Learning Unsupervised Metaformer for Anomaly Detection

Jhih-Ciang Wu[1,2], Ding-Jie Chen[1], Chiou-Shann Fuh[2], and Tyng-Luh Liu[1]
[1]Institute of Information Science, Academia Sinica, Taiwan
[2]Department of Computer Science and Information Engineering, National Taiwan University, Taiwan

Abstract

Anomaly detection (AD) aims to address the task of classification or localization of image anomalies. This paper addresses two pivotal issues of reconstruction-based approaches to AD in images, namely, model adaptation *and* reconstruction gap. *The former generalizes an AD model to tackling a broad range of object categories, while the latter provides useful clues for localizing abnormal regions. At the core of our method is an unsupervised universal model, termed as* Metaformer*, which leverages both meta-learned model parameters to achieve high model adaptation capability and instance-aware attention to emphasize the focal regions for localizing abnormal regions,* i.e.*, to explore the reconstruction gap at those regions of interest. We justify the effectiveness of our method with SOTA results on the MVTec AD dataset of industrial images and highlight the adaptation flexibility of the universal Metaformer with multi-class and few-shot scenarios.*

1. Introduction

The principal goal of image Anomaly Detection (AD) is to classify whether an image depicts an abnormal version of the target object and if exist, localize those regions of anomaly. The technique to detect the various anomalies of interest is crucial for industrial inspection to ensure that the resulting products meet the required standards [15]. However, since the anomalies (or the defects) can deviate from the normal ones in numerous ways, it is hard to exhaustively pre-define an anomaly prior and collect enough anomaly data for training an anomaly detection model. Instead, most of the previous methods use anomaly-free data to construct its representative distribution for indirectly discriminating the deviated data as anomalies. Hence, the AD task is also known as out-of-distribution detection.

Driven by the attempt to model the one-class distribution of the anomaly-free data, the embedding-based [5, 28] and the reconstruction-based methods [24, 35, 39] comprise the two main trends for tackling the AD problem. The former seeks to learn an embedding function for making the anomaly-free data close to each other in the embedding space, and the latter aims to leverage a neural network for reconstructing each sample of the normal class. To determine the anomalies, the embedding-based methods draw on the resulting learned metrics, while the reconstruction-based ones employ reconstruction errors by contrast.

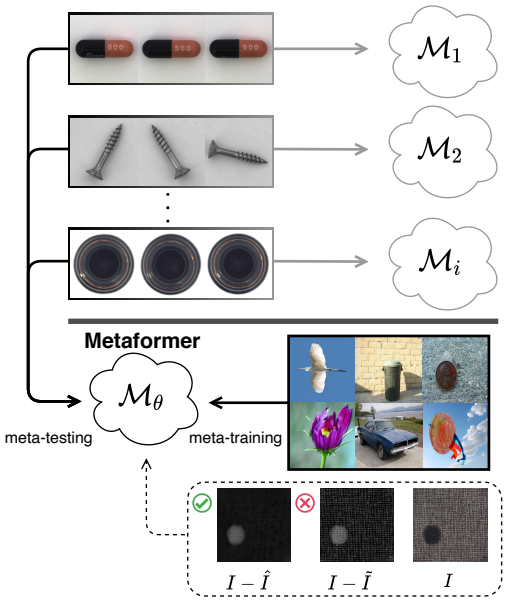

Figure 1. Model adaptation and reconstruction gap are two pivotal issues of reconstruction-based anomaly detection. Top: most AD techniques train a specific model for each category. Such an approach would become demanding as the number of categories increases. Bottom: we instead propose to train a universal model, *Metaformer* with instance-aware ability to simultaneously address the two key issues. Note that $I - \hat{I}$ and $I - \tilde{I}$ respectively denote the reconstruction errors with or without instance-aware attention.

We resolve the AD problem from the reconstruction-based point of view. Our formulation particularly pays attention to explore two key factors, *reconstruction gap* and *model adaptation*, in designing an effective AD framework. (See Figure 1.) Most of the reconstruction-based AD techniques include an autoencoder component. As the training data are typically sufficient and all from the "normal" class, a well-trained autoencoder is expected to satisfactorily re-

construct such samples not only in training but also in inference. The assumption implies that the reconstruction gap can be used to detect anomalies if a given image is out of the distribution of the normal class. Different from most existing reconstruction-based AD methods merely predicting the image-level anomalies, our approach introduces the *instance-aware* attention to further regulate the reconstruction gap for precisely localizing the pixel-level abnormal regions. Regarding the issue of model adaptation, we observe that prior arts on AD often need to collect a large number of anomaly-free examples to train an additional AD model for classifying a new object category. In real-world applications, an AD system could be deployed on edge devices of limited computational power, and such a data-eager training strategy may not be practical. To overcome the concern, we design a meta-learning strategy that enables our universal AD model to be fine-tuned with only a few anomaly-free supporting examples for handling a novel category.

The cornerstone of our method is the *Metaformer*, which leverages the meta-learned model parameters to effectively carry out the few-shot fine-tuning for performing AD of a novel object category and employs the instance-aware attention to emphasize the abnormal focal regions. Briefly speaking, the proposed Metaformer is a transformer-based instance-aware autoencoder that learns its model parameters using an unsupervised meta-learning strategy. Figure 1 overviews the proposed AD model. Figure 2 illustrates the steps of our meta-training, meta-testing, and inference. Figure 3 sketches the key components of our Metaformer.

To enable the Metaformer for efficient model adaptation, we learn its model parameters with an unsupervised meta-training strategy. Namely, the training comprises numerous few-shot image reconstruction tasks to obtain the parameters for the universal model, which can be rapidly fine-tuned using a few anomaly-free examples from each underlying novel class in meta-testing. It follows that the fine-tuned Metaformer is ready for performing the AD inference for the novel object category. We note that the meta-training stage does not have access to any of the images used in the meta-testing stage and the testing/inference stage of a novel category. In addition, to empower the Metaformer to more precisely uncover the abnormal regions, we introduce instance-aware attention to regularize the autoencoder (AE) to focus on the *instance area* while reconstructing an image. In our formulation, we first establish the instance prior based on saliency prediction and then carry out the AE regularization via an attention mechanism.

To the best of our knowledge, the proposed method is the first to address the image AD task by employing an adaptive instance-aware reconstruction method. We characterize our main contributions as follows:

- We introduce unsupervised few-shot meta-training to learn the universal *Metaformer* that exhibits the efficient flexibility of model adaptation to an arbitrary object category of interest.
- We couple the *instance-aware* attention mechanism with the autoencoder such that anomaly detection based on reconstruction gap can emphasize the area of target object rather than the distracting background.
- We provide extensive experimental results and comparisons to demonstrate that our method achieves the overall SOTA performance on both the anomaly classification and anomaly localization.

2. Related Work

In this section, we concisely review the recent research efforts relevant to the tasks of anomaly detection, meta-learning, and instance-aware attention.

Anomaly Detection The task of anomaly detection involves either image-level anomaly classification, which classifies whether an image is abnormal [2, 5, 13, 28, 35, 36], or pixel-level anomaly localization, which further localizes the abnormal regions [4, 5, 32].

Previous methods for anomaly detection are mostly cast as a one-class problem [1, 29] due to the scarcity of anomaly samples. For example, [14] introduces the memory block that enforces reconstructive images more like the given regular class. There are also several GAN-based approaches [2, 3, 7, 24, 25, 30] that apply adversarial training to enhance the performance of AD.

The MVTec AD dataset is recently introduced in [4], which includes the annotations of abnormal regions for evaluating an AD task about industrial inspection. Methodology-wise, techniques to deal with the AD problems can be divided into several essential types. The most popular one is the reconstruction-based approach that is conventionally established based on an autoencoder. The benchmark presented in MVTec AD comprises the convolutional autoencoder with ℓ_2 loss and structural similarity (SSIM) loss named ℓ_2-AE and SSIM-AE, respectively. GANomaly [2] extends the general autoencoder architecture and proposes an encoder-decoder-encoder network that reconstructs both the input image and the bottleneck with the adversarial training technique. Another autoencoder variant for AD [36] employs energy based model (EBM) and uses the energy score and the reconstruction error as the scoring function. Reconstruction-by-inpainting anomaly detection (RIAD) [35] treats AD as a self-supervised task that randomly crops patches for every instance and inpaints it via an autoencoder. A similar work [13] utilizes transformation in geometry and trains a multi-class model. Even though these autoencoder-based models achieve good AD accuracy, their reconstructed images generally tend to be

blurry. The shortcoming could cause detecting many abnormal regions, but most of them are not relevant to the AD task. In [32], the authors adopt an attention mechanism and designs an attention expansion loss to preserve spatial information. In comparison with this work, we address this issue by integrating the instance-aware attention mechanism with the autoencoder to improve the AD localization.

More recently, DifferNet [28] adopts the normalizing flow [26] as a density estimation of the image features extracted by convolutional neural networks. Then the anomaly score is computed based on the likelihoods of multiple transformations per image. Using the teacher-student knowledge distilling approach, US [5] learns a discriminative embedding for making the student networks produce regression errors and uncertainties. To compare the effectiveness of feature distribution for anomaly localization, the US model is further evaluated by fitting different algorithms such as K-Means, OC-SVM, and 1-NN into the teacher network. Observe that existing methods perform data augmentation or model ensemble, which results in relatively high cost. Moreover, they usually train a single model per category in the dataset. We tackle this issue by applying meta-learning to train an adapted universal model.

To address the AD task on retinal images, P-Net [39] employs the external edge-structure information to encode the relation between structure and texture for the subsequent image reconstruction to detect the abnormal regions. They obtain structure information using an off-the-shelf edge detector. In comparison and to be explained later, we learn an instance-prior generator and Metaformer using the same dataset in an unsupervised manner. This advantage brings more flexibility and less cost for real-world applications.

Meta-learning The objective of meta-learning is to enable a model for fast adaptability by training it on various learning tasks. The learned model is hence able to adapt to novel tasks with a few supporting examples. To this end, one sort of meta-learning method [11, 12, 19, 22] aims to explicitly maximize the model sensitivity concerning the novel-task losses against the model parameters. To borrow the fast model adaptability for addressing the model adaptation issue in the AD task, we employ Model-Agnostic Meta-Learning (MAML) [11] to train parameters of Metaformer for sensitivity on a given task distribution. Note that we explicitly train the Metaformer in an unsupervised manner. Precisely, the task distribution for use is defined without any additional annotations or attributes.

Instance-aware Attention To address the reconstruction gap for precisely localizing the pixel-level abnormal regions, we introduce the instance-aware attention to regulate the reconstructed image focusing on the instance regions. In practice, we factor the instance-aware attention into an

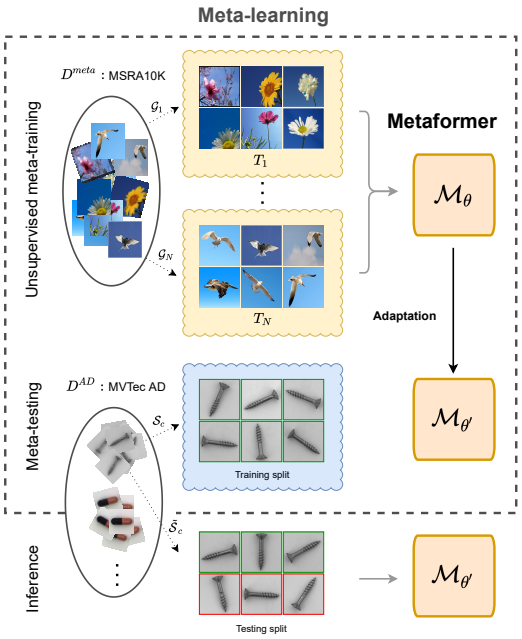

Figure 2. The model learning pipeline. We first train Metaformer using various tasks T_i from MSRA10K in the meta-training phase. The learned universal model $\mathcal{M}_\theta$ updates parameters efficiently to obtain $\mathcal{M}_{\theta'}$ in the meta-testing stage. The fine-tuned model $\mathcal{M}_{\theta'}$ measures the performance using the original testing split $\tilde{\mathcal{S}}_c$ in MVTec AD. The green boxes and red boxes indicate the anomaly-free examples and anomalous examples, respectively.

instance-aware generator and an attention mechanism. The former depicts the region of interest, and the later enable our image reconstruction focusing on these instance regions.

Our instance-aware generator considers the saliency detection approach [8, 18, 20, 37, 38] to depict the region of interest for anomaly detection. A saliency detection method aims at finding salient objects in an image. For concerning the unsupervised learning, we employ the generator within the Visual-Effect GAN [8] as our instance-aware generator, which is trained by using the annotation-free Flickr images. Though the other methods [18, 20, 37] are unsupervised ones, the hand-crafted methods [18, 20] generate noisy results, and the deep detection [37] needs to ensemble multiple saliency detection results retrieved from other methods.

Our attention mechanism aims to leverage the instance-aware prior for making the autoencoder pay attention to reconstructing the region of interest. To this end, we form the dependencies between the instance-aware prior and the AE output via the transformer [31], which is devised for addressing the machine translation task yet shows its convincing improvement on various tasks such as image caption [10], instance segmentation [21], sketch classification [27], and image super-resolution [34].

3. Method

We introduce the Metaformer, which aims to tackle the issues of model adaptation and reconstruction gap, to classify the image-level anomalies and localize the pixel-level abnormal regions. For dealing with the model adaption issue, we learn one single Metaformer model by leveraging the meta-learning strategy, which comprises the steps of meta-training and meta-testing. For the reconstruction gap issue, we propose an instance-aware image reconstruction accomplished within Metaformer. To illustrate our method, we first elaborate on our model learning strategy and then show the components of the Metaformer.

3.1. Model Learning Strategy

Our model learning strategy includes the *meta-training* step and *meta-testing* step. The learned model is then available for tackling AD tasks in the *inference* step. Figure 2 shows the pipeline and required data in these steps. Briefly, the *unsupervised meta-training* is used to learn the universal Metaformer for capturing the concept of the general category-independent instance-aware image reconstruction. While dealing with a category-dependent AD task, *i.e.*, one specific novel image category, the meta-trained model's fast adaptive tuning ability enables the universal Metaformer to be rapidly fine-tuned with a few anomaly-free examples of that category in the *meta-testing* step. Therefore, the fine-tuned model is ready to carry out the anomaly detection of that image category from AD dataset in the *inference* step.

3.1.1 Unsupervised Meta-training & Meta-testing

In meta-learning, a *meta-task* implies the application that needs to be achieved by the learned model, and we define the meta-task as a few-shot image reconstruction. We employ the MAML algorithm to carry out the meta-training step and the meta-testing step, yet the unsupervised clustering is considered for sampling a meta-task while learning the universal Metaformer within the meta-learning step.

Meta-task In our meta-learning, each meta-task mimics one few-shot image reconstruction, *i.e.*, only a few supporting examples are allowed for model tuning per reconstruction. The standard AD datasets often evaluate an AD model through category by category testing, which implies that a single meta-task is formulated on an image group of a specific category. Without losing generality, we assume that a meta-task is defined on a set of similar-structure images.

Our unsupervised meta-training uses numerous few-shot image reconstruction meta-tasks to train the model to capture the general category-independent image reconstruction concept. Precisely, given a meta-training dataset $\mathcal{D}^{meta}$, we first extract features using ResNet18 [16] and perform k-means algorithm on feature space to divide $\mathcal{D}^{meta}$ into N coarse groups. We then trim the groups according to the structural similarity to the center features and form the fine image groups $\{\mathcal{G}_i\}$. For a meta-task T_i dealing with one specific image category, we formally define the meta-task $T_i = \{\mathcal{M}_\theta, \mathcal{G}_i \subset \mathcal{D}^{meta}\}$ comprises an AD model $\mathcal{M}_\theta$ and a fine image group $\mathcal{G}_i$. The structural similarity between images x and y derived from [33] is defined as

$$ssim(x, y) = \frac{4\mu_x\mu_y\sigma_{xy} + \epsilon}{(\mu_x^2 + \mu_y^2)(\sigma_x^2 + \sigma_y^2) + \epsilon}, \qquad (1)$$

where μ, σ are average intensity and standard deviation of the given image. ϵ is a small constant that prevents zero division. Note that the structure-based clustering is carried out according to the structural similarity among images, and the ground-truth for each reconstructed image is essentially its original one without accessing any annotations. Hence, given an image $I \in \mathcal{G}_i$, our meta-training step aims to learn the Metaformer $\mathcal{M}_\theta$ for reconstructing the image I as $\hat{I} = \mathcal{M}_\theta(I)$, where the θ denotes the model parameters.

To capture the category-dependent image reconstruction concept of a specific image category c in an AD dataset, the universal model $\mathcal{M}_\theta$ simply fine-tunes its parameters with the training split $\mathcal{S}_c \subset \mathcal{D}^{AD}$ of AD dataset (see Figure 2). We define the meta-task $T_c = \{\mathcal{M}_{\theta'}, \mathcal{S}_c \subset \mathcal{D}^{AD}\}$ comprises a fine-tuned AD model $\mathcal{M}_{\theta'}$ that is trained with the subgroup $\mathcal{S}_c$.

Meta-training The meta-training aims to learn a universal AD model with high adaptation capability. To this end, our unsupervised meta-training employs numerous few-shot image reconstructions as meta-tasks to capture the general concept of the reconstruction, where each image group for the corresponding meta-task results from an unsupervised image clustering.

Given the Metaformer $\mathcal{M}_\theta$, the meta-training process adapts $\mathcal{M}_\theta$ concerning numerous meta-task T_i. Intuitively,

Algorithm 1 Unsupervised meta-learning for Metaformer

Hyperparameters: α, β
Input: Wild dataset $\mathcal{D}^{meta}$, training split $\mathcal{S}_c$ of AD dataset

Construct T_i via grouping images in $\mathcal{D}^{meta}$ into N clusters
Initialize θ and build Metaformer $\mathcal{M}_\theta$
/ Unsupervised meta-training for Metaformer */*
for $i = 1$ **to** N **do**
  Evaluate $\nabla_\theta \mathcal{L}(T_i)$
  Compute gradients for adaptive parameters with the optimizer: $\theta_i' = \theta - \alpha\nabla_\theta\mathcal{L}(T_i)$
end for
Update $\theta \leftarrow \theta - \beta\nabla_\theta \sum_{i=1}^{N} \mathcal{L}(\{\mathcal{M}_{\theta_i'}, \mathcal{G}_i\})$
/ Meta-testing for task adaptation */*
Update $\theta' \leftarrow \theta - \alpha\nabla_\theta\mathcal{L}(T_c)$

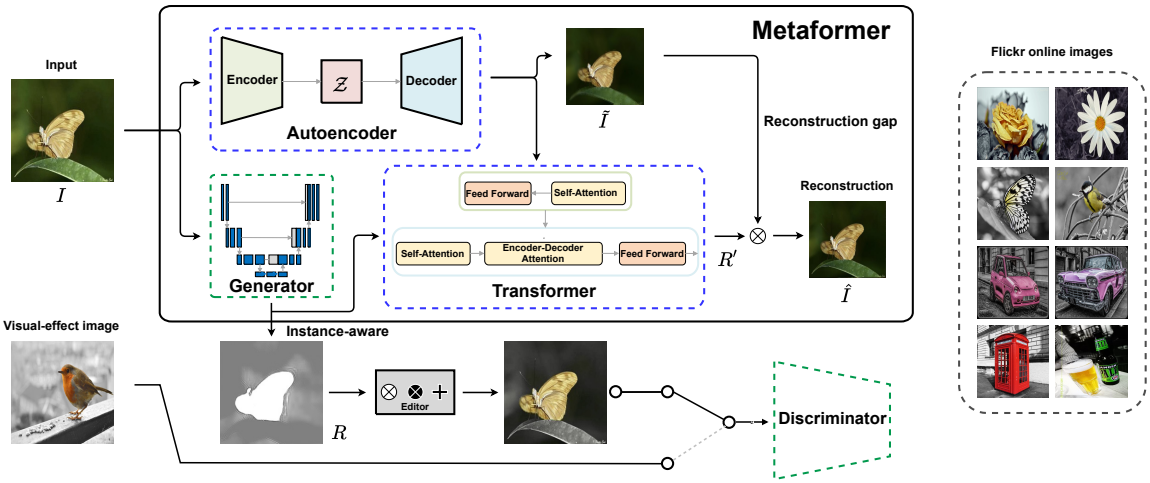

Figure 3. An overview of our Metaformer. The Metaformer consists of three modules: a generator, an autoencoder, and a transformer. We train a GAN (green dashed boxes) using MSRA10K and Flickr online images with specific visual effect (black dashed box). All parameters of the generator are fixed and put into Metaformer after training. Next, we start to train the remaining modules (blue dashed boxes).

the image reconstruction loss is derived from the difference between the reconstructed image $\hat{I} = \mathcal{M}_\theta(I)$ and the input image I. We then define the meta-task loss $\mathcal{L}(T_i)$ as

$$\mathcal{L}(T_i) = \sum_{I \in \mathcal{G}_i} L(\mathcal{M}_\theta(I); I)\,, \tag{2}$$

where the loss function $L(a; b)$ measures the difference between a and b. With the loss, the model parameters can thus be updated from θ to θ_i' concerning the meta-task T_i via gradient updates:

$$\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}(T_i)\,, \tag{3}$$

where α is the learning rate. Since meta-training aims to learn robust model parameters θ to form the universal model $\mathcal{M}_\theta$, the task-adapted parameters θ' in (3) will be minimized its reconstruction loss among all meta-tasks via loss calculation for retrieving the best θ. Therefore, the objective function of meta-training is formulated as

$$\min_\theta \sum_{i=1}^{N} \mathcal{L}(\{\mathcal{M}_{\theta_i'}, \mathcal{G}_i\})\,. \tag{4}$$

As a result, the meta-training loss in (4) is summing over all tasks, yet we sample a mini-batch of meta-tasks per training iteration as MAML. Therefore, the meta-training optimization across meta-tasks and the model parameters are hence updated as

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_{i=1}^{N} \mathcal{L}(\{\mathcal{M}_{\theta_i'}, \mathcal{G}_i\})\,, \tag{5}$$

where β is the meta learning rate.

Meta-testing In the meta-testing step, the meta-trained Metaformer $\mathcal{M}_\theta$ is expected to reconstruct a given image from a category-dependent AD task after performing the fast adaptation with a few supporting examples. In this paper, each novel meta-task is defined on the specific-category set of the anomaly-free training images in the MVTec AD dataset. Similar to the updating function in (3), the model parameters are updated concerning the image group $\mathcal{S}_c$ in MVTec AD dataset via one gradient update:

$$\theta_c' = \theta - \alpha \nabla_\theta \mathcal{L}(\{\mathcal{M}_{\theta_c}, \mathcal{S}_c\})\,. \tag{6}$$

We summarize the meta-learning procedure in Algorithm 1.

Loss We define loss function as a couple of the mean squared error l_{mse} and the structural similarity loss l_{ssim} [33] as

$$L(\hat{I}; I) = l_{mse}(\hat{I}, I) + \lambda l_{ssim}(\hat{I}, I)\,, \tag{7}$$

where λ is the weight between the loss terms.

3.1.2 Inference

After a few-shot model tuning with the specific-category supporting images, the tuned Metaformer $\mathcal{M}_{\theta'}$ is ready to reconstruct the specific-category images in testing split $\tilde{\mathcal{S}}_c$ for inference.

3.2. Instance-aware Metaformer

To resort to the reconstruction gap issue for precisely localizing anomaly regions, we design the Metaformer as a transformer-based instance-aware autoencoder consisting of three modules: *autoencoder*, *instance-prior generator*, and *transformer*. The details of Metaformer are illustrated in Figure 3. We describe each component as follows.

Autoencoder Most conventional AD methods utilize the autoencoder-like models to deal with anomaly detection since only anomaly-free images allowed for model training. Our autoencoder module employs the similar model designed in [5], which is symmetrical with five layers in both the encoder and decoder. The detailed architectures are provided in the supplementary material. The autoencoder $\mathcal{A}$ first encodes an input image I into a latent representation, and then it decodes the latent representation $\mathcal{Z}$ into an intermediate image $\tilde{I}$. Briefly, we represent this module as $\tilde{I} = \mathcal{A}(I)$.

Instance-prior Generator We employ a module trained in an unsupervised manner for extracting the instance-prior to depict the foreground regions. Inspired by [8], our instance-prior generator $\mathcal{P}$ employs the generator part within the Visual-Effect GAN, which comprises a generator, an editor, and a discriminator. We use the same architecture as [8], and we use 4,061 Flickr online images with `color selectivo` visual effects (see Figure 3) to learn to extract the internal representations for discriminating the image foreground from the image background. Please refer to [8] for the details of training the GAN model.

Briefly, we represent the response map R generated from this module as $R = \mathcal{P}(I)$. Note that once the Visual-Effect GAN has been trained, we directly use its generator as our instance-prior generator without any parameter fine-tuning. Namely, only the model parameters within the autoencoder module and the transformer module will be tuned during our meta-learning process.

Transformer Our attention mechanism is the *transformer* [31], which is purposed to solve the language translation problem. The transformer comprises an encoder-decoder pair, which performs the intra-attention on both the encoder and decoder, and also carries out the inter-attention between the encoder-decoder pair. In practice, the transformer's attention mechanism reformulates the feature representation of the encoder's input by concerning the decoder's input, hence enabling the encoder's input to simulate the feature representation possessed by the decoder's input.

In an image reconstruction process within our AD task, we propose to train an instance-aware autoencoder for addressing the reconstruction gap issue. Our idea is leveraging an instance-prior to highlight the reconstruction errors locating in the foreground area. In this way, we can guide autoencoder parameters focusing on the image foreground due to the relatively large reconstruction errors compared to the image background. Instead of directly applying the instance-prior to the reconstructed image, our transformer module reformulates the feature representation of the instance-prior by concerning the reconstructed image. Hence, the reformulated instance-prior can be treated as a re-weighting indicator to be applied onto the reconstructed image by element-wise multiplication. Here, we represent the reformulated instance-prior R' generated from this module as $R' = \mathcal{T}(R)$. Therefore, the instance-aware reconstructed image $\hat{I}$ is obtained as

$$\hat{I} = \mathcal{A}(I) \otimes \mathcal{T}(\mathcal{P}(I); \mathcal{A}(I)) \tag{8}$$

where $\mathcal{A}$, $\mathcal{P}$, and $\mathcal{T}$ denote the autoencoder, instance-prior generator, and transformer, respectively. The transformer $\mathcal{T}(a; b)$ denotes that feature a is reformulated concerning feature b. In practice, we use the vanilla transformer of eight heads in three layers.

The anomalous regions can be determined by the difference between I and $\hat{I}$ in a self-supervised manner once the reconstructed image is generated.

4. Experiments

Implementation Details. Our meta-learning formulates the meta-tasks by using the MSRA10K dataset [9] as $\mathcal{D}^{meta}$ and MVTec AD dataset [4] as $\mathcal{D}^{AD}$. Before the meta-learning step, we use structural similarity and k-means to construct meta-tasks that contain around 200 images per group in our experiment. We set hyperparameters $\alpha = \beta = 0.0001$ and $\lambda = 0.1$. The optimizer is Adam, with learning rates decaying 10% every 20 epochs in meta-training stage. The total epochs in meta-training and meta-testing are 100 and 30, respectively. Note that our instance-aware generator is trained on its own. We train the Visual-Effect GAN for 200 epochs. The optimizer is Adam, with learning rates of 0.0002 decaying 10% every 50 epochs. All of our networks are trained under the batch size of 64.

Dataset. We evaluate our method's performance with state-of-the-art methods selected from anomaly classification and anomaly localization on MVTec AD [4] and Magnetic Tile Defects (MTD) dataset [17]. Each image from the two datasets is resized to 256 $\times$ 256, and only the anomaly-free images are used for training. The MVTec AD contains 5,354 images in 15 categories of textures and objects. Each category comprises anomaly-free and several defect types such as broken, contamination, and bent objects. The various cases of irregular defects cause the MVTec AD challenging for anomaly detection. The MTD dataset comprises 1,344 instances of 952 being anomaly-free ones and 392 anomalous. The MTD dataset has five defect types of break, blowhole, crack, fray, and uneven. While evaluating this dataset, we adopt the equivalent setting of [28] to reserves 20% anomaly-free instances randomly and all the anomalous images for evaluation.

Metrics. For assessing the performance in image-level anomaly classification, we calculate the Area Under Curve (AUC), which is the standard threshold-independent metric used as [2]. For comparing the performance in pixel-level

Table 1. Comparison the methods of image-level anomaly classification and pixel-level anomaly localization on the MVTec AD dataset.

Category		Image-level Anomaly Classification Methods (AUC metric)							Pixel-level Anomaly Localization Methods (PRO metric)						
		GeoTrans	GANomaly	DSEBM	US	RIAD	DifferNet	Metaformer	ℓ_2-AE	1-NN	OC-SVM	K-Means	SSIM-AE	US $p = 65$	Metaformer
Textures	carpet	0.437	0.699	0.413	0.916	0.842	0.929	**0.940**	0.456	0.512	0.355	0.253	0.647	0.695	**0.878**
	grid	0.619	0.708	0.717	0.810	**0.996**	0.840	0.859	0.582	0.228	0.125	0.107	0.849	0.819	**0.865**
	leather	0.841	0.842	0.416	0.882	**1.000**	0.971	0.992	0.819	0.446	0.306	0.308	0.561	0.819	**0.959**
	tile	0.417	0.794	0.690	0.991	0.987	**0.994**	0.990	0.897	0.822	0.722	0.779	0.175	**0.912**	0.881
	wood	0.611	0.834	0.952	0.977	0.930	**0.998**	0.992	0.727	0.502	0.336	0.411	0.605	0.725	**0.848**
Objects	bottle	0.744	0.892	0.818	0.990	**0.999**	0.990	0.991	0.910	0.898	0.850	0.495	0.834	**0.918**	0.888
	cable	0.783	0.757	0.685	0.862	0.819	0.959	**0.971**	0.825	0.806	0.431	0.513	0.478	0.865	**0.937**
	capsule	0.670	0.732	0.594	0.861	**0.884**	0.869	0.875	0.862	0.631	0.554	0.387	0.860	**0.916**	0.879
	hazelnut	0.359	0.785	0.762	0.931	0.833	0.993	**0.994**	0.917	0.861	0.616	0.698	0.916	**0.937**	0.886
	metal nut	0.813	0.700	0.679	0.820	0.885	0.961	**0.962**	0.830	0.705	0.319	0.351	0.603	**0.895**	0.869
	pill	0.630	0.743	0.806	0.879	0.838	0.888	**0.901**	0.893	0.725	0.544	0.514	0.830	**0.935**	0.930
	screw	0.500	0.746	**0.999**	0.549	0.845	0.963	0.975	0.754	0.604	0.644	0.550	0.887	0.928	**0.954**
	toothbrush	0.972	0.653	0.781	0.953	**1.000**	0.986	**1.000**	0.822	0.675	0.538	0.337	0.784	0.863	**0.877**
	transistor	0.869	0.792	0.741	0.818	0.909	0.911	**0.944**	0.728	0.680	0.496	0.399	0.725	0.701	**0.926**
	zipper	0.820	0.745	0.584	0.919	0.981	0.951	**0.986**	0.839	0.512	0.355	0.253	0.665	0.933	**0.936**
Mean		0.672	0.762	0.709	0.877	0.917	0.949	**0.958**	0.791	0.640	0.479	0.423	0.694	0.857	**0.901**

Table 2. Comparison the methods of image-level anomaly classification on the MTD dataset.

Method	GeoTrans	GANomaly	DSEBM	ADGAN	OCSVM	1-NN	DifferNet	Metaformer
mAUC	0.755	0.766	0.572	0.464	0.587	0.800	0.977	**0.993**

anomaly localization, we use the per-region-overlap (PRO) metric proposed by Bergmann *et al*. [6]. We follow [5] to compute the PRO value that scans over false-positive rates by increasing the thresholds to keep the false-positive rates within the range $[0, 0.3]$. The main property of the PRO metric is that the weights for each overlap region are equal. Hence, the localization that is only focusing on the large regions will be penalized.

4.1. Anomaly Classification

In the image-level anomaly classification task, we compare our model to GeoTrans [13], GANomaly [2], DSEBM [36], US [5], RIAD [35], and DifferNet [28].

The left part in Table 1 shows the comparison results of anomaly classification on the MVTec AD dataset. As seen in Table 1, our model outperforms the other state-of-the-art models in seven classes and matches in the toothbrush category. Recent works involve MTD for AD problems with proper data splits. We follow the setting described in [28] to form training/testing splits. Our mean AUC, as shown in Table 2, achieves new art on both MVTec AD and MTD, improving around 1% and 1.6%, respectively.

4.2. Anomaly Localization

We consider the AD methods in the PRO metric for the pixel-level anomaly localization task, including ℓ_2-AE [4], 1-NN [23], OCSVM, K-Means, SSIM-AE, and US [5]. The results we compared are all reported by [5].

The right part in Table 1 shows the comparison results of anomaly localization on the MVTec AD dataset. In particular, $p = 65$ is the hyperparameter of receptive field size for training the teacher network in US. As shown in Table 1, Metaformer achieves the highest score in nine classes, which implies that Metaformer can detect both small and large defects better. Besides, our method improves the current score by about 4.4%.

4.3. Ablation Study

Model Configuration To verify the effectiveness of each module in the Metaformer, we consider three model configurations, *i.e*., autoencoder ($\mathcal{A}$), autoencoder with a generator ($\mathcal{A} + \mathcal{P}$), and the full Metaformer model ($\mathcal{A} + \mathcal{P} + \mathcal{T}$). Table 3 shows the ablation study on these configurations. All the three model configurations are trained in our meta-learning strategy, as shown in Algorithm 1. Precisely, the reconstruction errors of the configuration ($\mathcal{A}$) are merely derived from the stand-along autoencoder. The configuration ($\mathcal{A} + \mathcal{P}$) employs the additional instance-prior yet directly affects the reconstructed image $\tilde{I}$ using the element-wise multiplication. Our full Metaformer model ($\mathcal{A} + \mathcal{P} + \mathcal{T}$) employs the transformer to reformulates the instance-prior concerning the reconstructed image $\tilde{I}$ before such the element-wise multiplication. The full model obtains the best score in all categories and further enhances the mean PRO score by 1.8% compare to ($\mathcal{A}+\mathcal{P}$). Besides, the result shows that all the modules bring positive contributions, and using such the instance-prior helps the AD task as demonstrated on the performance improvement from 1.9% to 3.7% mPRO metric.

Few-shot Scenario As we mentioned in the introduction, the edge devices for deploying the AD systems prefer the fast model adaptation in real-world AD applications. Intuitively, an AD model using fewer supporting examples for

Table 3. Effect of each module in the proposed Metaformer.

Category		Model Configuration		
		$\mathcal{A}$	$\mathcal{A}+\mathcal{P}$	$\mathcal{A}+\mathcal{P}+\mathcal{T}$
Textures	carpet	0.852	0.877	**0.878**
	grid	0.844	0.852	**0.865**
	leather	0.901	0.940	**0.959**
	tile	0.850	0.861	**0.881**
	wood	0.797	0.824	**0.848**
Objects	bottle	0.852	0.856	**0.888**
	cable	0.880	0.914	**0.937**
	capsule	0.861	0.878	**0.879**
	hazelnut	0.840	0.851	**0.886**
	metal_nut	0.817	0.846	**0.869**
	pill	0.917	0.915	**0.930**
	screw	0.941	0.943	**0.954**
	toothbrush	0.847	0.864	**0.877**
	transistor	0.828	0.886	**0.926**
	zipper	0.939	0.935	**0.936**
mPRO		0.864	0.883	**0.901**

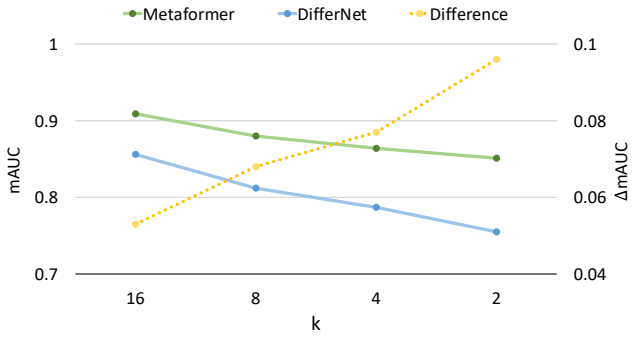

Figure 4. Effect of available supporting examples on model's performance. The green and the blue lines indicate the mAUC values (given by the left y-axis) of Metaformer and DifferNet for various k-shot settings, respectively. The yellow dotted line shows the mAUC differences (by the right y-axis) between the two methods.

its model adaptation shows the lower cost of model training or training-data collection. Here we experiment with a few-shot configuration on MVTec AD dataset to discuss such a model adaption issue. The main AD competitor is DifferNet in this experiment, and we reproduce its results of the few-shot setting with its released code. Figure 4 shows the experimental results, in which each k-shot indicates that there are only k supporting images available for model training. In Figure 4, our Metaformer outperforms DifferNet in all numbers of k, especially the performance gap is more noticeable when using fewer supporting examples. The results demonstrate that our model shows the robust ability of the few-shot model adaptation for dealing with the AD task.

4.4. Visualization

The visualization for qualitative analysis with the corresponding instance-prior is presented in Figure 5. We show

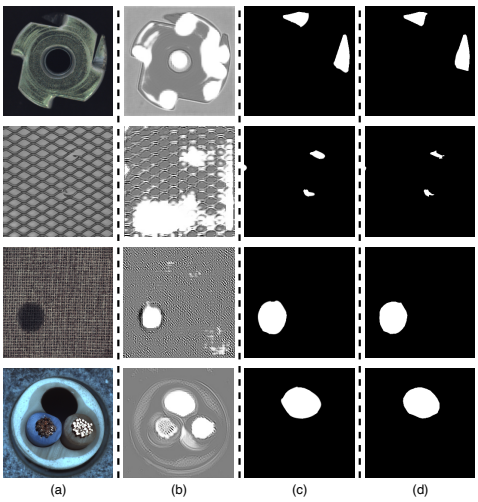

Figure 5. Qualitative results of the proposed Metaformer. (a) The selected images from the MVTec AD dataset. (b) The instance-priors obtained by the generator $\mathcal{P}$. (c) The thresholded predictions from Metaformer. (d) The ground-truths.

that the response map R provides a compelling clue that enhances the reconstruction in focal regions. On the other hand, the Metaformer is trained with a wild dataset consisting of various anomaly-free images. Our method, therefore, can reconstruct various low-level features and manifest abnormal regions. We provide some failure cases in the supplementary material for more discussions.

5. Conclusion

We have presented our universal Metaformer trained via unsupervised meta-learning to tackle the two common issues existing in most previous reconstruction-based AD methods, *i.e.*, model adaptation and reconstruction gap. Rather than maintaining one specific model per image category as other AD methods, our Metaformer resolves the model adaptation issue by an unsupervised meta-learning strategy to learn one universal model. With such a universal model, our Metaformer is able to tackle a novel category via few-shot fine-tuning. To deal with the reconstruction gap issue for precisely localizing the abnormal regions, our Metaformer employs an instance-aware transformer to leverage the instance-priors for guiding the image reconstruction. With such guidance, the autoencoder can focus on the instance area for precisely reconstructing its detail regions. The experimental results on the MVTec AD dataset show the notable performance gain over current state-of-the-art methods, demonstrating that our Metaformer can detect real-world anomaly images for industrial inspection.

Acknowledgement. This work was supported in part by the MOST grants 110-2634-F-001-009 and 110-2221-E-001-017 of Taiwan. We are grateful to National Center for High-performance Computing for providing computational resources and facilities.

References

[1] Davide Abati, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Latent space autoregression for novelty detection. In *CVPR*, pages 481–490, 2019. 2

[2] Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *ACCV*, pages 622–637. Springer, 2018. 2, 6, 7

[3] Samet Akçay, Amir Atapour-Abarghouei, and Toby P Breckon. Skip-ganomaly: Skip connected and adversarially trained encoder-decoder anomaly detection. In *IJCNN*, pages 1–8. IEEE, 2019. 2

[4] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*, pages 9592–9600, 2019. 2, 6, 7

[5] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *CVPR*, pages 4183–4192, 2020. 1, 2, 3, 6, 7

[6] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In *VISAPP*, pages 372–380. SciTePress, 2019. 7

[7] Philippe Burlina, Neil Joshi, and I-Jeng Wang. Where's wally now? deep generative and discriminative embeddings for novelty detection. In *CVPR*, June 2019. 2

[8] Ding-Jie Chen, Jui-Ting Chien, Hwann-Tzong Chen, and Tyng-Luh Liu. Unsupervised meta-learning of figure-ground segmentation via imitating visual effects. In *AAAI*, volume 33, pages 8159–8166, 2019. 3, 6

[9] Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, Philip H. S. Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(3):569–582, 2015. 6

[10] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *CVPR*, pages 10578–10587, 2020. 3

[11] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pages 1126–1135, 2017. 3

[12] Chelsea Finn, Aravind Rajeswaran, Sham M. Kakade, and Sergey Levine. Online meta-learning. In *ICML*, pages 1920–1930, 2019. 3

[13] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In *NeurIPS*, pages 9758–9769, 2018. 2, 7

[14] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *ICCV*, pages 1705–1714, 2019. 2

[15] Saeed Hosseinzadeh Hanzaei, Ahmad Afshar, and Farshad Barazandeh. Automatic detection and classification of the ceramic tiles' surface defects. *Pattern Recognit.*, 66:174–189, 2017. 1

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, June 2016. 4

[17] Yibin Huang, Congying Qiu, and Kui Yuan. Surface defect saliency of magnetic tile. *Vis. Comput.*, 36(1):85–96, 2020. 6

[18] Bowen Jiang, Lihe Zhang, Huchuan Lu, Chuan Yang, and Ming-Hsuan Yang. Saliency detection via absorbing markov chain. In *ICCV*, pages 1665–1672, 2013. 3

[19] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 3

[20] Xiaohui Li, Huchuan Lu, Lihe Zhang, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via dense and sparse reconstruction. In *ICCV*, pages 2976–2983, 2013. 3

[21] Justin Liang, Namdar Homayounfar, Wei-Chiu Ma, Yuwen Xiong, Rui Hu, and Raquel Urtasun. Polytransform: Deep polygon transformer for instance segmentation. In *CVPR*, pages 9131–9140, 2020. 3

[22] Dougal Maclaurin, David Duvenaud, and Ryan P. Adams. Gradient-based hyperparameter optimization through reversible learning. In *ICML*, pages 2113–2122, 2015. 3

[23] 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. 7

[24] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. Ocgan: One-class novelty detection using gans with constrained latent representations. In *CVPR*, pages 2898–2906, 2019. 1, 2

[25] Stanislav Pidhorskyi, Ranya Almohsen, and Gianfranco Doretto. Generative probabilistic novelty detection with adversarial autoencoders. In *NeurIPS*, pages 6822–6833, 2018. 2

[26] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *ICML*, pages 1530–1538, 2015. 3

[27] Leo Sampaio Ferraz Ribeiro, Tu Bui, John Collomosse, and Moacir Ponti. Sketchformer: Transformer-based representation for sketched structure. In *CVPR*, pages 14153–14162, 2020. 3

[28] Marco Rudolph, Bastian Wandt, and Bodo Rosenhahn. Same same but differnet: Semi-supervised defect detection with normalizing flows. In *WACV*, Jan. 2021. 1, 2, 3, 6, 7

[29] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *ICML*, pages 4393–4402, 2018. 2

[30] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *IPMI*, pages 146–157. Springer, 2017. 2

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 3, 6

[32] Shashanka Venkataramanan, Kuan-Chuan Peng, Rajat Vikram Singh, and Abhijit Mahalanobis. Attention guided anomaly localization in images. In *ECCV*, pages 485–503, 2020. 2, 3

[33] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 4, 5

[34] Fuzhi Yang, Huan Yang, Jianlong Fu, Hongtao Lu, and Baining Guo. Learning texture transformer network for image super-resolution. In *CVPR*, pages 5791–5800, 2020. 3

[35] Vitjan Zavrtanik, Matej Kristan, and Danijel Skčaj. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, page 107706, 2020. 1, 2, 7

[36] Shuangfei Zhai, Yu Cheng, Weining Lu, and Zhongfei Zhang. Deep structured energy based models for anomaly detection. In *ICML*, pages 1100–1109, 2016. 2, 7

[37] Jing Zhang, Tong Zhang, Yuchao Dai, Mehrtash Harandi, and Richard Hartley. Deep unsupervised saliency detection: A multiple noisy labeling perspective. In *CVPR*, pages 9029–9038, 2018. 3

[38] Ting Zhao and Xiangqian Wu. Pyramid feature attention network for saliency detection. In *CVPR*, pages 3085–3094, 2019. 3

[39] 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. In *ECCV*, volume 12365, pages 360–377. Springer, 2020. 1, 3