**100.** ± 0.0	93.4 ± 0.1	**100.** ± 0.0	99.9 ± 0.0
tile	84.3	**98.5** ± 0.2	87.2 ± 1.5	**98.7** ± 0.2	**98.4** ± 0.2
wood	94.8	**98.5** ± 0.5	96.7 ± 0.5	98.5 ± 0.5	**99.5** ± 0.0
Mean (Textures)	88.9	87.9	**89.3**	87.9	**95.9**
Mean (All)	74.5	**81.9**	76.0	82.2	**85.3**

Table 1. Image-level ROC-AUC results for 0-shot and 1-shot on the MVTec datasets. MAEDAY performs surprisingly well even on objects and textures the model was not trained on (ZSAD). In the 1-shot case, the embedding-based method, PC [19], has higher performance when evaluating a single model. However, MAEDAY adds new kind of information and hence a MAEDAY+PC ensemble outperforms an ensemble of 2 PC models. All 1-shot results are presented with mean±std over 3 runs with different shot selection (same shots for all methods).



(a) Image-level ROC-AUC (b) Pixel-level ROC-AUC

Figure 2. ROC-AUC for 0-4 shot on the MVTec dataset.

are used by all methods. When an ensemble of models is used, the same shots are used for all models, and the models' output images and pixel-level scores are summed. For the zero-shot test, per the task definition, the training set is not used.

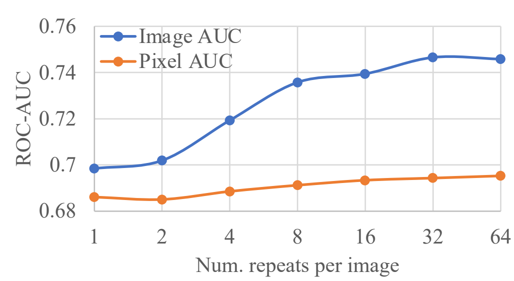


Figure 3. **Number of repetitions per image**. Scores for each image are averaged over multiple reconstructions with different random masks. We observe performance saturation at ~ 32 repetitions.

Table 1 summarizes the results for image-level zero and one-shot anomaly detection performance. Even though for the zero-shot case MAEDAY uses no normal training data, we observe relatively strong results. For the textures datasets it even outperforms the SOTA 1-shot results. In the 1-shot case, we observe 1.5% improvement of MAEDAY

	0-Shot	1-Shot			
	Single-Model	Single-Model		Ensemble	
	MAE0	PC	MAEDAY	2*PC	MAEDAY + PC
Objects					
bottle	50.7	**97.9** ± 0.1	50.8 ± 0.5	**98.1** ± 0.1	95.9 ± 0.3
cable	65.5	**90.3** ± 1.2	73.1 ± 3.1	**91.3** ± 1.0	84.2 ± 0.7
capsule	48.1	**97.1** ± 0.1	48.4 ± 3.6	**97.2** ± 0.1	95.3 ± 1.3
hazelnut	94.1	88.5 ± 1.5	**94.0** ± 0.2	88.8 ± 1.5	**98.3** ± 0.1
metal-nut	39.6	**89.6** ± 0.8	47.0 ± 0.7	**90.1** ± 0.6	68.4 ± 1.2
pill	61.5	**94.7** ± 0.4	62.0 ± 1.1	**95.1** ± 0.3	91.3 ± 1.2
screw	96.9	88.6 ± 0.5	**96.4** ± 0.4	88.8 ± 0.5	**97.4** ± 0.0
toothbrush	72.3	**95.0** ± 0.2	77.6 ± 3.0	**95.2** ± 0.2	92.2 ± 0.5
transistor	59.7	**92.3** ± 1.0	61.9 ± 0.2	**92.3** ± 0.8	86.0 ± 1.9
zipper	76.2	**96.9** ± 0.4	73.9 ± 0.6	**97.1** ± 0.3	96.2 ± 0.4
Mean (Objects)	66.5	93.0	69.9	93.3	90.5
Textures					
carpet	76.2	**98.9** ± 0.0	78.4 ± 1.7	**99.0** ± 0.0	98.2 ± 0.2
grid	95.4	55.7 ± 0.3	**96.7** ± 0.3	55.9 ± 0.3	**96.6** ± 0.2
leather	94.6	**99.1** ± 0.0	96.4 ± 0.5	99.1 ± 0.0	**99.4** ± 0.0
tile	30.9	**94.8** ± 0.5	37.4 ± 2.1	**94.9** ± 0.5	90.1 ± 1.1
wood	78.8	**92.0** ± 0.2	80.0 ± 0.4	92.1 ± 0.2	**92.9** ± 0.4
Mean (Textures)	75.2	88.1	79.7	88.2	95.4
Mean	69.4	91.4	71.6	91.7	**92.2**

Table 2. Pixel-level AUC-ROC on MVTec datasets. See Table 1 for details.

thanks to the finetuning on the normal sample. Yet, this is still 5.9% lower than the SOTA embedding-based method (PatchCore). Next, we test the performance of an ensemble of two models. While the ensemble of two PatchCore models outperforms a single one thanks to the stochastic nature of the method, the gain is limited by the fact they use the same embeddings and the same kind of information. The ensemble of MAEDAY with PatchCore outperform the two PatchCore ensemble by 3.1%.

Table 2 summarizes the results for pixel-level zero and one-shot anomaly detection (segmentation) performance. While the gap between MAEDAY and PatchCore is higher for pixel-level detection, we observe similar trends to image-level performance. For a single model PatchCore outperforms MAEDAY, but an ensemble of MAEDAY and PatchCore is better than an ensemble of two PatchCore models. We attribute the lower pixel-level performance to the fact that, even though MAEDAY is mostly able to detect the anomalies, often the detected anomaly only partially covers the full anomaly region. Examples of segmentation maps produced by MAEDAY are presented in Figure 4. Examples of the recovered images from masked inputs are presented in Figure 5, while the recovered images tend to be blurry they usually provide enough signal for detecting anomalies.

We explored finetuning MAEDAY on more shots in Figure 2. The improvement in performance of MAEDAY saturates at about 4 shots, making it a best fit for the low shot scenario. For more shots, we observed similar results to 1-shot, where an ensemble of MAEDAY and PatchCore sets a new SOTA.

Num. of repetitions per image We tested the effect of averaging the anomaly score from multiple inferences of the same image with different random masks. We used a varying number of repetitions per image, from as little as a single run to 64 runs. In Figure 3 we summarized the results. The performance seems to saturate at $\sim$ 32 repetitions.

LoRA In Table 5 we compare the performance of finetuning the original model parameters vs. training a low-rank version of them using LoRA. For finetuning without LoRA we used a learning rate of $1e-4$ with all other hyperparameters unchanged. We observe 0.6% improvement in image-level performance and 0.8% in pixel-level performance when using LoRA.

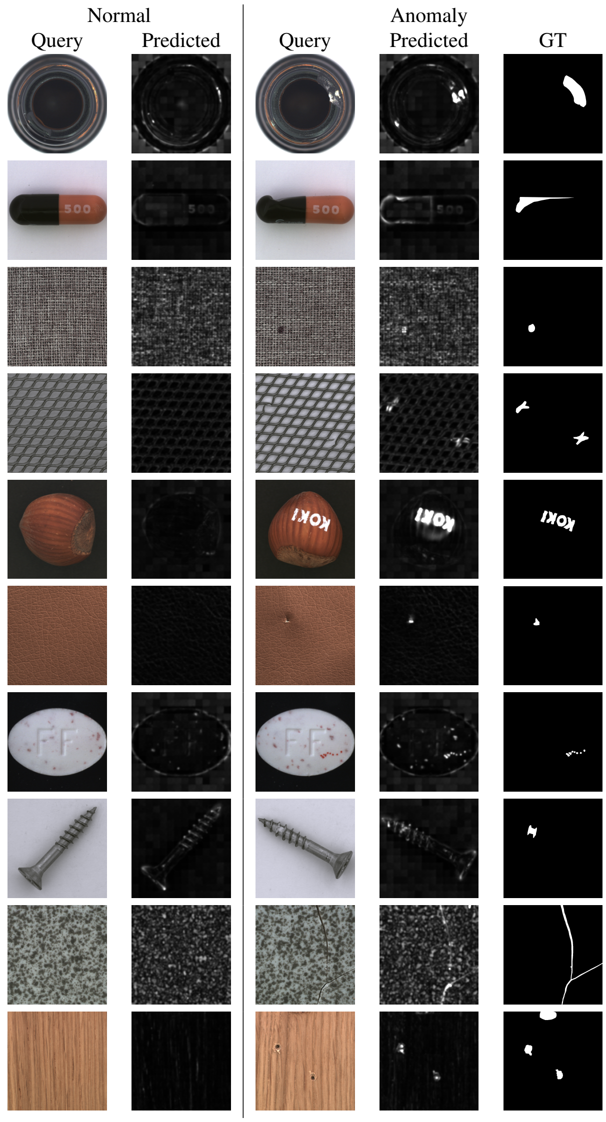


Figure 4. **1-shot examples** from the MVTec dataset. Usually the anomaly is detected but the predicted anomaly area tends to be smaller than the ground-truth, hence the pixel-level ROC-AUC is smaller than the image-level.

Training and inference time We compare training and inference time in Table 4. We tested the running time for the 0-shot and 1-shot cases. Time was measured on an A100 GPU. For PatchCore training includes extracting features using a pretrained model and performing CoreSet clustering and amounts for 4 seconds. The reported training time for MAEDAY was measured when training for 50 iterations and took around 100 seconds. There is a trade-off between finetuning MAEDAY which takes time and using MAEDAY in its 0-shot form which does not require training but with an accuracy drop of 1.5%. The inference was performed in batches for PatchCore (batch-size=32). For MAEDAY, each query image is processed individually since we use the batch dimension to run multiple instances of the same image with different random masks. Despite the parallelization in PatchCore (and the lack of in MAEDAY) the inference time is in the same order of magnitude with 0.07 seconds for PatchCore and 0.15 for MAEDAY. This is partially thanks to the fact the MAE's encoder inputs are only 25% of the tokens. The 75% of the tokens that need to be reconstructed are only introduced later as inputs to the decoder which is a much smaller network.

Method	Shots	Indoor	Outdoor	Mean
PatchCore	1	**98.2**	73.2	85.7
MAEDAY	0	95.6	**85.6**	**90.6**

Table 3. **Foreign Object Detection** ROC-AUC performance for zero-shot Foreign Object Detection (ZSFOD). MAEDAY, which is a 0-shot method, outperforms the 1-shot AD baseline.

	PatchCore 1-shot	MAEDAY 0-shot	MAEDAY 1-shot
Training	4s	0	100s
Infer. [per image]	0.07s	0.15s	0.15s

Table 4. Training and inference time. Tested for 0/1 shot. MAEDAY performs inference on a single image at a time to allow 32 repeats of the same image in the batch dimension (with different random mask). For PatchCore we used batch size of 32. Despite that, the inference time is not dramatically higher for MAEDAY compared to PatchCore. Tested on an A100 GPU.

4.1. Foreign Object Detection

We also tested a proof-of-concept of using MAEDAY for Zero-Shot Foreign Object Detection (ZSFOD). FOD is a very important task in several real-world scenarios, e.g. in airport runways, where even very small objects on the ground can be dangerous for the planes. Unlike classic FOD where models are trained for detecting specific types of objects, here no training data of either an empty surface or the objects to be detected are provided. We treat FOD as detecting anomalies in the background surface texture. We captured videos of the ground in two environments, indoors (wooden floor) and outdoors (asphalt pavement). Some of the frames contain foreign objects. Objects include larger tools, e.g. a wrench, and smaller objects, e.g. a bolt. We extracted and labeled 20-50 frames with foreign objects and a similar number without any object for each of the environments. This dataset will be released.

Since we are the first to perform the task of ZSFOD, we chose to compare MAEDAY against the SOTA 1-shot AD method, PatchCore [19]. This is a very strong baseline since it is using an object-free reference. Table 3 summarizes the results. We observed strong results by MAEDAY for ZSFOD, MAEDAY performs close to (Indoors) or better (Outdoors) compared to 1-shot PatchCore. Examples of images from the dataset along with their recovered outputs by MAEDAY and the final segmentation results are presented in Figure 6.

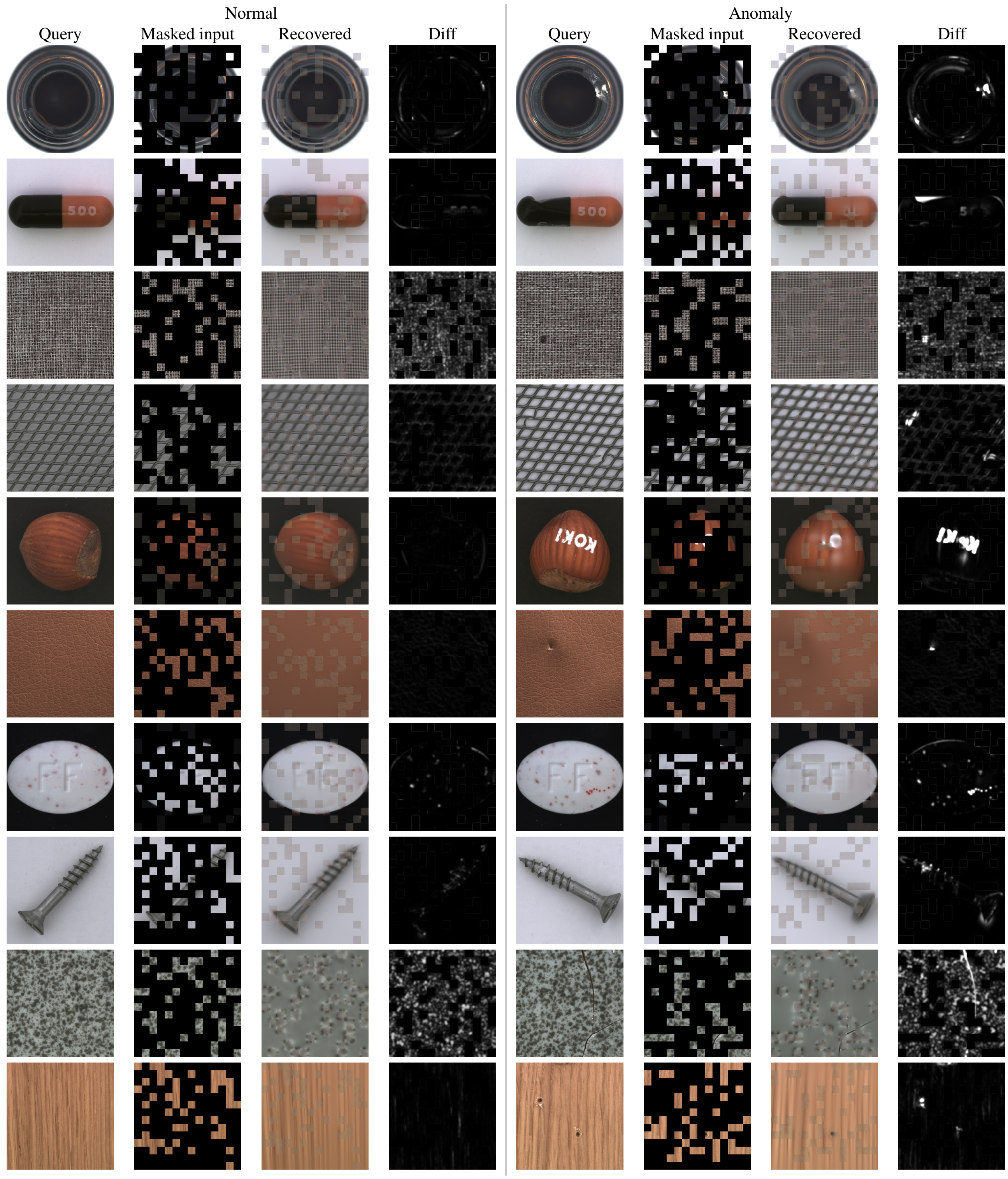


Figure 5. Examples of reconstruction for both normal and anomalous images from the MVTech dataset. The model is usually able to recover (a blurry version of) the normal images. In many cases this is enough for detecting anomalous regions.

5. Conclusions and Future Work

We have suggested MAEDAY, using an ImageNet pre-trained MAE for the task of few-shot anomaly detection (FSAD). While image-reconstruction-based methods are not the strongest methods for AD, we showed they provide additional valuable information. An ensemble of an

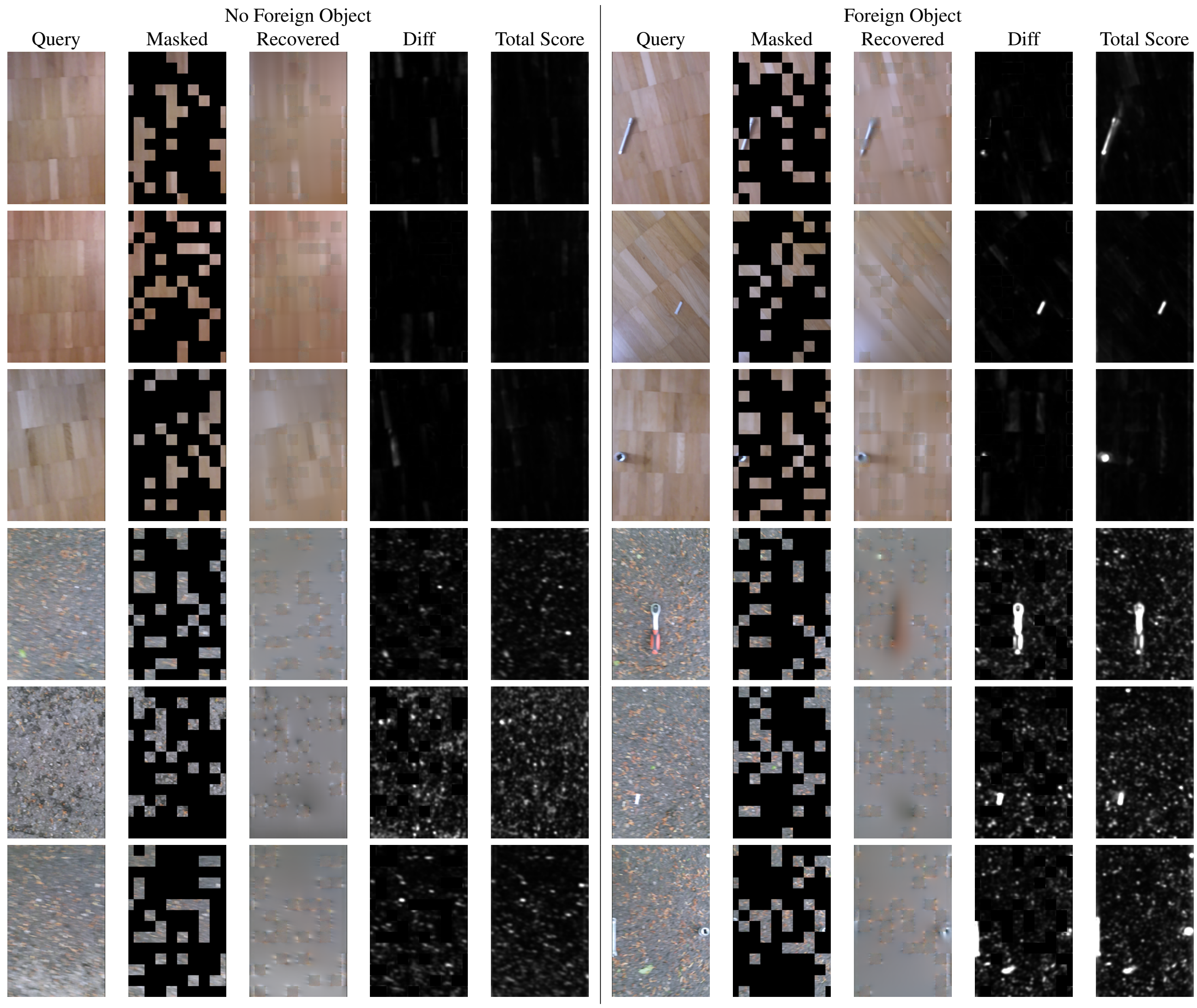


Figure 6. **ZSFOD MAEDAY Foreign Object Detection results** with neither clean surface reference nor object references. "Total Score" is the average of "Diff" produced with 32 different random masks applied to the same image.

	Full-finetuning	LoRA finetuning
Image ROC-AUC	75.4	76.0
Pixel ROC-AUC	70.9	71.6

Table 5. **LoRA ablation** Using LoRA for finetuning in a low-rank space improves performance.

embedding-based method and MAEDAY sets a new SOTA for FSAD.

We have also suggested the new Zero-Shot Anomaly-Detection task (ZSAD), performing anomaly detection with no reference images. We have shown MAEDAY can be used for this task and performs surprisingly well despite working with novel objects and textures. Specifically for textures, MAEDAY outperforms the reference-based FSAD SOTA baseline.

We explored a new task of Foreign Object Detection (FOD) on the ground, with no prior reference to either a free-of-objects surface or to the objects to be detected. We treated this problem as ZSAD where the objects are an anomaly in the surface texture. We showed better results for this task compared with SOTA FSAD where an image of the surface is provided for reference. The dataset is also made available to the community.

In future work MAEDAY can be extended to better use the few available shots in the FSAD setup. We can feed the model tokens (patches) from both the query image and the reference image(s). The model can be trained to use the transformer's attention mechanism to share information between the reference tokens and the query tokens. This way the recovered patches are not just guessed according to their surrounding patches but are more likely to fit the normal patch distribution.

References

[1] Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. The mvtec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. *International Journal of Computer Vision*, 129(4):1038–1059, 2021. 3

[2] Mei Chen, Takeo Kanade, Dean Pomerleau, and Henry A Rowley. Anomaly detection through registration. *Pattern Recognition*, 32(1):113–128, 1999. 2

[3] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. *arXiv preprint arXiv:2005.02357*, 2020. 1, 2

[4] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition*, pages 475–489. Springer, 2021. 1, 2

[5] Ye Fei, Chaoqin Huang, Cao Jinkun, Maosen Li, Ya Zhang, and Cewu Lu. Attribute restoration framework for anomaly detection. *IEEE Transactions on Multimedia*, 2020. 1, 3

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 2

[7] Denis Gudovskiy, Shun Ishizaka, and Kazuki Kozuka. Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 98–107, 2022. 2

[8] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 1, 3

[9] Geoffrey E Hinton. Connectionist learning procedures. In *Machine learning*, pages 555–610. Elsevier, 1990. 1, 2

[10] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 3

[11] Chaoqin Huang, Haoyan Guan, Aofan Jiang, Ya Zhang, Michael Spratling, and Yan-Feng Wang. Registration based few-shot anomaly detection. *arXiv preprint arXiv:2207.07361*, 2022. 1, 2

[12] Nathalie Japkowicz, Catherine Myers, Mark Gluck, et al. A novelty detection approach to classification. In *IJCAI*, volume 1, pages 518–523. Citeseer, 1995. 1, 2

[13] Ying Jing, Hong Zheng, Wentao Zheng, and Kaihan Dong. A pixel-wise foreign object debris detection method based on multi-scale feature inpainting. *Aerospace*, 9(9):480, 2022. 2

[14] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9664–9674, 2021. 2

[15] Travis Munyer, Daniel Brinkman, Xin Zhong, Chenyu Huang, and Iason Konstantzos. Foreign object debris detection for airport pavement images based on self-supervised localization and vision transformer. *arXiv preprint arXiv:2210.16901*, 2022. 2

[16] Travis Munyer, Pei-Chi Huang, Chenyu Huang, and Xin Zhong. Fod-a: A dataset for foreign object debris in airports. *arXiv preprint arXiv:2110.03072*, 2021. 2

[17] Mohammad Noroozi and Ankit Shah. Towards optimal foreign object debris detection in an airport environment. *Expert Systems with Applications*, 213:118829, 2023. 2

[18] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015. 2

[19] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14318–14328, 2022. 1, 2, 4, 6

[20] Marco Rudolph, Bastian Wandt, and Bodo Rosenhahn. Same same but differnet: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1907–1916, January 2021. 1, 2

[21] Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, pages 4–11, 2014. 1, 2

[22] 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. 2

[23] Shelly Sheynin, Sagie Benaim, and Lior Wolf. A hierarchical transformation-discriminating generative model for few shot anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8495–8504, October 2021. 1, 2

[24] Leo Tolstoy. *Anna karenina*. 1877. 1

[25] Julian Wyatt, Adam Leach, Sebastian M Schmon, and Chris G Willcocks. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 650–656, 2022. 3

[26] Xuan Xia, Xizhou Pan, Nan Li, Xing He, Lin Ma, Xiaoguang Zhang, and Ning Ding. Gan-based anomaly detection: A review. *Neurocomputing*, 2022. 2

[27] Xudong Yan, Huaidong Zhang, Xuemiao Xu, Xiaowei Hu, and Pheng-Ann Heng. Learning semantic context from normal samples for unsupervised anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3110–3118, 2021. 1, 3

[28] Jiawei Yu, Ye Zheng, Xiang Wang, Wei Li, Yushuang Wu, Rui Zhao, and Liwei Wu. Fastflow: Unsupervised anomaly detection and localization via 2d normalizing flows. *arXiv preprint arXiv:2111.07677*, 2021. 2

[29] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112:107706, 2021. 1, 3

[30] Houssam Zenati, Chuan Sheng Foo, Bruno Lecouat, Gaurav Manek, and Vijay Ramaseshan Chandrasekhar. Efficient gan-based anomaly detection. *arXiv preprint arXiv:1802.06222*, 2018. 2

[31] Jiaxin Zhang, Kyle Saleeby, Thomas Feldhausen, Sirui Bi, Alex Plotkowski, and David Womble. Self-supervised anomaly detection via neural autoregressive flows with active learning. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 2

[32] Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *European Conference on Computer Vision*, pages 392–408. Springer, 2022. 2