Dual Memory Networks Guided Reverse Distillation for Unsupervised Anomaly Detection

Chi Dai Tran[0000−0001−6345−8658], Long Hoang Pham[0000−0002−3240−657X], Duong Nguyen-Ngoc Tran[0000−0001−7537−6377], Quoc Pham-Nam Ho[0009−0006−7256−4798], and Jae Wook Jeon[0000−0003−0037−112X]

Department of Electrical and Computer Engineering,
Sungkyunkwan University, Suwon, South Korea
{tdc2000, phlong, duongtran, hpnquoc, jwjeon}@skku.edu

Abstract. Visual anomaly detection, which is essential for industrial applications, is typically framed as a one-class classification assignment. Recent techniques employing the teacher-student framework for this task have proven effective in both accuracy and processing time. However, they often assume that real-world anomalies are uncommon, emphasizing anomaly-free data while neglecting the importance of aberrant data. We contend that such a paradigm is suboptimal as it fails to differentiate between regular and irregular situations adequately. To overcome this issue, we proposed a novel **D**ual **M**emory **G**uided **R**everse **D**istillation (DM-GRD) framework to learn feature representations for both standard and abnormal data. Specifically, to obtain anomalous patterns, original images are first augmented with a simple Fourier transformation followed by Perlin noise. A teacher network then randomly receives arbitrary images to extract high-level features. To combat "forgetting" and "over-generalization" difficulties in a student network, two memory banks are introduced to independently store typical and atypical features while maximizing the distance margins between them. Next, a multi-scale feature fusion module is trained to integrate valuable information from the memory banks. Finally, a student network ingests this data to match the instructor network for the same images. Experiments on three industrial benchmark datasets reveal that DM-GRD outperforms current state-of-the-art memory bank and knowledge distillation alternatives, showcasing the robust generalization capability of the proposed framework. The code is publicly available at https://github.com/SKKUAutoLab/DM-GRD.

Keywords: Anomaly detection · Knowledge distillation · Memory bank.

1 Introduction

Anomaly detection (AD) is a foundation task in computer vision, discovering suspicious defects that could imply errors. This work is vital for several applications, including medical image analysis [38,11,44], quality control in manufacturing [2,45,24], and video surveillance [33,35,4]. Due to the paucity of unusual samples in the actual world, gathering enough anomalous prototypes for

training is time-demanding. As a result, unsupervised AD methods have grown popular in industrial contexts. They exclusively utilize normal data for training, diminishing the need for an extensive collection of atypical examples.

An alternate strategy makes use of knowledge distillation (KD), which has lately shown promise in AD [43,13,7,36,31]. In this setup, a pre-trained teacher network, typically trained on a large-scale dataset such as ImageNet [8], is used alongside a student network with a similar design that is trained purely on high-quality images to replicate the teacher's behavior. The underlying concept is that transferring knowledge from the teacher to the student network enables the student to learn diverse regular features, facilitating the detection of abnormal patterns not seen during training. Consequently, errors at both image and pixel levels can be recognized during testing. Nonetheless, three factors undermine the reliability of this hypothesis. First, the student network may become overly generalized due to the same architecture, producing features that resemble those created by the teacher [27]. Second, the identical input data may induce the student to falsely detect minor anomalies, such as dust particles in the testing dataset [13]. Last, summing or multiplying the output layers of the student network for feature aggregation is suboptimal [43]. To address these challenges, a new KD design, termed reverse distillation (RD) [36], was proposed. In RD, the instructor and the pupil take on the roles of encoder and decoder, respectively. The learner receives representative features from the teacher as input and is trained to reconstruct them at various scales.

For the KD and RD paradigms, we found three current constraints. First, with a small architecture, the student is susceptible to the "forgetting" issue [13]. Second, owing to the rarity of anomalous samples, the student network easily misidentifies anomalies because there is no consensus on what an anomaly is. We also argue that earlier methods [13,39,16] merely store typical prototypes on a memory module straining to process complex samples. Ultimately, providing a compact representation capable of embracing all normal circumstances to the learner is tough if we only lean on distillation tasks [36].

To solve the above shortcomings, we aim to preserve genuine normal and anomalous data, enabling precise descriptions of atypical patterns. In line with this notion, we present a novel Dual Memory Guided RD framework to overcome the drawbacks of the KD and RD approaches. The framework is made up of a teacher, a student, two memory banks, and a multi-scale feature fusion module. The student network serves as a decoder, accepting input features from the multi-scale feature fusion network, while two memory networks selectively store relevant normal and abnormal characteristics retrieved by the teacher.

We assess the efficacy of DM-GRD on three renowned benchmark datasets for AD and localization tasks. Experimental results signify that the proposed method outperforms existing state-of-the-art (SOTA) memory bank and KD solutions on image, pixel, and instance levels. Ablation studies are also undertaken to validate the durability and versatility of the proposed components. In summary, our key contributions are highlighted as follows:

– We propose a straightforward yet effective anomaly synthesis strategy to get supervision signals for free, integrating Fourier transformations to enhance the generality of anomalous patterns.
– We introduce a novel dual-branch memory module to address the "forgetting" and "over-generalization" problems in the student network. Besides, we design a self-supervised separation loss to maintain distinct feature representations within the two memory modules.
– We employ a multi-scale feature fusion network to merge multi-level features from the memory modules. Equipped with attention mechanisms used in image classification and video understanding, DM-GRD can learn both local and global representations among multi-scale feature maps.
– Through experimental results on three standard benchmark datasets, we illustrate that DM-GRD achieves state-of-the-art performance for anomaly detection and localization tasks, underscoring the generalizability of our approach across multiple domains while responding to real-time demands.

2 Related Work

2.1 Knowledge Distillation Approaches

These methods train a student network exclusively with anomaly-free data, detecting anomalies when the student network's features deviate from those of the teacher network. Paul et al. [3] uses local descriptors extracted by a teacher network as surrogate labels for an ensemble of multiple student networks, exploiting discrepancies among these students to spot anomalies. Instead of relying on the last layer's values of the student network, MKD [31] executes feature distillation at various layers from an expert network into a cloner network, utilizing distance vectors between them to pinpoint anomalous regions. To tackle the limits of the KD framework, RD [7] directly incorporates the teacher's outputs at different stages into the student's input, functioning as an autoencoder architecture and delivering outstanding outcomes compared to ordinary KD methods. These efforts focus on maximizing similarity in KD's feature representations for normal inputs, whereas our work attempts to distill both normal and anomalous inputs, soaring the model's endurance in handling challenging abnormal samples.

2.2 Memory Bank Approaches

To encode diverse normal patterns from training data, the memory module [12] has been implemented into deep neural networks to model the normality distribution. To address the "forgetting" issue in student networks, MemKD [13] presents a normality recall memory module (NR Memory) that boosts the student's generated regular features by recalling stored normal information. An exemplar set is then constructed for the NR Memory, allowing the recall of prior knowledge from the query feature. PatchCore [29] builds a memory bank to store nominal patch features and measures patch-level distances between test targets

and standard features. A greedy coreset subsampling strategy [1] is given to eliminate redundancy in extracted patch features. To avoid the "over-generalization" problem, TrustMAE [34] engages a memory-augmented auto-encoder coupled with a sparse memory addressing mechanism. A trust-region memory updating scheme is proposed to keep memory slots noise-free by updating memory items inside a predefined trust region. Unlike approaches that store only regular features in a memory module, our solution divides normal and abnormal features into two memory banks, enabling better separation between them.

2.3 Supervised Approaches

Since unsupervised methods rely heavily on the quality of normal images, they often fail on noisy datasets. An emerging trend centers on supervised methods, leveraging known anomalies for training. PRNet [42] learns a residual representation of multi-scale prototypes alongside a multi-size self-attention mechanism. Various anomaly generation algorithms are given by considering both seen and unseen appearances to diversify anomaly patterns. To work with only a few known anomalies, BGAD [40] proposes a boundary-guided semi-push-pull contrastive learning mechanism, presuming that a compact dividing boundary can maintain normal features within a margin region. DifferNet [30] uses normalizing flows to estimate the density of extracted features in low-dimensional data distributions. Furthermore, a multi-scale feature extractor is employed to assign meaningful likelihoods to the images. However, these methods struggle with imbalanced datasets, particularly with hidden anomalies, leading to poor performance. Likewise, subtle anomaly regions are usually overshadowed by normal ones, complicating the localization of anomalies of varied sizes.

3 Preliminaries

3.1 Problem Formulation

Assume there exists a training set $\mathcal{S}_{train}$, containing only anomaly-free samples, and a testing set $\mathcal{S}_{test}$, which includes both normal and abnormal samples from the same object class. Our objective is to fit a model on S_{train} that can detect and segment anomalies in S_{test}. In the next section, we briefly depict the RD method [7], which acts as one of the core components of our framework.

3.2 Reverse Distillation for Anomaly Detection

The RD framework consists of three modules, a fixed teacher network E, a trainable one-class bottleneck embedding (OCBE) module, and a student network D. The OCBE module is positioned after the 4^{th} block of ResNet [14] within E. This module projects high-dimensional representations into a low-dimensional space, preventing unusual traits from spreading to D. Unlike conventional KD schemes, D's reversible design reduces the impact of abnormalities, while its symmetrical shape assembles an architectural similarity between E and D.

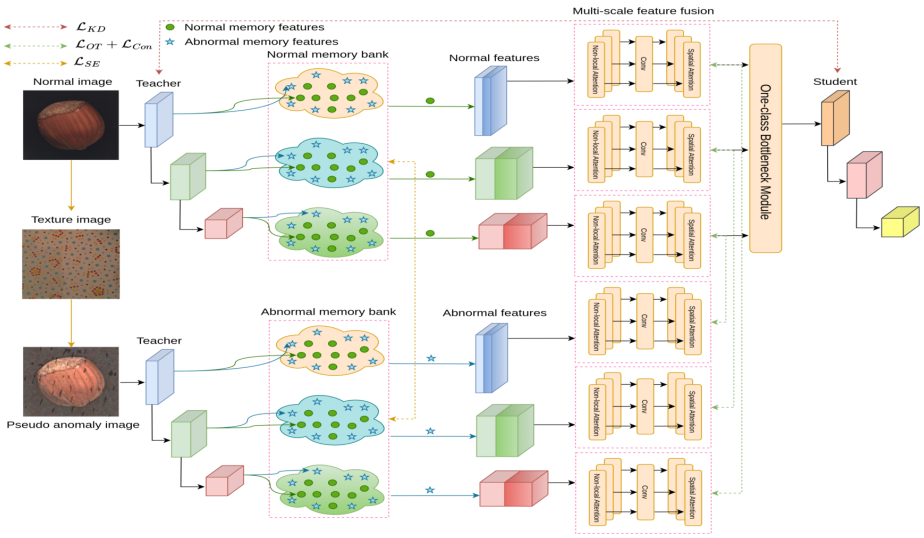

Fig. 1: An overview of DM-GRD. First, pseudo anomaly images are generated by a simple augmentation strategy. The instructor model's output features are then moved to two memory modules to select the most similar features. Next, these features are passed to the MSFF module. While the abnormal multi-scale features are used to optimize $\mathcal{L}_{OT}+\mathcal{L}_{Con}$, the regular ones are sent to the student model and minimized with the teacher model through $\mathcal{L}_{KD}$. Simultaneously, the regular and abnormal memory properties are updated by $\mathcal{L}_{SE}$.

Formally, let ϕ be the output of the OCBE block. Given an input image $I \in \mathbb{R}^{C\times H\times W}$ (where C, H, and W denote the channel, height, and width of I). The teacher E pulls features for I from multiple layers, denoted as $\{F_E^i = E^K(I)\}$, where E^i defines the i^{th} block in E. The student D is then exploited to rebuild these features, denoted as $\{F_D^i = D^K(\phi)\}$, where D^i defines the i^{th} block in D. To optimize D, a cosine similarity loss between F_E and F_D is defined as follows:

$$\mathcal{L}_{\text{KD}} = 1 - \sum_{i=1}^{K}\{\frac{(F_E^i(h,w))^\top \cdot (F_D^i(h,w))}{\|F_E^i(h,w)\|_2 \|F_D^i(h,w)\|_2}\} \tag{1}$$

where K refers to the number of feature layers for training, h and w are the height and width of the i^{th} feature map, and $\|\cdot\|_2$ represents the l_2 norm.

4 Proposed Method

The proposed framework is made up of four major components, a pre-trained teacher network E, a learnable student network D, two memory banks, and a multi-scale feature fusion (MSFF) network, which are trained in an end-to-end fashion. The overall architecture of DM-GRD is visualized in Fig 1.

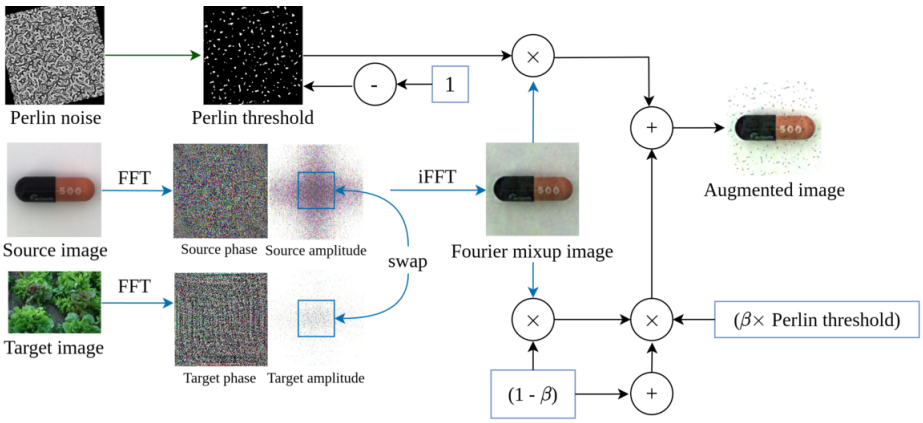

Fig. 2: The generation process of simulated anomaly images. Best viewed in color.

4.1 Anomaly Simulation Strategy

Our fundamental premise is that the RD paradigm is effective during inference if the learner can learn from the teacher to recognize both normal and aberrant characteristics. Hence, fake anomalies need to be simulated during training. The construction of the proposed anomaly simulator is depicted in Fig 2.

A noise image is generated by a Perlin noise generator [26,41] and binarized by a threshold λ to produce a Perlin threshold map $M_{pt} \in [0, 1]$. Inspired by the semantic-preserving property of the Fourier phase component, we devise a data augmentation technique to enhance the generalization ability of simulated anomaly images across several domains. The mixed amplitude of a source image I_{source} and a target image I_{target} is formulated as follows:

$$\mathcal{A}(I_{source \to target}) = (1 - \gamma)\mathcal{A}(I_{source}) + \gamma\mathcal{A}(I_{target}), \tag{2}$$

where $\gamma \sim \mathcal{N}(0, 1)$ is a hyperparameter to control the strength of the augmentation, $\mathcal{A}(I_{source}), \mathcal{A}(I_{target}) \in [3, H_1 : H_2, W_1 : W_2]$ with H_1, H_2, W_1, and W_2 are low-frequency spectrum regions and represented as:

$$\begin{aligned} H_i &= \mathbb{1}\left(\lfloor\frac{H}{2}\rfloor - \lfloor\min(h, w) \times 0.1\rfloor\right), \\ W_i &= \mathbb{1}\left(\lfloor\frac{W}{2}\rfloor - \lfloor\min(h, w) \times 0.1\rfloor\right), \end{aligned} \tag{3}$$

where $i \in \{1, 2\}$. If $i = 2$, $H_2 = H_2 + 1$ and $W_2 = W_2 + 1$.

The mixed amplitude is then blended with the original phase spectrum to form a new Fourier representation:

$$\mathcal{F}(u, v) = \mathcal{A}(I_{source \to target}) \times e^{-j \times \mathcal{P}(I_{source})(u,v)}, \tag{4}$$

which is then fed to the inverse Fourier transformation to generate a Fourier mixup image F_M. Finally, an augmented image I_a is defined as follows:

$$I_a = (1 - M_{pt}) \odot F_M + (1 - \beta) \odot F_M + (\beta \odot M_{pt}) \odot F_M, \tag{5}$$

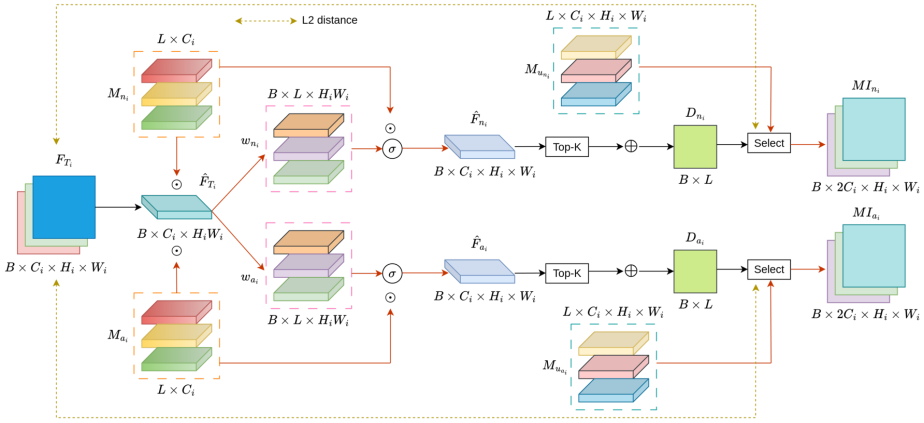

Fig. 3: The dual-branch memory network. $\oplus$ is the concatenation operation.

where $\odot$ is the element-wise multiplication operation and $\beta \in [0.1, 1.0]$ is a random hyperparameter to generate diverse anomalous images.

4.2 Dual Memory Networks

Previous approaches merely required a memory module to hold typical prototypes. We find that a single unit is insufficient to distinguish hard samples. Thus, two updatable normal and abnormal memory modules are provided to store corresponding templates $\mathbf{M} \in \mathbb{R}^{L \times C_i}$ and $\mathbf{M}_u \in \mathbb{R}^{L \times C_i \times H_i \times W_i}$, where $\mathbf{M}$ is memory features, $\mathbf{M}_u$ is updatable memory features, and L is the number of memory samples. Fig 3 illustrates an overview of our memory network.

Initially, we select a small number of normal and aberrant images from S_{train} and utilize a pre-trained teacher network to extract high-level features. Concurrently, $\mathbf{M}_u$ is acquired in the same manner. Afterward, we randomly sample L normal and abnormal images from S_{train} and feed them into the teacher network to obtain representative features from blocks 1, 2, and 3, respectively.

Given a query target $F_{T_i} \in \mathbb{R}^{B \times C_i \times H_i \times W_i}$, where B is the batch size, it is flattened to $\hat{F}_{T_i} \in \mathbb{R}^{B \times C_i \times H_i W_i}$. The cosine similarity between $\mathbf{M}_i$ and F_{T_i} is computed to yield the similarity vector $w_i \in \mathbb{R}^{B \times L \times H_i W_i}$, which affects the information retention in the foremost areas. Later, $\mathbf{M}_i$ is aggregated by w_i to get the normalized feature $\overline{F}_i \in \mathbb{R}^{B \times H_i W_i \times C_i}$ before reshaping to $\tilde{F}_i \in \mathbb{R}^{B \times C_i \times H_i \times W_i}$.

In the next stage, we apply topk selection along the channel dimension to determine the most similar snippets to the query target, which are further concatenated to form L difference information $D_i \in \mathbb{R}^{B \times L}$. Thereafter, we take the minimum indices of $\mathbf{M}_{u_i}$ to gain selected features $SF_{u_i} \in \mathbb{R}^{B \times C_i \times H_i \times W_i}$. Then, $L2$ distance is calculated to derive the best difference information:

$$\begin{aligned} &DI = d_{L2}(SF_{u_i}, F_{T_i}), \\ &\text{s.t.} \quad SF_{u_i} = \operatorname{argmin} D_i. \end{aligned} \tag{6}$$

To proceed, DI joins with D_i along the channel dimension to attain the concatenated memory information $MI \in \mathbb{R}^{B\times 2C_i\times H_i\times W_i}$ for normal and abnormal targets. Eventually, MI is moved to the MSFF module for feature aggregation.

4.3 Multi-scale Feature Fusion Network

Since MI may cause overabundance, it slows down the inference speed. Inspired by the recent success of the MSFF network in AD [42,39,47], our intuitive idea is to comprehensively exchange visual information at each scale while capturing both local and global features from the memory information.

To begin with, MI_i is fused by three MSFF blocks to halve the number of channels. In each block, MI_i is forwarded to a 3×3 convolution that preserves the number of channels. In the meantime, non-local attention [37] captures the global interaction between channels of MI_i. For the memory information features weighted by the non-local attention, the channels are reduced by half via two 3×3 convolutions to form the multi-scale representation $f_i \in \mathbb{R}^{B\times C_i\times H_i\times W_i}$. Next, spatial attention maps $M_{s_1}, M_{s_2}, M_{s_3}$ are computed as follows:

$$\begin{aligned} M_{s_1} &= \sigma(f^{7\times 7}([f_{1_{\text{avg}}}; f_{1_{\text{max}}}])), \\ M_{s_2} &= \sigma(f^{7\times 7}([f_{2_{\text{avg}}}; f_{2_{\text{max}}}])) \odot Upsample(M_{s_1}), \\ M_{s_3} &= \sigma(f^{7\times 7}([f_{3_{\text{avg}}}; f_{3_{\text{max}}}])) \odot Upsample(M_{s_2}), \end{aligned} \tag{7}$$

where σ denotes the sigmoid function and $f^{7\times 7}$ represents a 7×7 convolution.

Finally, the spatial attention maps are weighted by the multi-scale representations to obtain fused features f_{out_i}, that are supplied into the student network.

4.4 Training Objectives

We propose a unified loss for training DM-GRD, including four components, an optimal transport loss $\mathcal{L}_{OT}$, a contrastive loss $\mathcal{L}_{Con}$, a knowledge distillation loss $\mathcal{L}_{KD}$, and a separation loss L_{SE}. The total loss is defined as follows:

$$\mathcal{L} = \mathcal{L}_{KD} + \alpha * (\mathcal{L}_{OT} + \mu * \mathcal{L}_{Con}) + \tau * L_{SE}, \tag{8}$$

where α, μ, and τ are regularization parameters.

Optimal transport loss. Similar to [32,36], we employ the unbalanced Sinkhorn divergence to enforce proximity among standard samples within the normal memory bank. The optimal transport loss is defined as:

$$\mathcal{L}_{OT} = \frac{1}{m}\frac{1}{K}\sum_{i,j=1}^{m}\sum_{k=1}^{K} S_{\varepsilon,\rho}(\sigma(f_{out_k}(MI_{i,k})), \sigma(f_{out_k}(MI_{j,k}))), \tag{9}$$

where $S_{\varepsilon,\rho}(\cdot,\cdot)$ denotes the unbalanced Sinkhorn divergence.

Contrast loss. We raise the discrimination between normal and abnormal features by enforcing their separation in MSFF. The contrast loss is defined as:

$$\mathcal{L}_{Con} = \frac{1}{k}\sum_{k=1}^{K} \max(0, \cos(f_{out_k}(MI_{i,k}), f_{out_k}(\overline{MI}_{i,k})) - \theta), \tag{10}$$

where θ is a margin and $\overline{MI}$ is the concatenated abnormal memory information.

Separation loss. With the updatable memory features $M^n_{u_k}$ and $M^a_{u_k}$ of normal and abnormal targets, they are split by using the triplet loss as follows:

$$\begin{aligned} \mathcal{L}_{SE} &= \sum_{k=1}^{K}[\|S^a_k - S^p_k\|_2^2 - \|S^a_k - S^n_k\|_2^2 + \eta], \\ S^a_k &= \|\text{topk}(M^n_{u_k}; D^n_k)\|_2, \\ S^p_k &= \|\text{topk}(M^a_{u_k}; D^a_k)\|_2, \\ S^n_k &= \|\text{topk}(M^a_{u_k}; D^a_k)\|_2, \end{aligned} \tag{11}$$

where $\eta = 1$, S^a_k is the anomaly part of a normal image, S^p_k means the anomaly part of an anomaly image, and S^n_k signifies the normal part of a normal image.

5 Experimental Results and Analysis

5.1 Datasets

MVTec [2] is a well-known AD dataset, comprising ten object classes and five textures, totaling more than 5,000 images. The test data provide both image- and pixel-level annotations for calculating anomaly localization metrics. Anomalies in this dataset frequently manifest in irregular shapes.

BTAD [23] is a small dataset that consists of three categories with over 2,500 images. Similar to MVTec, the test data include both normal and abnormal samples, with anomalies predominantly found in body and surface defects.

VisA [46] stands out as the largest industrial AD dataset, composed of 10,821 images with 9,621 normal and 1,200 anomalous samples distributed over 12 objects. Anomalous images exhibit various imperfections, encompassing surface and structural flaws such as scratches, dents, and missing parts.

5.2 Implementation Details

Experimental settings. DM-GRD adopts the RD framework and utilizes WResNet50 [14] as its backbone. Input images are shrunk to 256 $\times$ 256 without applying data augmentation techniques for fair comparison. The training process spans 200 epochs, with a batch size of 16. We use the Adam optimizer [17] with a learning rate of 0.005. To attenuate overfitting, Cosine Annealing [21] is employed with a minimum learning rate of 0.0001, a warmup ratio of 0.1, and 20 warmup steps. For the unified loss, the hyperparameters α, μ, and τ are set to 0.2, 0.1, and 0.1, respectively. In the memory module, L is chosen as 30 samples from S_{train}. These configurations were determined through a grid search.

Evaluation metrics. Following prior studies [36,3,7,43], we evaluate AD and localization scores at the image- and pixel-levels using the area under the receiver operator curve, i.e., I-AUC and P-AUC. Likewise, the area under the per-region-overlap (PRO) curve estimates the instance-level AD performance.

Fig. 4: Simulated anomaly images with $\beta \in [0.1, 0.2, 0.3, 0.4, 0.5]$.

Table 1: Anomaly detection and localization performance in terms of I-AUC and P-AUC on the MVTec dataset. The best results are bolded, and the second-best are underlined. More results can be found in the supplementary material.

	Memory Bank		Supervised			Knowledge Distillation		
Category	PatchCore [29]	CFA [18]	DevNet [25]	DRA [10]	PRNet [42]	RD [7]	RD++ [36]	**DM-GRD**
Carpet	98.70/99.00	97.30/99.28	88.20/96.98	92.50/98.20	99.70/99.00	98.90/98.90	**100.0**/99.20	**100.0**/**99.31**
Grid	98.20/98.70	99.20/98.12	96.56/96.24	98.60/86.00	99.40/98.40	**100.0**/99.30	**100.0**/99.30	**100.0**/**99.32**
Leather	**100.0**/99.30	**100.0**/99.37	96.23/98.89	98.90/93.80	**100.0**/**99.70**	**100.0**/99.40	**100.0**/99.40	**100.0**/99.62
Tile	98.70/95.40	99.40/95.25	95.99/89.39	**100.0**/92.30	**100.0**/**99.60**	99.30/95.60	99.70/96.60	99.70/97.13
Wood	99.20/95.00	99.70/91.53	99.26/89.66	99.10/82.90	**100.0**/**97.80**	99.20/95.30	99.30/95.80	99.91/96.68
Bottle	**100.0**/98.60	**100.0**/98.84	99.53/96.02	**100.0**/91.30	**100.0**/**99.40**	**100.0**/98.70	**100.0**/98.80	**100.0**/98.84
Cable	99.50/98.40	**99.80**/**98.97**	92.39/92.86	94.20/86.60	98.90/98.80	95.00/97.40	99.20/98.40	99.68/98.15
Capsule	98.10/98.80	97.30/**99.11**	81.55/93.16	95.10/89.30	98.00/98.50	96.30/98.70	**99.00**/98.80	97.30/98.85
Hazelnut	**100.0**/98.70	**100.0**/98.85	100.0/95.27	**100.0**/89.60	**100.0**/**99.70**	99.90/98.90	**100.0**/99.20	**100.0**/99.30
Metal nut	**100.0**/98.40	**100.0**/99.15	99.89/91.68	99.10/79.50	**100.0**/**99.70**	**100.0**/97.30	**100.0**/98.10	**100.0**/97.14
Pill	96.60/97.40	97.90/98.93	82.77/85.25	88.30/84.50	**99.30**/**99.50**	96.60/98.20	98.40/98.30	97.46/98.51
Screw	98.10/99.40	97.30/98.91	95.99/63.04	**99.50**/54.00	95.90/97.50	97.00/99.60	98.90/**99.70**	98.54/99.50
Toothbrush	**100.0**/98.70	**100.0**/98.96	93.33/84.72	87.50/75.50	**100.0**/**99.60**	99.50/99.10	**100.0**/99.10	**100.0**/99.40
Transistor	**100.0**/96.30	**100.0**/98.06	84.00/83.31	88.30/79.10	99.70/**98.40**	96.70/92.50	98.50/94.30	99.96/96.30
Zipper	99.40/98.80	99.60/**99.02**	99.91/98.89	99.70/96.90	99.70/98.80	98.50/98.20	98.60/98.80	**99.89**/98.03
Average	99.10/98.06	99.17/98.15	93.71/90.36	96.10/85.30	99.40/**99.00**	98.46/97.81	99.44/98.25	**99.50**/98.41

5.3 Main Results

Anomaly detection and localization. Table 1 and Table 2 compare the AD and localization outcomes of DM-GRD to recent SOTA models on MVTec AD. Our method improves upon the baseline RD, promoting the average I-AUC and P-AUC by up to 1.04% and 0.6%, and PRO by a substantial margin of 4.67%. DM-GRD also outperforms RD++ by 0.06%, 0.16%, and 3.61% in I-AUC, P-AUC, and PRO, respectively. Although RD and RD++ yield competitive results, their reliance on the KD task hinders their ability to classify complex samples such as screws, cables, and transistors. By leveraging two memory modules for normal and abnormal features, DM-GRD surpasses its memory bank counterparts, e.g., PatchCore and CFA, by 0.33% to 5.2% on all metrics. Notably, without the mechanism to retrieve normal information from the memory bank, RD and RD++ perform worse than PatchCore and CFA on PRO.

To test the generalization of DM-GRD, we also evaluate it on two other datasets, namely BTAD and VisA. The quantitative results are presented in Tab 3 and Tab 4. DM-GRD surpasses its KD competitors, such as RD and RD++, on the BTAD dataset by a sizable margin ranging from 0.54% to 14.61%. On the VisA dataset, DM-GRD gets an I-AUC of 96.10%, beating RD by 0.1% and RD++ by 0.2%. Regarding PRO, our method exceeds RD++ by 2.97% and RD by a significant gap of 25.47%. Aside from PatchCore, both SPADE and PaDiM exhibit inferior results compared to RD and RD++. These findings con-

Table 2: Anomaly localization results of PRO on the MVTec dataset.

	Memory Bank		Supervised			Knowledge Distillation			
Category	PatchCore [29]	CFA [18]	DevNet [25]	DRA [10]	PRNet [42]	RD [7]	RD++ [36]	MKD [31]	**DM-GRD**
Carpet	96.60	96.54	85.80	92.20	97.00	97.00	97.70	92.50	**99.22**
Grid	96.00	94.04	79.80	71.50	95.90	97.60	97.70	72.90	**99.29**
Leather	98.90	97.43	88.50	84.00	99.20	99.10	99.20	97.50	**99.64**
Tile	87.30	89.26	78.90	81.50	**98.20**	90.60	92.40	74.30	97.81
Wood	89.40	90.54	75.40	69.70	95.90	90.90	93.30	76.50	**97.98**
Bottle	96.20	95.76	83.50	77.6	97.00	96.60	97.00	88.60	**98.97**
Cable	92.50	94.17	80.90	77.70	97.20	91.00	93.90	66.20	**98.04**
Capsule	95.50	93.66	83.60	79.10	92.50	95.80	96.40	90.10	**98.49**
Hazelnut	93.80	95.75	83.60	79.10	92.50	95.50	96.30	94.30	**99.11**
Metal nut	91.40	94.54	76.90	76.70	95.80	92.30	93.00	76.90	**97.98**
Pill	93.20	97.19	69.20	77.00	97.20	96.4	97.00	86.40	**98.84**
Screw	97.90	95.23	31.10	30.10	92.40	98.20	98.60	85.20	**99.49**
Toothbrush	91.50	91.14	33.50	56.10	95.60	94.50	94.20	87.30	**98.45**
Transistor	83.70	95.35	39.10	49.00	94.80	78.0	81.80	68.10	**97.56**
Zipper	97.10	95.95	81.30	91.00	95.50	95.40	96.30	86.50	**98.11**
Average	93.40	94.44	71.40	73.30	96.10	93.93	94.99	82.90	**98.60**

Table 3: Anomaly localization results on the BTAD dataset at P-AUC/PRO.

	Memory Bank			Supervised		Knowledge Distillation		
Class	PatchCore [29]	CFA [18]	REB [22]	BGAD [40]	PRNet [42]	RD [7]	RD++ [36]	**DM-GRD**
01	97.03/64.92	95.90/72.00	94.70/-	**98.20**/83.00	96.60/81.40	96.60/75.30	96.20/73.20	96.73/**91.26**
02	95.83/47.27	96.00/53.20	95.60/-	**97.90**/64.80	95.10/54.40	96.70/68.20	96.40/71.30	97.18/**84.90**
03	99.19/67.72	98.60/94.10	99.70/-	99.80/99.30	99.60/98.30	99.70/87.80	99.70/87.40	**100.0**/**99.44**
Average	97.35/59.97	96.83/73.10	97.20/-	**98.60**/82.40	97.10/78.00	97.67/77.10	97.43/77.30	97.97/**91.71**

solidate our prediction that depending on a single memory module may degrade model performance when applied to challenging datasets.

Complexity analysis. To verify the method's feasibility for real-time industrial applications, we measure DM-GRD with other SOTA models from the perspective of the total number of parameters, inference time, and training time. The statistics are detailed in Table 5. Compared to memory bank approaches, they consume more memory than DM-GRD, resulting in a runtime appropriate for offline applications. Although our WResNet50 backbone version covers more parameters than RD++, it maintains strong performance at a lower latency. Remarkably, in the condensed variant, e.g., ResNet18, we defeat RD less than three milliseconds and compete with RD++ in terms of P-AUC and PRO.

Discussion. In this study, we focus on ameliorating the efficiency and shortening the processing time of both memory bank and KD methods. However, much of the computational overhead arises from adopting an extensive backbone. How to apply a smaller backbone, e.g., ResNet18, while maintaining high performance is an intriguing question that we leave for future research.

5.4 Ablation Study

Study on training objectives. Table 6 summarizes DM-GRD's performance on several loss combinations. In RD++, training the projection layers

Table 4: Anomaly detection results on the VisA dataset at I-AUC/PRO.

	Memory Bank			Knowledge Distillation		
Category	PatchCore [29]	SPADE [5]	PaDiM [6]	RD [7]	RD++ [36]	**DM-GRD**
Candle	**98.60**/94.00	91.00/93.20	91.60/**95.70**	92.20/92.20	96.40/93.80	95.00/94.91
Capsules	81.60/85.50	61.40/36.10	70.70/76.90	90.10/56.90	92.10/95.80	**92.47**/**98.43**
Cashew	97.30/**94.50**	97.80/57.40	93.00/87.90	**99.60**/79.00	97.80/91.20	95.36/92.40
Chewing gum	99.10/84.60	85.80/93.90	98.80/83.50	**99.70**/92.50	96.40/88.10	96.70/**99.60**
Fryum	96.20/85.30	88.60/91.30	88.60/80.20	96.60/81.00	95.80/90.00	**97.16**/**95.31**
Macaroni1	97.50/95.40	95.20/61.30	87.00/92.10	98.40/71.30	94.00/**96.90**	**98.50**/95.74
Macaroni2	78.10/94.40	87.90/63.40	70.50/75.40	**97.60**/68.00	88.00/97.70	**97.60**/**98.43**
PCB1	**98.50**/94.30	72.10/38.40	94.70/91.30	97.60/43.20	97.00/95.80	93.87/**96.97**
PCB2	**97.30**/89.20	50.70/42.20	88.50/88.70	91.10/46.40	97.20/90.60	92.99/**91.37**
PCB3	**97.90**/90.90	90.50/80.30	91.00/84.90	95.50/80.30	96.80/93.10	96.44/**97.42**
PCB4	99.60/90.10	83.10/71.60	97.50/81.60	96.50/72.20	**99.80**/91.90	99.34/**97.55**
Pipe fryum	**99.80**/95.70	81.10/61.70	97.00/92.50	97.00/68.30	99.60/95.60	97.80/**98.26**
Average	95.10/91.20	82.10/65.90	89.10/85.90	96.00/70.90	95.90/93.40	**96.10**/**96.37**

Table 5: Complexity comparison between memory bank and KD models on the MVTec dataset. The test conditions were conducted on Intel Core i5-12600K and NVIDIA TITAN X. [DM-GRD[1]]: with ResNet18, [DM-GRD[2]]: with WResNet50.

Type	Method	Params(M)	Latency(ms)	Training Time(h)	I-AUC	P-AUC	PRO
Memory Bank	SPADE	68.9	1417.7	0.03	85.40	95.50	89.50
	PaDiM	68.9	19567.9	**0.02**	90.80	96.60	91.30
	PatchCore	68.9	23.8	**0.02**	99.10	98.06	93.40
	CFA	66.8	54.8	0.63	99.17	98.15	94.44
KD	RD	161.1	9.4	2.9	98.46	97.81	93.93
	RD++	176.6	12.9	5.4	99.44	98.25	94.99
	DM-GRD[1]	**34.1**	**6.4**	2.0	98.61	98.21	98.20
	DM-GRD[2]	264.7	11.9	7.7	**99.50**	**98.41**	**98.60**

with $\mathcal{L}_{OT}$ leads to more effective condensation of regular features. Thus, we can notice a moderate gain compared to RD, especially in PRO. In the case of DM-GRD, by reinforcing the contrast between normal and abnormal features in two memory modules with $\mathcal{L}_{SE}$, it outplays RD without requiring $\mathcal{L}_{OT}$. Likewise, the condensed features derived by $\mathcal{L}_{OT}$ and $\mathcal{L}_{Con}$ prevent abnormal signals from impacting the student network, thereby boosting P-AUC and PRO criteria.

Study on the number of memory items. Table 9 examines the effects of different amounts of information expanded to the memory modules. Even though a larger L improves query performance, it can introduce extra parameters and make the model's convergence tougher. Hence, for the sake of higher P-AUC and PRO, we randomly picked L as 30 from S_{train} for most objects. For classes with fewer training samples, i.e., Toothbrush, we set it to a smaller value, e.g., 10.

Study on noise levels. Table 8 reports the influence of noise levels added to training images. For PRO, the results indicate negligible variance in varied noise intensities. Conversely, I-AUC and P-AUC tend to perform more stable

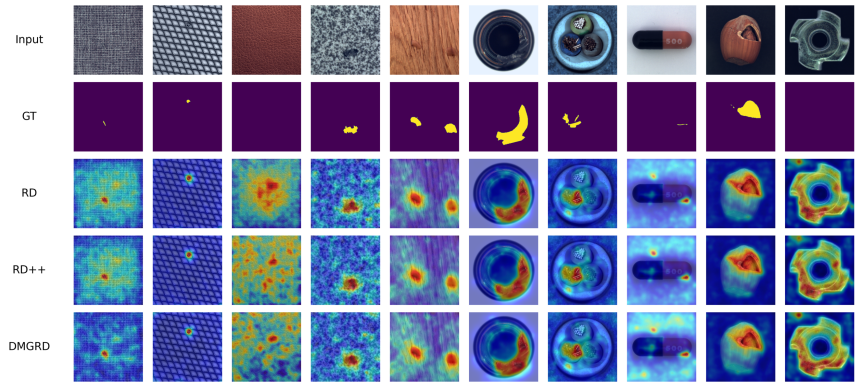

Fig. 5: Visualization of KD methods for anomaly localization in MVTec AD.

Fig. 6: An example of AD methods for generating abnormal regions.

with higher refinement. These findings suggest that random β values are more proper for generalizing in anomaly detection and localization tasks.

Study on the generalization of different backbones. Table 7 compares the performance of the proposed method to its KD equivalents on various ResNet backbones. DM-GRD consistently outperforms RD++ in all backbones, indicating that deeper networks yield better AD results. Except for I-AUC, the P-AUC and PRO criteria of both methods on ResNet50 and WResNet50 backbones are almost identical, making them suitable for real-time industrial applications.

5.5 Visualization Analysis

Anomaly localization. Fig 5 depicts the anomaly maps generated by KD models on MVTec AD. In the third column, RD incorrectly ranks the middle portion of the leather as anomalous due to a lack of atypical data during training, despite the image being normal. Similarly, in the first and ninth columns, where there is a tiny incision in the carpet and a triangular hole in the hazelnut, DM-GRD discovers anomalous regions more precisely than RD and RD++.

Types of noise. Fig 6 intuitionally visualizes the abnormal zones produced by several anomaly-generating mechanisms. Simplex noise [36] creates more natural pseudo-anomalies than Gaussian noise. However, real-world abnormalities frequently appear as damage, rendering it unsuitable for mimicking fake samples. Although CutPaste [19] and CutX [20] are simple techniques, cutting and pasting certain overlapping sections of an image can lead to artificial visuals. As opposed to our method, Perlin noise [41] generates more realistic faults but lacks the diversity of bogus images, which is useful for Out-of-Distribution (OOD) AD.

Table 6: Study on training objectives on the MVTec dataset.

Method	I-AUC	P-AUC	PRO
RD	98.46	97.81	93.93
RD++	**99.44**	98.25	94.99
Ours ($\mathcal{L}_{KD} + \mathcal{L}_{OT}$)	94.48	98.05	97.93
Ours ($\mathcal{L}_{KD} + \mathcal{L}_{Con}$)	98.19	97.93	97.88
Ours ($\mathcal{L}_{KD} + \mathcal{L}_{SE}$)	98.79	97.96	97.87
Ours ($\mathcal{L}_{KD} + \mathcal{L}_{OT} + \mathcal{L}_{Con}$)	97.40	**98.38**	**98.29**

Table 7: Study on different backbones on the MVTec dataset.

	I-AUC		P-AUC		PRO	
Backbone	RD++	**Ours**	RD++	**Ours**	RD++	**Ours**
ResNet18	**98.63**	98.40	97.64	**97.82**	93.65	**97.65**
ResNet50	**99.05**	98.61	98.17	**98.21**	94.78	**98.20**
WResNet50	99.44	**99.50**	98.25	**98.41**	94.99	**98.60**

Table 8: Study on noise levels on the MVTec dataset.

β	I-AUC	P-AUC	PRO
0.1	97.69	97.77	98.25
0.2	97.87	97.59	97.99
0.3	97.96	97.85	98.21
0.4	97.07	97.88	98.25
0.5	**98.01**	**98.08**	**98.33**

Table 9: Study on the number of memory items on MVTec AD.

L	I-AUC	P-AUC	PRO
5	97.48	97.89	97.75
10	96.27	97.85	97.81
30	**99.50**	**98.41**	**98.60**
50	96.66	97.71	97.50
100	99.11	97.95	98.25

6 Discussion and Conclusion

Conclusion. In this paper, drawing inspiration from the efficiency of the RD architecture, we design a novel model named DM-GRD for the AD task. DM-GRD introduces several key innovations, a simple approach for simulating pseudo anomalies, two memory banks to alleviate the "forgetting" and "over-generalization" problems in student networks, and a MSFF network for feature aggregation at each scale. Extensive experiments on several datasets demonstrate the generalization of DM-GRD compared to existing memory bank and KD approaches. Remarkably, our shortened version surpasses its baseline while being three milliseconds faster. We expect that the method will prove advantageous for industrial applications and contribute to further advances in the field.

Limitation. While DM-GRD shows effectiveness, it is suboptimal to simply create irregular artifacts by randomly adding noise to the images. In reality, abnormal regions often vary in size and only occur in specific areas. Thus, understanding these traits can lead to more accurate anomaly detection.

Future work. We intend to adapt DM-GRD for OOD AD, where images encompass a variety of anomalous scenarios, making generalization a crucial factor. In addition, with the recent developments in generative models [28,9,15], it is worthwhile to apply them to generate higher-quality anomaly images. Since our dual memory networks are flexible and can be integrated into various topologies, we are also excited to test them on different KD architectures.

Acknowledgement This work was supported by the Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korean government(MSIT) (No. 2021-0-01364, An intelligent system for 24/7 real-time traffic surveillance on edge devices).

References

1. Agarwal, P.K., Har-Peled, S., Varadarajan, K.R., et al.: Geometric approximation via coresets. Combinatorial and computational geometry **52**(1), 1–30 (2005)
2. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9592–9600 (2019)
3. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4183–4192 (2020)
4. Chen, Y., Liu, Z., Zhang, B., Fok, W., Qi, X., Wu, Y.C.: Mgfn: Magnitude-contrastive glance-and-focus network for weakly-supervised video anomaly detection. In: Proceedings of the AAAI conference on artificial intelligence. vol. 37, pp. 387–395 (2023)
5. Cohen, N., Hoshen, Y.: Sub-image anomaly detection with deep pyramid correspondences. arXiv preprint arXiv:2005.02357 (2020)
6. Defard, T., Setkov, A., Loesch, A., Audigier, R.: Padim: a patch distribution modeling framework for anomaly detection and localization. In: International Conference on Pattern Recognition. pp. 475–489. Springer (2021)
7. Deng, H., Li, X.: Anomaly detection via reverse distillation from one-class embedding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9737–9746 (2022)
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
9. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Advances in neural information processing systems **34**, 8780–8794 (2021)
10. Ding, C., Pang, G., Shen, C.: Catching both gray and black swans: Open-set supervised anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7388–7398 (2022)
11. Fernando, T., Gammulle, H., Denman, S., Sridharan, S., Fookes, C.: Deep learning for medical anomaly detection–a survey. ACM Computing Surveys (CSUR) **54**(7), 1–37 (2021)
12. Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., Colmenarejo, S.G., Grefenstette, E., Ramalho, T., Agapiou, J., et al.: Hybrid computing using a neural network with dynamic external memory. Nature **538**(7626), 471–476 (2016)
13. Gu, Z., Liu, L., Chen, X., Yi, R., Zhang, J., Wang, Y., Wang, C., Shu, A., Jiang, G., Ma, L.: Remembering normality: Memory-guided knowledge distillation for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16401–16409 (2023)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
15. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
16. Hou, J., Zhang, Y., Zhong, Q., Xie, D., Pu, S., Zhou, H.: Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection. In: Proceedings

of the IEEE/CVF International Conference on Computer Vision. pp. 8791–8800 (2021)
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
18. Lee, S., Lee, S., Song, B.C.: Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. IEEE Access **10**, 78446–78454 (2022)
19. Li, C.L., Sohn, K., Yoon, J., Pfister, T.: Cutpaste: Self-supervised learning for anomaly detection and localization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9664–9674 (2021)
20. Long, J., Yang, Y., Hua, L., Ou, Y.: Self-supervised augmented patches segmentation for anomaly detection. In: Proceedings of the Asian Conference on Computer Vision. pp. 1926–1941 (2022)
21. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983 (2016)
22. Lyu, S., Mo, D., keung Wong, W.: Reb: Reducing biases in representation for industrial anomaly detection. Knowledge-Based Systems p. 111563 (2024)
23. Mishra, P., Verk, R., Fornasier, D., Piciarelli, C., Foresti, G.L.: Vt-adl: A vision transformer network for image anomaly detection and localization. In: 2021 IEEE 30th International Symposium on Industrial Electronics (ISIE). pp. 01–06. IEEE (2021)
24. Napoletano, P., Piccoli, F., Schettini, R.: Semi-supervised anomaly detection for visual quality inspection. Expert Systems with Applications **183**, 115275 (2021)
25. Pang, G., Ding, C., Shen, C., Hengel, A.: Explainable deep few-shot anomaly detection with deviation networks. arxiv 2021. arXiv preprint arXiv:2108.00462
26. Perlin, K.: An image synthesizer. ACM Siggraph Computer Graphics **19**(3), 287–296 (1985)
27. Pirnay, J., Chai, K.: Inpainting transformer for anomaly detection. In: International Conference on Image Analysis and Processing. pp. 394–406. Springer (2022)
28. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
29. Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., Gehler, P.: Towards total recall in industrial anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14318–14328 (2022)
30. Rudolph, M., Wandt, B., Rosenhahn, B.: Same same but differnet: Semi-supervised defect detection with normalizing flows. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 1907–1916 (2021)
31. Salehi, M., Sadjadi, N., Baselizadeh, S., Rohban, M.H., Rabiee, H.R.: Multiresolution knowledge distillation for anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14902–14912 (2021)
32. Séjourné, T., Feydy, J., Vialard, F.X., Trouvé, A., Peyré, G.: Sinkhorn divergences for unbalanced optimal transport. arXiv preprint arXiv:1910.12958 (2019)
33. Sultani, W., Chen, C., Shah, M.: Real-world anomaly detection in surveillance videos. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6479–6488 (2018)
34. Tan, D.S., Chen, Y.C., Chen, T.P.C., Chen, W.C.: Trustmae: A noise-resilient defect classification framework using memory-augmented auto-encoders with trust regions. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 276–285 (2021)

35. Tian, Y., Pang, G., Chen, Y., Singh, R., Verjans, J.W., Carneiro, G.: Weakly-supervised video anomaly detection with robust temporal feature magnitude learning. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4975–4986 (2021)
36. Tien, T.D., Nguyen, A.T., Tran, N.H., Huy, T.D., Duong, S., Nguyen, C.D.T., Truong, S.Q.: Revisiting reverse distillation for anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 24511–24520 (2023)
37. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7794–7803 (2018)
38. Xiang, T., Zhang, Y., Lu, Y., Yuille, A.L., Zhang, C., Cai, W., Zhou, Z.: Squid: Deep feature in-painting for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23890–23901 (2023)
39. Yang, M., Wu, P., Feng, H.: Memseg: A semi-supervised method for image surface defect detection using differences and commonalities. Engineering Applications of Artificial Intelligence **119**, 105835 (2023)
40. Yao, X., Li, R., Zhang, J., Sun, J., Zhang, C.: Explicit boundary guided semi-push-pull contrastive learning for supervised anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24490–24499 (2023)
41. Zavrtanik, V., Kristan, M., Skočaj, D.: Draem-a discriminatively trained reconstruction embedding for surface anomaly detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8330–8339 (2021)
42. Zhang, H., Wu, Z., Wang, Z., Chen, Z., Jiang, Y.G.: Prototypical residual networks for anomaly detection and localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16281–16291 (2023)
43. Zhang, X., Li, S., Li, X., Huang, P., Shan, J., Chen, T.: Destseg: Segmentation guided denoising student-teacher for anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3914–3923 (2023)
44. Zhao, H., Li, Y., He, N., Ma, K., Fang, L., Li, H., Zheng, Y.: Anomaly detection for medical images using self-supervised and translation-consistent features. IEEE Transactions on Medical Imaging **40**(12), 3641–3651 (2021)
45. Zipfel, J., Verworner, F., Fischer, M., Wieland, U., Kraus, M., Zschech, P.: Anomaly detection for industrial quality assurance: A comparative evaluation of unsupervised deep learning models. Computers & Industrial Engineering **177**, 109045 (2023)
46. Zou, Y., Jeong, J., Pemula, L., Zhang, D., Dabeer, O.: Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In: European Conference on Computer Vision. pp. 392–408. Springer (2022)
47. Zuo, Z., Wu, Z., Chen, B., Zhong, X.: A reconstruction-based feature adaptation for anomaly detection with self-supervised multi-scale aggregation. In: ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 5840–5844. IEEE (2024)