Robust Industrial Anomaly Detection via Style Shift Estimation and Cascade Distillation

Shaojiang Yuan, Mengke Song, Jia Song, Xinyu Liu, Chenglizhao Chen*

***Abstract*—Knowledge distillation (KD) has demonstrated remarkable performance in industrial anomaly detection (IAD) field, but its immense potential is constrained by the trade-off between generalization of normal sample and robustness of abnormal sample. High generalization often weakens the robustness against anomalies, leading to overlooked defects, while enhancing robustness can cause normality forgetting, where normal samples are misclassified as anomalies. This dilemma stems from the existing methods commonly assuming style invariance across normal and abnormal areas, focusing solely on the abnormal content information while style information is usually neglected and put on the shelf. In this paper, our main novelty lies in leveraging anomaly style shift estimation for enlarging abnormal robustness while maintaining high normal generalization. This is the first attempt to introduce the style information into the knowledge distillation-based industrial anomaly detection methods. By employing the bidirectional fusion of style and content information, our approach achieves more comprehensive and practical anomaly detection and localization results. Thus, our method can provide a more balanced and robust solution for the future industrial anomaly detection task. Experimental results on the MVTec2D, MVTec3D, BTAD, and KolektorSDD2 datasets demonstrate that our method outperforms major state-of-the-art (SOTA) methods in both accuracy and speed, with image-level and pixel-level AUROC scores of 99.6% and 98.6% on MVTec2D, respectively. Furthermore, the efficiency of our method is validated in a real-world plastic crate defect detection system on a logistics production line.**

***Index Terms*—Industrial anomaly detection, anomaly localization, defect detection, salient object detection**

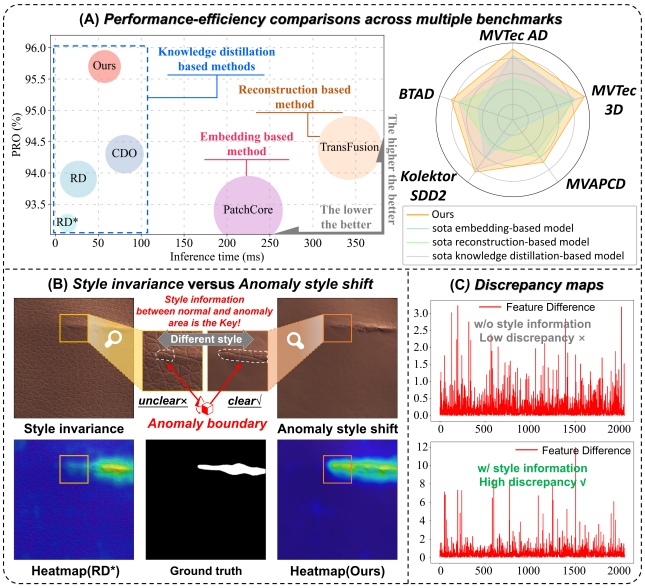


Fig. 1. Leveraging the style information, our method can conquer the low discrepancy issue of existing knowledge-distillation methods and accurately identify these easily confused anomalous pixels. (A) Comparison of PRO (per-region overlap), inference time, and training memory consumption (circle size) across various industrial anomaly detection methods on the MVTec AD benchmark [22]. Notably, RD* [23] is an optimized version of the RD [15] model. (B) Visualization of the effect of the style information shift. The qualitative results showcase the performance using style invariance (same visual style between normal and anomalous regions) versus anomaly style shift. (C) Discrepancy maps between teacher and student network comparing results with and without style information.

I. Introduction

With advancements in industrial automation, manufacturers have significantly improved the quality of goods, resulting in a notable disproportion of normal products to abnormal ones [1], [2]. Nowadays, in the initial stages of product manufacturing, it is often challenging to gather sufficient defect samples to train supervised defect detection models [3]–[5]. Therefore, unsupervised and semi-supervised industrial anomaly detection (IAD) and localization [6]–[21] are becoming increasingly crucial in the quality control of industrial products.

Unsupervised industrial anomaly detection methods isolate anomaly data by learning the distribution of normal data only. These approaches essentially involve searching the normal template feature that most similarly matches the test data and then calculating the differences to locate defects.

Among existing unsupervised industrial anomaly detection methods, knowledge distillation-based (KD) [14]–[16], [24]–[27] methods have gained significant attention due to their simple model architectures and efficient performances, as illustrated by the comparison in Fig. 1 (A). The core hypothesis of KD model is that, during the training process for a single category, the student network will only acquire the representational capability of the pre-trained teacher network for the specific category, failing to generalize to unseen data.

However, as research progresses [26]–[30], it has been discovered that the overgeneralization of student networks is uncontrollable, which reduces the interpretability of the T-S model, and this paper designs a toy case to analyze this issue. As shown in Fig. 2 (A), we directly integrate an out-of-distribution (OOD) class data into the in-distribution (ID) data to form a mixed class data. Given the large size of the OOD

The authors are with the Qingdao Institute of Software, College of Computer Science and Technology, China University of Petroleum (East China), Qingdao, P. R. China, and also with the Shandong Key Laboratory of Intelligent Oil & Gas Industrial Software. In addition, Chenglizhao Chen is also affiliated with the Key Laboratory of Forensic Science and Technology at College of Sichuan Province.

* Corresponding author: Chenglizhao Chen, cclz123@163.com.

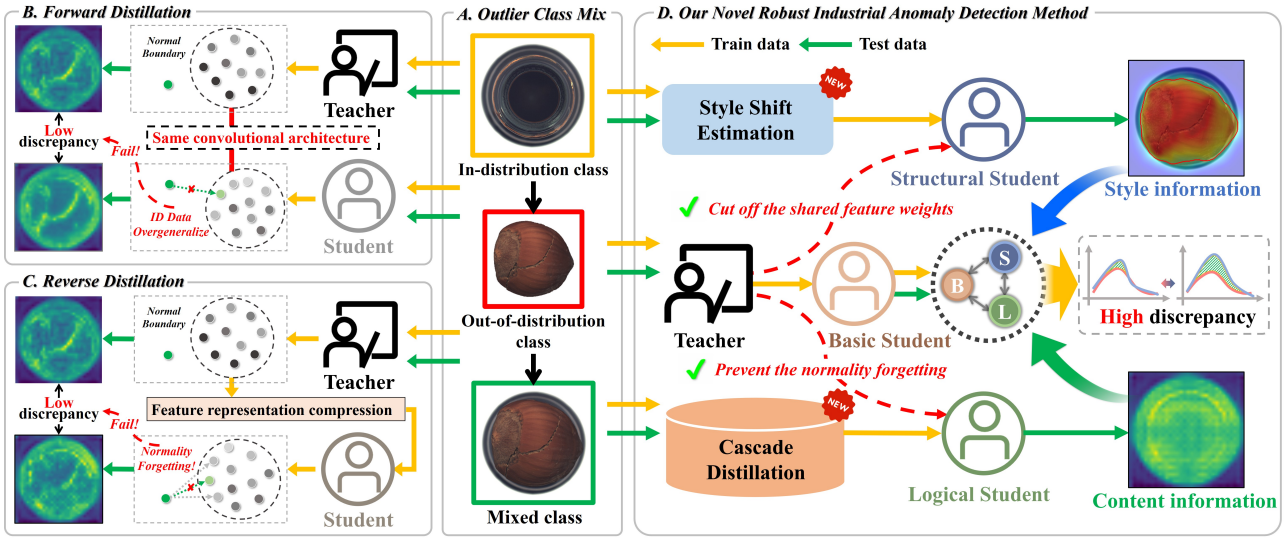


Fig. 2. The toy case and main scheme of the proposed robust industrial anomaly detection network. Toy case (A): combine the in-distribution (ID) class data with out-of-distribution (OOD) class data into mixed class data. Left top & bottom (B & C): the forward & reverse distillation methods that infer the mixed class data after training on the entire in-distribution data set. Right (D): our proposed robust industrial anomaly detection (RIAD) method that performs anomaly detection via a multi-level student learning strategy.

sample image, which nearly occupies the entire ID sample, the inference results of different models can more clearly reflect the problem faced by the KD-based IAD methods in balancing the generalization of ID data and the robustness of OOD data. As it is illustrated in the left part of Fig. 2, since the Forward T-S network [14] shares the same model architecture and input data, there arises an issue of ID data overgeneralization caused by "shared convolutional feature weights", which enables the student network to extract anomalous features from OOD data, making it difficult to directly distinguish abnormal sample from the normal class boundary. Reverse distillation [15] seems a better alternative as it alleviates the shared feature connection problem through feature representation compression and reconstruction process. However, this method similarly suffers from the overgeneralization problem, as it fails to fully utilize normal content information in the restoration process, overlooking the diversity of normal features. To address this issue, recent methods [31], [32] incorporate normality memory to enhance the robustness against OOD data. Nevertheless, the low-quality normal features searched from the memory bank can undermine the generalization of ID data, leading to "normality forgetting". In contrast to existing methods that assume style invariance and ignore the vital role of style information in IAD task, we highlight that leveraging anomaly style shift information can enable the model to better balance generalization and robustness. As illustrated in Fig. 1 (B), collaborating anomaly style shift enhances the distinction between anomalous and normal regions, resulting in more robust anomaly detection. Additionally, from a quantitative standpoint, a high discrepancy map between T-S network can be attained by using the style information, as demonstrated in Fig. 1 (C).

Thus, in this paper, we decouple the complex AD task into multi-level student learning tasks to fully utilize both style and content information. As we can see in Fig. 2 (D), for the structural-learning task, we propose a Style Shift Estimation (SSE) approach that estimates coarse defects in OOD data and transforms them into a style distinct from the ID image domain. This challenges the cross-style feature extraction capability of the student network without affecting the teacher, increasing T-S discrepancies in abnormal regions by disrupting their shared convolutional feature connection. For the logical learning task, our view is that the fundamental cause of "normality forgetting" is the close proximity between the T-S networks. The pre-trained teacher network possesses such abundant knowledge that the student network merely imitates the output distribution of the teacher and forgets normality memories. To resolve this, we draw from the idea that students debate peers more than teachers. Thus, we propose training the logical student without associating it with the teacher network. Instead, we utilize the Learnable Normal Prototype (LNP) along with our cascade distillation learning strategy, relying solely on the knowledge of a basic student to prevent memory forgetting about content information. We also design a semi-online anomaly incremental learning (SOAIL) module within the multi-level student learning scheme. The SOAIL module enables the meaningful bidirectional fusion of style and content information while continuously enhancing model deployment effectiveness through human interaction in a semi-supervised setting. The contributions of our work can be summarized as follows.

- This study proposes a novel Robust Industrial Anomaly Detection (RIAD) method, the first to consider style information, which casts the complex AD task as multi-level student learning tasks and can be applied to both unsupervised and semi-supervised scenarios.
- RIAD incorporates four key components: the Style Shift Estimation (SSE), Learnable Normal Prototype (LNP), semi-online anomaly incremental learning (SOAIL), and Cascade Distillation Learning modules. These modules

are designed to prevent the "shared feature weights" and "normality forgetting" problems, effectively balancing the generalization and robustness of T-S network.
- Extensive experiments on the MVTec AD, MVTec-3D AD, BTAD, and KolektorSDD2 benchmarks validate the superior performance of our RIAD method. Specifically, it achieves image-level and pixel-level AUROC scores of 99.6 and 98.6 on the MVTec AD dataset, respectively, surpassing primary state-of-the-art methods in accuracy and speed. Furthermore, we have validated its effectiveness across unsupervised and semi-supervised settings in a real-world industrial application.

The rest of this paper is organized as follows: in Section II, the related work of industrial image anomaly detection is introduced; the details of the RIAD method are represented in Section III and we report all the experiment results in Section IV; the Section V summarizes this paper.

II. Related work

This study surveys industrial image anomaly detection methods from unsupervised and semi-supervised perspectives, which are detailed below.

A. Unsupervised anomaly detection scenario

Unsupervised anomaly detection methods can be divided into three categories: embedding-based [6]–[8], reconstruction-Based [9]–[13], and knowledge distillation-based methods [14]–[16], [24]–[27]. ***1) Embedding-based methods*** leverage various encoder models to extract high-dimensional feature representations from test images. These features are then compared against pre-stored normal feature embeddings from training memory banks to detect anomalies. These methods often rely on metric learning techniques like nearest-neighbor searches [6] or distance metrics [7], [8] to quantify the similarity between test and normal features. Although the normal features of memory banks are diverse, they tend to consume significant computational memory space, which restricts the volume of the training set. Moreover, the necessity to search through the entire memory bank results in prolonged inference times, and the retrieved normal features may not match correctly. ***2) Reconstruction-based methods*** train models to restore normal samples by leveraging pseudo-anomalous data. During the testing phase, these models generate normal features that are compared with those of the test images to identify anomalies. Typically, these methods rely on advanced generative models such as Autoencoder [9], [10], Generative Adversarial Network [11], and Diffusion model [12], [13]. However, the effectiveness of image reconstruction is often limited by the insufficient diversity of pseudo-anomalous data, causing the model to struggle with distinguishing whether distorted features represent anomalous areas.

Compared to traditional reconstruction-based methods that primarily focus on restoring normal features, ***3) Knowledge distillation-based approaches*** aim to enable the student network to understand and retain normality knowledge. Methods like Uniform T-S [14] use a forward distillation scheme where the teacher and student networks compare cropped patches of varying sizes, but this often leads to overgeneralization. RD4AD [15] tackles this by using a reverse distillation scheme that compresses and restores features, partially addressing this problem but not fully solving normality forgetting. Other methods like CDO [16], DeSTSeg [24], Pull&Push [25], and RD++ [26] try to enhance anomaly localization by incorporating pseudo-anomalous data, but they still face challenges in sustaining normal feature memory.

Recently, ASTN [27] introduced an asymmetric T-S network to modify anomalous information representation. Still, teacher knowledge during training often overwhelms the student's normality memory, affecting anomaly detection precision. By comparison, our Cascade Distillation scheme prevents direct teacher knowledge transfer, ensuring the student network effectively retains normality knowledge.

B. Semi-supervised anomaly detection scenario

In real industrial applications, foreground detection [33]–[37] and salient object detection [38]–[42] are often employed to eliminate complex background noise, thereby enhancing the effectiveness of anomaly detection. However, the accuracy of unsupervised anomaly detection remains limited. As a result, semi-supervised anomaly detection methods [18]–[20], which utilize a small amount of labeled anomalous data, are increasingly applied in industrial scenarios where high precision is required. In semi-supervised scenarios, DevNet [18] focuses on learning deviations from normal patterns to effectively distinguish between normal and anomalous samples. PRN [19], on the other hand, learns to differentiate abnormal features by generating multi-scale normal prototypes. However, both methods struggle to address the imbalance issue between normal and anomalous samples [20]. On the contrary, our proposed SOAIL method addresses this challenge by leveraging labeled anomalous samples to learn the fusion of multi-level discrepancy maps. This approach allows for a more robust and stable improvement in industrial anomaly detection, especially when combined with human interaction.

III. Robust Industrial Anomaly Detection

A. The overview of the proposed method

The task of unsupervised anomaly detection is to identify and locate anomalous regions in a query set $\mathcal{I}^q = \{I_1^q, \ldots, I_n^q\}$ containing both normal and abnormal samples, based solely on training with normal set $\mathcal{I}^t = \{I_1^t, \ldots, I_n^t\}$. The goal of knowledge distillation in the field of anomaly detection is to remember the normality of the student network on the training set and avoid excessive generalization to out-of-distribution data. However, existing T-S networks [14]–[16] overlook the issues of "shared convolutional feature weights" and "normality forgetting", which result in a consistently small discrepancy between the T-S model.

As shown in Fig. 3, we propose Robust Industrial Anomaly Detection (RIAD) as a solution to address the problems.

Our RIAD method primarily consists of four parts: a basic Reverse Distillation (RD) student branch, a structure

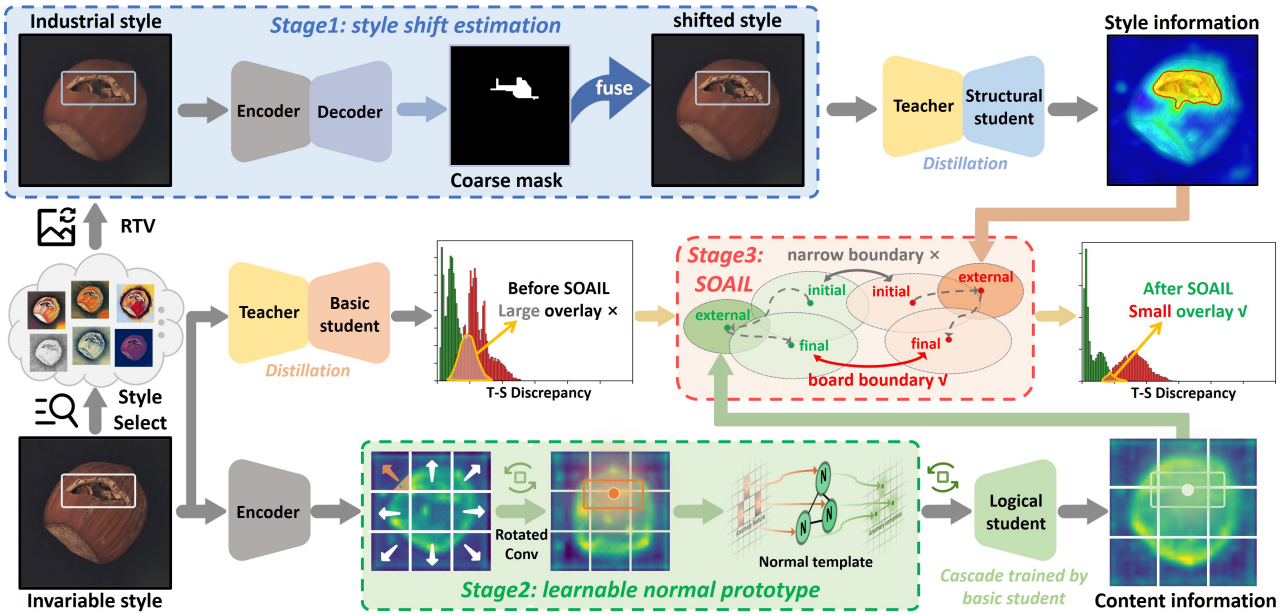


Fig. 3. The method architecture of our approach. The whole process could be divided into three stages: style shift estimation (SSE), learnable normal prototype (LNP), and semi-online anomaly incremental learning (SOAIL) module.

learning student branch, a logical normal generation student branch, and a semi-online anomaly incremental learning module. Given an input sample $I \in \mathcal{I}^q$, following reverse distillation network [15], the multi-scale knowledge features $f_{E_p}^k \in \mathbb{R}^{H_k \times W_k \times C_k}$ are extracted by pre-trained ResNet [43] encoder E_p and fed into bottleneck module for anomaly embedding representation ϕ_b. Then we use a basic decoder D_b to restore the normal features $f_{D_b}^k \in \mathbb{R}^{H_k \times W_k \times C_k}$, where $H_k \times W_k$ denotes the spatial dimension, C_k is the number of channels and k^{th} indicates the layer index in the teacher and student model. Mathematically, the basic student training loss $\mathcal{L}_B$ can be calculated as follows:

$$\mathcal{L}_B = \sum_{k=1}^{K} \left(1 - \frac{\left(f_{E_p}^k\right)^T \cdot f_{D_b}^k}{\left\| f_{E_p}^k \right\| \left\| f_{D_b}^k \right\|} \right), \tag{1}$$

where K = 3. The structural student D_s branch of our method is conducted based on the "industrial style feature" and the Style Shift Estimation (SSE) Approach. For the structural student branch, we also train D_s via a similar process, obtaining the loss $\mathcal{L}_S$. The logical student D_l branch is an AutoEncoder architecture and is performed based on the Learnable Normal Prototype (LNP). Notably, the logical student is trained solely by the basic student, with the aim of acquiring globally focused normal features. Finally, with the assistance of the semi-online anomaly incremental learning module, the multi-level discrepancy maps from the above three students and the teacher are fused by an iterative anomaly integrator to complete anomaly detection. Additionally, owing to the interpretability of cascade distillation, our model can interact with operators and continuously improve its performance through few-shot incremental learning.

B. Style shift estimation based structural student branch

In the structural student, our proposed Style Shift Estimation (SSE) Approach assumes that, compared to ***content*** generalization, the student network is less adept at local ***style*** generalization, thus exhibiting greater discrepancies from the teacher network in regions with style changes [44]–[46]. Based on this premise, in the structural student branch, our SSE first alters the global style of all query set $\mathcal{I}^q$, then employs local anomaly priors to generate coarse anomaly masks M_c, which are used to merge the predicted anomaly locations from the original style images with the style-altered images, achieving the local anomaly style shift.

1) Global industrial style transfer: For global image style transfer, as we can see in the stage 1 part of Fig. 3, instead of choosing various natural styles, we utilize the Relative Total Variation (RTV) [47] to extract structural information while smoothing texture information. Compared with the former, the latter approach modifies the visual style of the industrial sample image without affecting the inherent industrial style. The objective function of RTV is expressed as follows:

$$\underset{R}{\operatorname{argmin}} \sum_{n} (R_n - I_n)^2 + \omega \cdot \left(\frac{T_x(n)}{L_x(n) + \varepsilon} + \frac{T_y(n)}{L_y(n) + \varepsilon} \right), \tag{2}$$

where R is the industrial-style image, and $(R_n - I_n)^2$ encourages structural similarity to the query image. ω is a smoothness weight; a grid search over [0.1, 0.9] (step 0.1) showed that $\omega = 0.4$ best balances structure preservation and texture smoothing. T and L denote the windowed total and inherent variation, respectively, and ε is a small constant to prevent

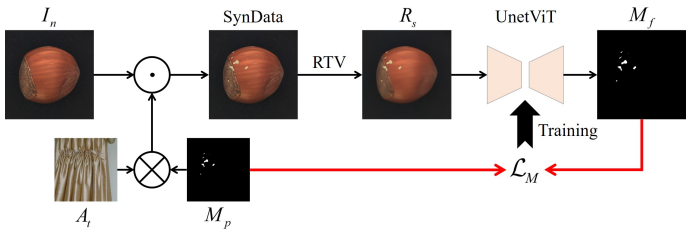


Fig. 4. The demonstration of anomaly style shift estimation approach.

division by zero. And the formulas of T and L are as follows:

$$\begin{cases} T_x = G_\sigma * |\partial_x R|, \\ T_y = G_\sigma * |\partial_y R|, \\ L_x = |G_\sigma * \partial_x R|, \\ L_y = |G_\sigma * \partial_y R|, \end{cases} \tag{3}$$

where G_σ is a Gaussian kernel with scale σ controlling its size for structure extraction. Among [1, 2, 3, 4, 5], $\sigma = 3$ yielded the best performance in most industrial scenarios.

2) Style shift estimation: Compared to the traditional RTV method that relies on parameter optimization, we design an end-to-end anomaly style shift estimation approach. As shown in Fig. 4, synthetic data I_s is generated using Perlin noise M_p, texture anomaly A_t, and normal sample I_n. This synthetic data I_s then undergoes the RTV process, resulting in an industrial-style synthetic sample R_s, which can be used for training anomaly foreground prediction. To enhance the detection capability of anomaly foregrounds, we employ an advanced transfer model, which is trained using the industrial-style synthetic sample R_s to predict the structural anomaly foreground mask M_f from the input sample. The procedure of mask generation is defined as:

$$\begin{aligned} &\underbrace{\text{SynData} \leftarrow \text{SynFuse}(I_n, A_t, M_p)}_{\Downarrow} \\ &M_f = \text{UnetViT}(\text{RTV}(\overline{\text{SynData}})), \end{aligned} \tag{4}$$

where SynFuse represents the anomaly feature fusion method from [48], SynData denotes generated synthesis anomaly data and UnetViT represents Unet-Transformer model [49]. The cross-entropy loss is employed to train the UnetViT with the difference between the generated mask M_f and the Perlin noise M_p, which can be calculated as $\mathcal{L}_M$:

$$\begin{aligned} \mathcal{L}_M = -\sum_i \Big[& M_f^i \log\left(M_p^i\right) \\ & + \left(1 - M_f^i\right) \log\left(1 - M_p^i\right) \Big], \end{aligned} \tag{5}$$

where i indexes each pixel of M_f and M_p. In the inference stage, the coarse anomaly mask is fused with the input image I and the industrial style image R to obtain the locally transformed anomaly style image $I_a \leftarrow \text{SynFuse}(I, R, M_f)$.

The structural student branch of Fig. 3 shows the difference between the test input image I and the local anomaly style image I_a. It is clearly visible that the style difference between the anomaly and the background is more pronounced in I_a. Furthermore, we feed I_a into the teacher P and structural-student D_s to obtain the structural T-S discrepancy map and then project the top m features with the largest discrepancies back to the patches of I_a. The image I_a after the anomaly style shift estimation approach shows higher discrepancies in anomalous pixels, and the features with the largest discrepancies correspond to the style-transformed anomaly regions in the original image. In contrast, the basic T-S discrepancy maps for I exhibit erroneous anomaly classifications in the mapping relationships.

C. Cascade distillation based logical student branch

The logical student branch is a crucial element of our method, as it bears the core responsibility of the knowledge distillation method in the field of industrial anomaly detection: for any input sample, the student network can generate a corresponding "perfectly" normal feature. As shown in the stage 2 part of Fig. 3, we design a Learnable Normal Prototype (LNP) module and a cascade distillation method to achieve this target.

1) Learnable normal prototype: To obtain a unified normal template, this module employs the "align & remember" strategy for normality learning. Compared to existing alignment methods, we use Adaptive Rotated Convolution (ARC) [50] to learn the rotation angles of input features. Notably, in the logical student branch, the pre-trained teacher is replaced by autoencoder E_a. Thus, given the autoencoder features $f_{E_a}^k \in \mathbb{R}^{H_k \times W_k \times C_k}$ of the query image, ARC determines the rotation angle θ and combination weight λ of the object through a routing function $\text{Routing}(\cdot)$, which consists of depthwise convolution, ReLU, pooling, and linear projection:

$$\theta, \lambda \leftarrow \text{Routing}(f_{E_a}^k). \tag{6}$$

Then the original convolution kernels W_i can be rotated as follows:

$$W_i' \leftarrow \text{Rotate}\left(W_i; \theta_i\right), i = 1, 2, \cdots, n, \tag{7}$$

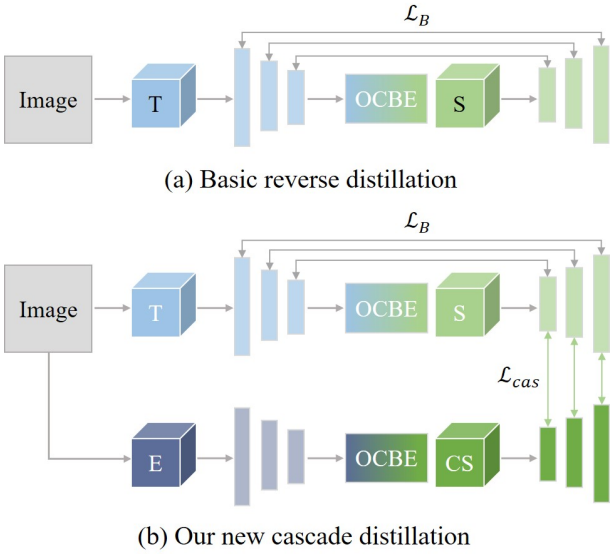
where $W_i' \in \mathbb{R}^{C_k \times C_k \times k \times k} (i = 1, 2, \ldots, n)$ is the rotated kernel, and $\text{Rotate}(\cdot)$ is the rotate procedure. Then, the aligned feature $g_{E_a}^k$ can be calculated as follows:

$$g_{E_a}^k = (\lambda_1 W_1' + \lambda_2 W_2' + \cdots + \lambda_n W_n') * f_{E_a}^k. \tag{8}$$

After the feature alignment operation, we propose to learn a set of normal representation features (named "prototypes") from the aligned features and generate normality priors to provide normal information. First, LNP projects the $g_{E_a}^k$ to a set of C_k-dimensional features $\check{g}_{E_a}^k = \left\{\check{g}_{E_a 1}^k, \check{g}_{E_a 2}^k, \ldots, \check{g}_{E_a N}^k\right\}$, where $N = H_k \times W_k$ is the total number of the input layer features. Next, we initial U prototypes $P = \{p_1, p_2, \ldots, p_U\}$ and normality factors $S = \{s_1, s_2, \ldots, s_U\}$ to learn the representative normal information, and the u-th position-wise learnable normality weight is measured as follows:

$$e_u = \frac{\exp\left(-s_u \left\| \check{g}_{E_a}^k - p_u \right\|^2\right)}{\sum_{j=1}^{U} \exp\left(-s_j \left\| \check{g}_{E_a}^k - p_j \right\|^2\right)}, \tag{9}$$

where $\check{g}_{E_a}^k - p_u$ is the distance between the N C-dimensional normal vectors and U prototype vectors. Following the [51],



(a) Basic reverse distillation

(b) Our new cascade distillation

Fig. 5. Schematic comparison between the basic reverse distillation method and our proposed cascade distillation approach.

we set the number of prototype vectors to 50. After that, we sum over the U results to obtain e and perform the channel-wise multiplication $\otimes$ between aligned feature $g_{E_a}^{\check{k}}$ and normality prototype weights e to obtain the unified normal template $f_{E_a}^{\check{k}}$, which is formulated as:

$$\begin{aligned} &\underbrace{e \leftarrow \sum_{u=1}^{U} \mathrm{BRM}\left(e_u\right)}_{\Downarrow} \\ &f_{E_a}^{\check{k}} = g_{E_a}^{\check{k}} \otimes \overline{e}, \end{aligned} \tag{10}$$

where BRM$(\cdot)$ contains BN layer with ReLU and the mean layer. To ensure e remembers the normality, we restrict the normal similarity between the aligned normal feature $g_{E_a}^{\check{k}}$ and normal template $f_{E_a}^{\check{k}}$ in the training stage:

$$\mathcal{L}_N = \sum_{k=1}^{K} (\left\| f_{E_a}^{\check{k}} - g_{E_a}^{\check{k}} \right\|^2). \tag{11}$$

2) Cascade distillation: In order to reduce the anomaly information from the pre-trained teacher network, we trained our logical student solely from the basic student. As shown in Fig. 5, unlike traditional reverse distillation where the student directly mimics the teacher's features and may potentially inherit noise or anomalous patterns, our cascade distillation introduces an intermediate basic student. The basic student first adapts to normality, and the logical student then distills knowledge from it. Finally, the logical student learns from the basic student that has already adapted to normal patterns.

Formally, given the logical student feature $f_{D_l}^k$ and basic student $f_{D_b}^k$, we calculate their feature cosine distance along the channel axis as logical student distillation loss:

$$\mathcal{L}_{cas} = \sum_{k=1}^{K} (1 - \frac{\left(f_{D_b}^k\right)^T \cdot f_{D_l}^k}{\left\| f_{D_b}^k \right\| \left\| f_{D_l}^k \right\|}). \tag{12}$$

D. Semi-online anomaly incremental learning module

To fully leverage the T-S discrepancy maps from BSB, SSB, and LSB for effective anomaly localization, we design a semi-online anomaly incremental learning module (SOAIL) that supports both unsupervised and semi-supervised anomaly detection scenarios. This module is designed for human-machine interaction and can adapt to increasing amounts of interactive anomaly data provided by staff. In semi-supervised scenarios, the anomaly incremental learning module can better fuse multi-level discrepancy maps to generate the anomaly map. Specifically, SOAIL consists of multiple MLP layers, LayerNorm (LN) layers, and ReLU layers. For a given anomaly data sample $I_a \in \mathcal{J}^q$ and the corresponding ground truth mask M_a, the T-S discrepancy maps DM_B, DM_S, and DM_L are generated by BSB, SSB, and LSB, respectively. The generated anomaly map of SOAIL is as follows:

$$M_{fuse} \leftarrow \mathrm{SOAIL}(DM_B, DM_S, DM_L). \tag{13}$$

Then, we use the cross-entropy loss to measure the difference between the generated anomaly map M_{fuse} and the ground truth M_a, which can be calculated as $\mathcal{L}_{\mathrm{SOAIL}}$:

$$\begin{aligned} \mathcal{L}_{\mathrm{SOAIL}} = -\sum_i \Big[& M_a^i \log\left(M_{fuse}^i\right) \\ & + \left(1 - M_a^i\right) \log\left(1 - M_{fuse}^i\right) \Big], \end{aligned} \tag{14}$$

where i indexes each pixel of M_{fuse} and M_a.

E. Loss function

In the unsupervised scenario, our RIAD method employs a loss function composed of a three-branch distillation loss, a coarse defect detection loss, and a normality remembering loss:

$$\mathcal{L} = \mathcal{L}_B + \mathcal{L}_S + \mathcal{L}_{cas} + \lambda_1 \mathcal{L}_M + \lambda_2 \mathcal{L}_N, \tag{15}$$

where λ_1 and λ_2 are balance factors. Following the prior works [19], [51]–[53], we set the λ_1 = 0.3 and λ_2 = 0.7. In the semi-supervised scenario, to address the imbalance issue between normal and anomalous data during training, we restrict the incremental learning process to the SOAIL module. Accordingly, the loss function in this stage is defined as:

$$\mathcal{L} = \mathcal{L}_{\mathrm{SOAIL}}. \tag{16}$$

IV. Experiments

In this section, we conduct a series of experiments to verify the effectiveness of the proposed method and compare it with state-of-the-art methods. The datasets involved in these experiments include industrial anomaly detection datasets MVTec AD [22], BTAD [54], MVTec-3D AD [55], and the traditional defect detection dataset KolektorSDD2 [56]. In these unsupervised and semi-supervised experiments, Image-AUC, Pixel-AUC, and PRO are used as evaluation metrics.

TABLE I
THE QUANTITATIVE LOCALIZATION RESULTS (PIXEL-AUC / PRO) OF VARIOUS METHODS ON MVTEC AD IN THE UNSUPERVISED SETTINGS

Category	Reconstruction-based			Embedding-Based				Knowledge distillation-Based						Ours
	DDAD (ICCV 23)	Transfusion (ECCV 24)	NSA+SSMCTB (TPAMI 24)	PatchCore (CVPR 22)	SimpleNet (CVPR 23)	DMAD (CVPR 23)	SFRAD (TNNLS 24)	RD4AD (CVPR 22)	DeSTSeg (CVPR 23)	CDO (TII 23)	PullPush (TCSVT 23)	ATSN (TIM 24)	DBKD (TIM 25)	
carpet	98.9/95.8	-/95.9	95.6/88.2	99.0/96.6	98.2/93.2	99.1/-	99.0/94.8	98.9/97.0	96.1/93.6	99.1/96.8	**99.5**/**98.3**	99.1/97.3	98.7/97.0	**99.3**/**97.9**
grid	99.1/**98.4**	-/**98.0**	99.2/97.6	98.7/96.0	98.8/94.1	99.2/-	98.8/95.1	**99.3**/97.6	99.1/96.4	98.4/96.0	**99.4**/97.7	99.0/97.2	98.1/95.1	**99.4**/97.8
leather	**99.5**/**99.1**	-/96.2	**99.5**/94.1	99.3/98.9	99.2/90.5	**99.5**/-	99.4/97.3	99.4/**99.1**	**99.7**/**99.0**	99.2/98.3	**99.7**/98.7	99.4/**99.1**	99.4/99.1	**99.5**/**99.1**
tile	92.1/95.1	-/95.0	**99.1**/**97.8**	95.6/87.3	97.0/84.3	96.0/-	95.8/80.2	95.6/90.6	98.0/**95.5**	97.2/90.5	96.8/90.3	95.0/88.8	96.5/92.9	**98.1**/93.6
wood	94.5/93.0	-/**94.8**	93.5/92.7	95.0/89.4	94.5/86.2	95.5/-	95.5/88.8	95.3/90.9	**97.7**/**96.1**	95.9/92.9	95.2/93.2	94.8/92.4	95.2/91.5	**96.7**/93.5
Average	96.8/**96.3**	-/95.9	97.4/94.1	97.5/93.6	97.5/89.7	97.9/-	97.7/91.2	97.7/95.0	**98.1**/96.1	97.9/94.9	**98.1**/95.6	97.5/95.0	97.5/95.1	**98.6**/**96.4**
bottle	97.7/95.0	-/**97.3**	98.4/96.4	98.6/96.2	98.0/91.6	98.9/-	98.6/94.8	98.7/96.6	**99.2**/96.6	**99.3**/**97.2**	98.7/95.6	98.8/96.4	99/97.1	**99.2**/**97.2**
cable	95.6/89.5	-/85.5	97.5/81.5	**98.4**/92.5	97.6/92.1	98.1/-	98.3/96.7	97.4/91.0	97.3/86.4	97.6/**94.2**	95.7/87.5	98.0/92.7	98/92.6	**98.7**/**93.6**
capsule	97.5/91.4	-/92.1	97.9/92.7	98.8/95.5	98.9/94.6	98.3/-	**99.1**/95.1	98.7/95.8	**99.1**/94.2	98.6/93.0	97.8/89.9	98.6/95.7	98.9/**96.4**	**99.0**/**96.1**
hazelnut	97.3/91.1	-/**97.6**	97.9/**98.4**	98.7/93.8	97.9/91.5	99.1/-	98.6/95.7	98.9/95.5	**99.6**/97.6	**99.2**/97.4	98.6/96.1	98.9/95.1	**99.2**/95.5	**99.2**/**97.6**
metal_nut	96.8/93.0	-/94.1	98.3/**97.5**	98.4/91.4	**98.8**/89.4	97.7/-	**98.7**/94.5	97.3/92.3	98.6/95.0	98.5/**95.7**	97.8/93.2	96.0/88.4	98.4/93.5	98.2/94.3
pill	92.5/94.5	-/**96.2**	98.4/90.1	97.4/93.2	98.6/93.6	98.7/-	98.5/95.8	98.2/96.4	**98.7**/95.3	**98.9**/96.6	98.6/94.9	98.0/96.1	98/96.3	**98.7**/**97.2**
screw	99.0/95.6	-/97.0	96.4/95.1	99.4/**97.9**	99.3/91.4	**99.6**/-	99.5/97.3	**99.5**/**98.2**	98.5/92.5	99.0/94.3	96.9/85.6	99.1/96.6	99.3/97	**99.5**/**98.2**
toothbrush	98.6/**95.7**	-/94.1	95.4/86.7	98.7/91.5	98.5/90.2	**99.4**/-	98.8/90.0	99.1/**94.5**	99.3/94.0	98.9/90.5	99.1/91.8	98.9/93.2	**99.2**/94.4	**99.4**/94.2
transistor	93.1/90.1	-/83.9	88.3/68.4	96.3/83.7	97.6/68.5	95.4/-	**96.6**/91.8	92.5/78.0	89.1/85.7	95.3/**92.6**	**99.2**/**97.6**	94.5/83.9	96.4/79.5	95.1/89.7
zipper	98.3/93.6	-/**97.2**	94.7/95.8	98.8/97.1	**98.9**/92.5	98.3/-	98.8/95.6	98.2/95.4	**99.1**/**97.4**	98.2/94.3	97.9/93.0	98.3/95.0	98.3/96.4	98.5/95.5
Average	96.6/93.0	-/93.5	96.3/90.3	**98.4**/93.3	**98.4**/89.5	**98.4**/-	98.6/94.7	97.9/93.4	97.5/93.5	**98.4**/**94.6**	98.0/92.5	97.9/93.3	**98.5**/93.9	**98.5**/**95.4**
Total_Avg.	96.7/94.1	-/94.3	96.7/91.5	98.1/93.4	98.1/89.6	**98.2**/-	98.3/93.6	97.8/93.9	97.9/94.4	**98.2**/**94.7**	98.1/93.6	97.8/93.9	**98.2**/94.2	**98.6**/**95.7**

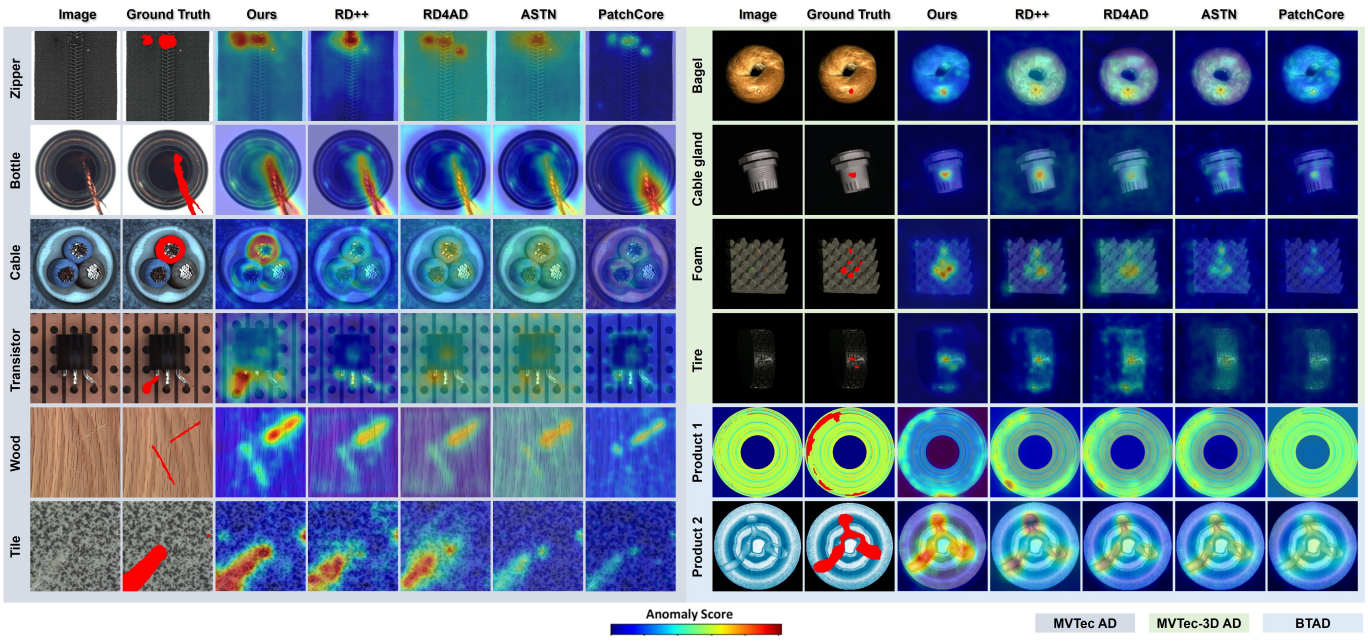


Fig. 6. Visualizations of our proposed method and several comparative methods on the MVTec AD, MVTec-3D AD and BTAD benchmarks.

A. Experiment settings

1) Dataset: **MVTec AD** [22] is a widely used dataset for industrial anomaly detection. It contains a total of 5354 high-resolution images, comprising 15 sub-datasets that can be further divided into 10 object sub-categories and 5 texture sub-categories. Each sub-category includes a training set with only normal samples and a test set with various types of anomalies. **BTAD** [54] contains 2540 images and is a publicly available dataset for industrial image anomaly detection. **MVTec-3D AD** [55] includes 10 sub-categories similar to those in the MVTec AD dataset, with the addition of over 4000 3D point cloud data to capture details and defects on three-dimensional surfaces. However, in this work, only RGB images are used for anomaly detection. **KolektorSDD2** [56] contains approximately 2000 normal images for the training set and about 1000 test images for defect detection on a single type of industrial surface. Compared to the aforementioned datasets, KolektorSDD2 provides some abnormal images that can be used both for unsupervised and semi-supervised anomaly detection scenarios.

2) Evaluation metric: **Image-AUC** and **Pixel-AUC** are used to evaluate the performance of a classifier in detecting and localizing anomalies at the image and pixel level. These metrics are calculated by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings and then computing the area under the "Receiver Operating Characteristic" (AUROC) curve. The formulas of TPR and FPR are expressed as follows:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{17}$$

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}, \tag{18}$$

where TP, TN, FP, and FN are the number of true positives, true negatives, false positives, and false negatives, respectively. **PRO** is also employed to evaluate the performance in localized anomalies at the pixel level. It measures the normalized area under the "Per-Region Overlap" curve (AUPRO) between detected anomaly regions and ground truth anomaly regions.

TABLE II
IMAGE-LEVEL ANOMALY DETECTION AUC(%) ON MVTEC AD. RESULTS ARE AVERAGED OVER ALL CATEGORIES

PatchCore (CVPR 22)	RD4AD (CVPR 22)	OCR-GAN (TIP 23)	CDO (TII 23)	DCAE (TII 24)	DeSTSeg (CVPR 23)	PullPush (TCSVT 23)	NSA+SSMCTB (TPAMI 24)	Transfusion (ECCV 24)	ATSN (TIM 24)	SFRAD (TNNLS 24)	DBKD (TIM 25)	Ours
99.1	98.5	98.3	98.2	**99.2**	98.6	94.8	97.7	**99.2**	98.4	99.1	98.1	99.6

TABLE III
THE QUANTITATIVE LOCALIZATION RESULTS (PIXEL-AUROC / PRO) ON THE MVTEC-3D AD DATASET WITH PURE RGB INPUTS

Category	Shape-Guided (ICML 23)	M3DM (CVPR 23)	CDO (TII 23)	MMRD (AAAI 24)	Ours
bagel	98.7/94.6	**99.1**/95.2	**99.3**/**97.5**	˜/**97.0**	98.9/96.5
cable_gland	99.1/97.2	**99.4**/97.2	99.2/**98.3**	˜/**98.3**	**99.5**/**99.2**
carrot	**99.1**/96.0	**99.4**/97.3	**99.4**/**98.1**	˜/**98.2**	**99.1**/97.3
cookie	97.6/91.4	97.1/89.1	**98.1**/86.3	˜/**92.4**	**97.8**/**92.6**
dowel	98.5/95.8	**99.7**/93.2	98.8/97.6	˜/97.6	**99.0**/**98.1**
foam	91.2/77.6	**95.6**/**84.3**	89.1/70.5	˜/87.5	**99.3**/**96.7**
peach	99.3/93.7	**99.4**/97.0	**99.6**/**98.6**	˜/**98.1**	99.1/97.1
potato	**99.1**/94.9	99.0/95.6	**99.1**/**96.1**	˜/97.5	**99.2**/**98.1**
rope	99.3/95.6	99.3/96.8	**99.6**/**97.1**	˜/**98.4**	**99.5**/96.7
tire	99.2/95.7	**99.5**/96.6	**99.4**/**97.4**	˜/97.3	99.3/**97.8**
Total Avg.	98.1/93.3	**98.8**/94.2	98.2/93.75	˜/**96.2**	**99.1**/**97.0**

TABLE IV
THE IMAGE/PIXEL-LEVEL AUROC RESULTS ON THE BTAD DATASET

Category	PatchCore (CVPR 22)	RD4AD (CVPR 22)	RD++ (CVPR 23)	RealNet (CVPR 24)	Ours
01	88.7/95.0	96.3/96.6	96.8/96.2	**100**/**96.8**	**98.3**/**97.9**
02	76.0/94.9	86.6/96.0	**88.1**/**96.4**	86.7/96.2	**91.2**/**97.1**
03	**99.8**/99.2	**99.8**/99.5	99.6/**99.7**	99.6/**99.7**	**100**/**99.7**
Average	88.2/96.4	94.2/97.4	94.8/97.4	**95.4**/97.6	**96.5**/**98.2**

Consistent with [22], PRO is calculated with an average per-pixel FPR threshold of 0.3.

3) Two supervision scenarios:

- **The unsupervised scenario.** In traditional industrial anomaly detection, the unsupervised model is typically optimized with the training set containing only normal samples and the test set containing both normal samples and long-tailed distributed anomaly samples. In the local anomaly transfer module of this work, synthetic data is used solely to train the model for foreground anomaly prediction and does not participate in the anomaly detection model training process.
- **The semi-supervised scenario.** In practical industrial applications, to improve anomaly detection performance, few-shot abnormal data is included in the training set to aid in model performance. In this work, we primarily utilize a small number of anomaly samples in the semi-online anomaly incremental learning module to integrate multi-level discrepancy maps.

4) Implementation details: All images were resized to (256, 256) and normalized using the mean and standard deviation derived from the ImageNet dataset. The wide-resnet50 was used for the teacher network. To ensure consistency of the results, we re-evaluated the methods involved in the experiments and averaged the results from five runs to obtain the final results. All experiments were conducted on a machine equipped with an Intel i5-13600KF CPU, 32G DDR4 RAM, and an NVIDIA Tesla P100-16GB GPU.

B. Quantitative results

1) Unsupervised anomaly detection and localization: In this section, we compared various existing methods for unsupervised anomaly detection and localization to validate the effectiveness of our proposed method. Firstly, we conducted experiments on the MVTec AD dataset, with the results presented in Table I for anomaly localization and Table II for anomaly detection. In Table I, we compared three major categories of anomaly detection methods: Reconstruction-Based methods, Embedding-Based methods, and Knowledge distillation-Based methods. The Reconstruction-Based methods include DDAD [12], TransFusion [13], and NSA+SSMCTB [9]; the Embedding-Based methods include PatchCore [6], SimpleNet [7], DMAD [8], and SFRAD [57]; the KD-Based methods include RD4AD [15], DeSTSeg [24], CDO [16], Pull&Push [25], ATSN [27], and DBKD [58]. From the Pixel-AUC and PRO results in Table I, it can be seen that our approach consistently demonstrates stable and superior performance across various categories. As shown in the results, our method exhibits optimal or near-optimal performance across 12 out of 15 categories on the MVTec AD dataset. This represents 80% of the total categories, underscoring the robustness and generalizability of our method. For example, Our method shows significant improvements in the PRO for the "tile" and "transistor" categories, with increases of +2.7 and +10.2, respectively, in contrast to the latest method DBKD. compared to the baseline model RD4AD, our method exhibits improvements of +1.8 in PRO and +0.8 in Pixel-AUC. In Table II, we presented the Image-Level Anomaly Detection AUC results on the MVTec AD dataset. It can be seen that our method achieved an average AUC of 99.6, which is +1.1 percentage points higher than the baseline model RD4AD's 98.5, and outperformed the state-of-the-art method TransFusion by +0.4 points.

To achieve a more comprehensive comparison, we also evaluate our method on the MVTec-3D AD and BTAD datasets. As shown in Table III, our method demonstrated outstanding performance across several categories. For instance, in the foam category, our method achieved Pixel-AUROC and PRO scores of 99.3 and 96.7, respectively, significantly outperforming other methods. Notably, our method exhibited an improvement of +1 points in Pixel-AUROC compared to Shape-Guided [59] and an improvement of +0.8 points in PRO compared to the state-of-the-art method MMRD [51]. Additionally, comparative experiments on the BTAD dataset illustrate that our method consistently outperforms existing advanced methods such as RD++ [26] and RealNet [60]. As we can see in Table IV, our method achieved an Image-AUROC of 91.2 in category 02, significantly higher than the RealNet method, which scored 86.7.

TABLE V
COMPUTATIONAL COMPLEXITY (TRAINING MEMORY AND INFERENCE TIME) AND I-AUC/P-AUC/PRO RESULTS FOR DIFFERENT MODEL VARIANTS IN THE ABLATION STUDY ON THE MVTEC AD BENCHMARK.

Model variants	SSB		LSB			Fuse	Training Memory(MB)	Inference Time(ms)	Performance
	RTV	SSE	ARC	LNP	$\mathcal{L}_{cas}$	SOAIL			
(A)	×	×	×	×	×	×	100.2	14.2	98.5/96.9/93.2
(B)	✓	×	×	×	×	×	103.5 (+3.3)	16.7 (+2.5)	98.9/97.4/93.3
(C)	✓	✓	×	×	×	×	207.2 (+103.7)	37.4 (+20.7)	99.0/97.9/94.7
(D)	✓	✓	✓	×	×	×	213.5 (+6.3)	37.9 (+0.5)	99.1/98.0/94.5
(E)	✓	✓	✓	✓	×	×	226.5 (+13.0)	40.7 (+2.8)	99.3/98.2/95.2
(F)	✓	✓	✓	✓	✓	×	327.1 (+100.6)	55.1 (+14.4)	**99.6/98.6**/95.3
(G)	✓	✓	✓	✓	✓	✓	342.7 (+15.6)	57.3 (+2.2)	**99.6/98.6/95.7**

TABLE VI
COMPLEXITY COMPARISON BETWEEN THE PROPOSED METHOD AND CURRENT SOTA ALGORITHMS ON MVTEC AD. THE METHOD WITH THE SYMBOL * STANDS FOR THE RESULT OBTAINED BY RE-IMPLEMENTING THE ALGORITHM FRAMEWORKS FOR FASTER EXECUTION

Method	Training Memory(MB)	Inferece Time(ms)	I-AUC/P-AUC/PRO
PatchCore (CVPR 22)	1437.4	224.7	99.1/98.1/93.4
CDO (TII 23)	419.2	80.6	96.8/**98.2**/94.3
TransFusion (ECCV 24)	1144.4	342.8	**99.2**/98.1/**94.4**
RD4AD (CVPR 22)	355.4	**26.9**	98.5/97.8/93.9
RD4AD* (CVPR 22)	**100.2**	**14.2**	98.5/96.9/93.2
Ours	**342.7**	57.3	**99.6/98.6/95.7**

Overall, our method showcased state-of-the-art performance in unsupervised scenarios across three publicly benchmark datasets, particularly exhibiting significant improvements in both Pixel-AUROC and PRO metrics.

2) Computational complexity: We analyze the computational complexity of our RIAD method by outlining the time and space complexity of its key modules (SSB, LSB, and SOAIL) in Table V, and comparing its efficiency with state-of-the-art methods in Table VI. As shown in Table V, the SSB module enhances structured features with a space complexity of $O(L \times C^2 \times K^2 + C \times H \times W)$ and a time complexity of $O(L \times C^2 \times H \times W \times K^2)$, where L is the number of layers, C the number of channels, $H \times W$ the spatial resolution, and K the kernel size. Integrating the SSE module increases training memory by 107 MB and inference time by 23.2 ms, resulting in a pixel-level AUROC improvement from 96.9 to 97.4. The LSB module incurs a space complexity of $O(N \times C + U \times C + N \times U)$ and a time complexity of $O(N \times U \times C)$, where N is the number of spatial positions, U the number of prototypes, and C the feature dimension. Adding the LSB module increases training memory by 119.9 MB and inference time by 17.7 ms, significantly improving the pixel-level AUROC score from 94.7 to 95.3. Finally, integrating the SOAIL module leads to a modest increase in training memory (15.6 MB) and inference time (2.2 ms), yet contributes to an additional gain in the pixel-level PRO score from 95.3 to 95.7. As shown in Table VI, our proposed RIAD method achieves a PRO score of 95.7, outperforming RD4AD by 1.8% and RD4AD* by 2.5%. In terms of inference speed, RIAD achieves 57.3 ms, which is 3.9 times faster than PatchCore and 6 times faster than TransFusion, making it highly suitable for real-time industrial deployment. Although RIAD introduces a slightly higher training memory footprint (342.7 MB) compared to RD4AD* (100.2 MB), it maintains a favorable balance between accuracy and efficiency, significantly reducing inference latency while delivering state-of-the-art detection performance.

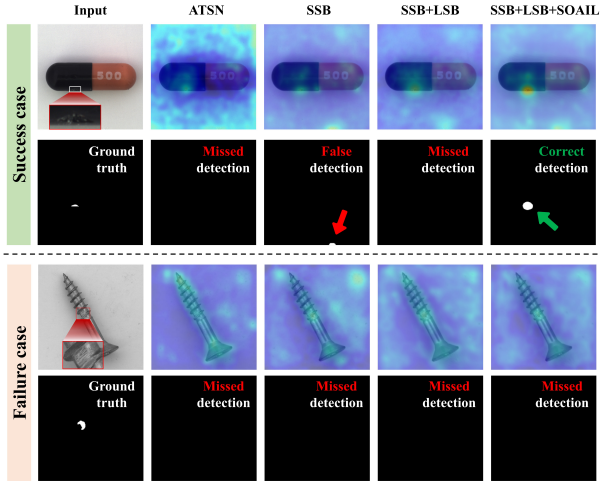


Fig. 7. Comparison of success and failure cases in detecting tiny defects across different methods.

C. Qualitative comparisons

To better illustrate the advantages of our method, Fig. 6 shows the anomaly localization results of various methods on the MVTec AD, MVTec-3D AD, and BTAD datasets. ***1)*** Our proposed method consistently achieves stable high-level segmentation across multiple benchmarks and categories. In contrast, PatchCore fails in anomaly localization on MVTec-3D AD and BTAD. ***2)*** Due to the feature representation capability of the structural student to transform the style of local anomalies, our method exhibits greater differences in anomaly regions, leading to a more complete representation of overall anomaly structures. For example, in the bottle, tile, and zipper categories of MVTec AD and the tire category of MVTec-3D AD, where anomalies are relatively hidden and dispersed, RD4AD and RD++ classify some anomalous pixels as false positives during localization. This is a critical error in high-precision segmentation tasks. In contrast, our method can more accurately localize the entire defective part. ***3)*** Unsupervised anomaly detection methods based on RD4AD generally fail to identify logical anomaly issues. For instance, ASTN and RD++ cannot locate missing parts of anomalies in

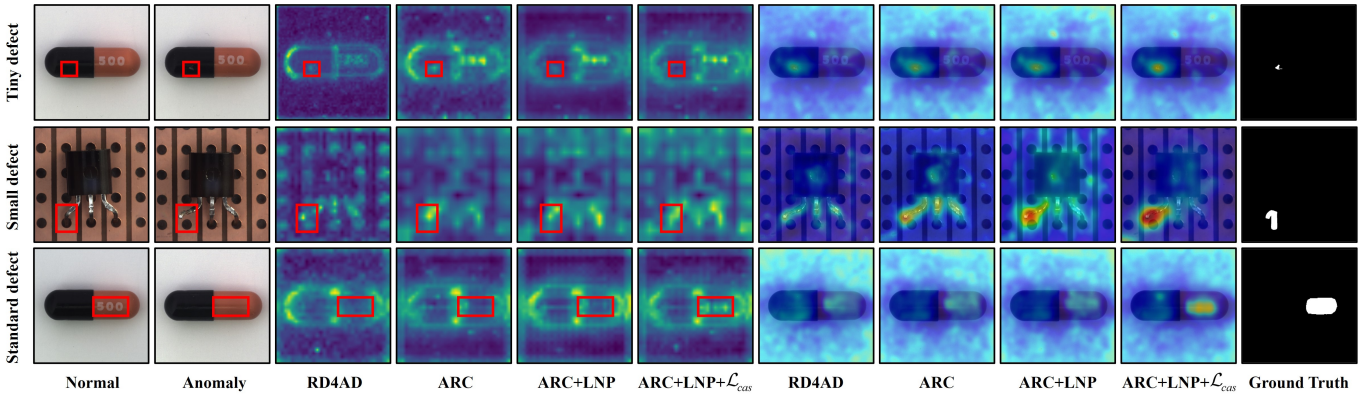


Fig. 8. Visualization of restored features from different variants of the LSB module on defects of varying sizes. From left to right: normal images, anomaly images, restored features, and anomaly maps from RD4AD, our logical student branch (LSB) with only ARC, with ARC and LNP, with ARC, LNP, and $\mathcal{L}_{\text{cas}}$, followed by the ground truth anomaly masks. The restored feature maps illustrate how different LSB configurations progressively enhance their ability to reconstruct normal patterns, leading to more precise anomaly localization. The integration of ARC, LNP, and $\mathcal{L}_{\text{cas}}$ enables a more effective suppression of normal features, thereby improving the detection of defects across different scales.

the transistor category of MVTec AD due to the "normality forgetting" problem. Conversely, our logical student can remember normality to some extent, supporting the effectiveness of our method. In conclusion, These qualitative comparisons, supported by the visual results in Fig. 6, demonstrate the robustness and superiority of our proposed method in anomaly detection.

D. Ablation study

In this section, extensive experiments were conducted to validate the various key components of the RIAD method. These components were combined into different model variants to test their performance on MTVec AD. Table V presents the quantitative results of each model's image AUROC, pixel AUROC, and PRO. Among them, model variant (A) is the baseline RD4AD* model.

1) SSB: The Structural Student Branch (SSB) primarily consists of RTV and SSE, where RTV transforms the global image style, and SSE preserves local anomaly information for anomaly style estimation. This combination enhances the T-S model discrepancy within anomaly regions, allowing for precise localization of the global anomaly structure. It is crucial to emphasize that SSE is essential among these two modules. As shown in Table V, the model (B) using only RTV performs worse than the baseline. This is because RTV, while transforming the global image style, partially diminishes the anomaly structure, resulting in a reduced T-S discrepancy under the global style transformation by RTV. In Table V, the quantitative results for model (C) reflect the findings of the qualitative experiments. Model (C) not only compensates for the decline observed in model (B) but also shows an improvement over the baseline, with increases of 1% in pixel-AUC and 1.5% in PRO. Additionally, it achieves a 0.5% improvement in image-AUC.

While the Structural Student Branch (SSB) is effective, it may negatively impact detection performance if small noise is mistakenly identified as an anomaly, as shown in the Fig. 7. This is because the SSB module focuses on capturing localized structural defects, which can sometimes overlook the overall normality. Therefore, it is necessary to incorporate the Logical Student Branch (LSB) and a fusion module to extract global features, thereby improving overall detection accuracy.

2) LSB: The purpose of the Logical Student Branch (LSB) is to maximize the memorization of normality without increasing memory cost. We employ ARC, LNP, and $\mathcal{L}_{\text{cas}}$ to prevent normality forgetting. From Table V, it can be observed that the performance of the model (D), which uses only ARC, is not outstanding. This is because the object angles in the MVTec AD dataset are mostly fixed. For instance, as shown in the third row of Fig. 8 for the capsule category, the restored feature after using ARC is similar to the feature from RD4AD. Subsequently, LNP was introduced to learn the normality template. However, it backpropagates the knowledge from the pre-trained model, giving the logical student strong anomaly reconstruction capabilities, which resulted in no improvement in the quantitative results of model (E).

To address this, we propose the cascade distillation loss $\mathcal{L}_{\text{cas}}$ that removes the knowledge of the pre-trained model and uses only the basic student for cascade distillation. This improvement significantly enhances the normality memorization capability of the LSB, leading model (F) to achieve the highest image-AUC of 99.6, pixel-AUC of 98.6, and a 0.8% improvement in PRO compared to SSB. As shown in Fig. 8, the reconstructed features after adding $\mathcal{L}_{\text{cas}}$ completely restore the normal information, and the anomaly maps achieve comprehensive and efficient logical anomaly localization.

3) SOAIL: Semi-online learning strategy is employed to fuse the discrepancy maps from the basic student branch, structural student branch, and logical student branch. The purpose of this module is to enhance anomaly detection by integrating information from different aspects of the model. Model variant (G) demonstrates a great improvement in performance compared to model (F), specifically achieving a 0.2% increase in PRO. This enhancement underscores the effectiveness of the SOAIL module in combining the strengths of the three branches to achieve more accurate and robust anomaly localization and detection. Additionally, we introduce

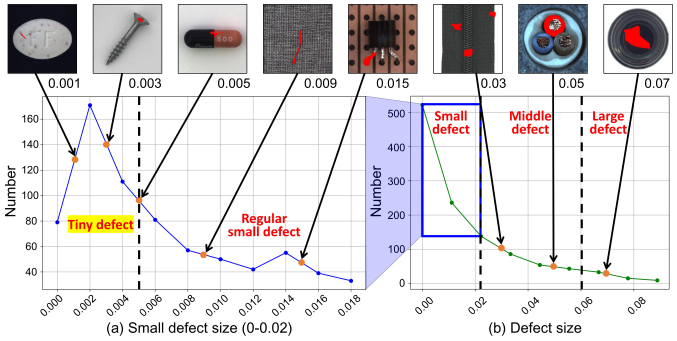


Fig. 9. The statistical distribution of defect sizes in the MVTec-2D dataset. (a) A detailed breakdown of small defects from (b), further categorized into tiny defects and regular small defects based on defect size. (b) The overall distribution of defect sizes, classified into small, middle, and large defects.

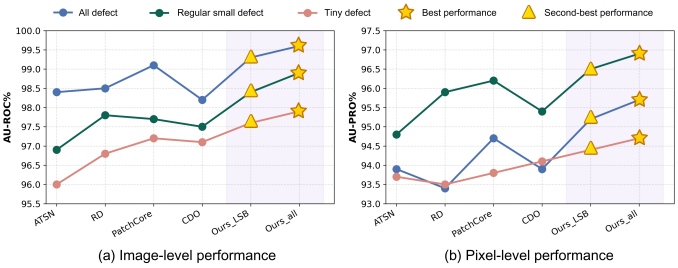


Fig. 10. Quantitative comparison of different methods across all defect, regular small defect, and tiny defect data sets. (a) Image-level performance in terms of AU-ROC%. (b) Pixel-level performance in terms of AU-PRO%. The best and second-best performances are highlighted with star and triangle markers, respectively.

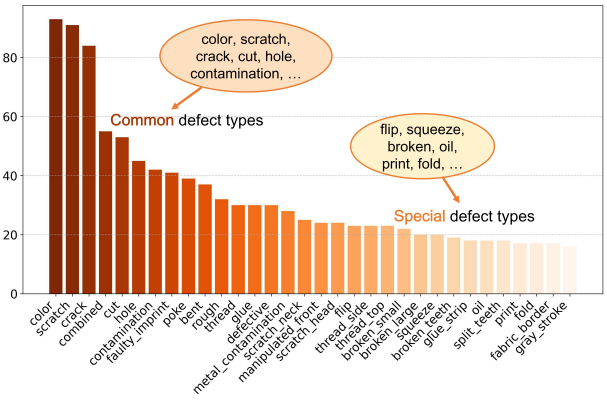


Fig. 11. The statistical distribution of common and special defect types in the MVTec-2D dataset. The x-axis represents different defect categories, while the y-axis indicates the number of images for each defect type.

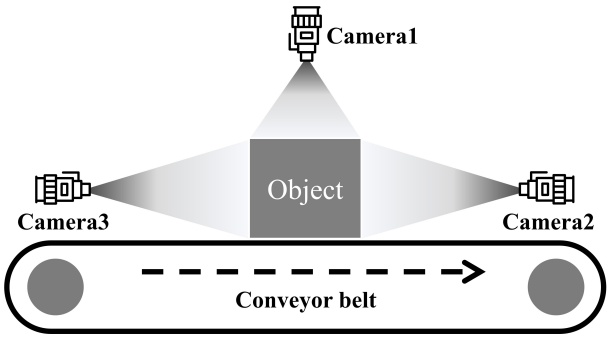


Fig. 12. Multi-view image acquisition schematic diagram.

the significant potential of SOAIL in semi-supervised anomaly detection in the subsequent sections.

E. Performance Across Diverse Defects and Settings

In this section, we discuss the performance and robustness of our proposed RIAD method across different defect sizes and various industrial environment settings.

1) Statistics of defect size: As shown in Fig. 9(a) and (b), we calculate the defect size statistics of the MVTec AD. Formally, we first define the defect size S as the ratio of the defect area D to the total image area I.

$$S = \frac{D}{I} \tag{19}$$

As shown in Fig. 9(b), we classify small defects as those with $S < 0.02$, middle defects as those within $0.02 \leq S \leq 0.06$, and large defects as those with $S > 0.06$. To further analyze the distribution of small defects, we present a more fine-grained breakdown in Fig. 9(a), where tiny defects are defined as those with $S < 0.005$, and regular small defects as those with $0.005 \leq S < 0.02$. According to the statistics, small defects, middle defects, and large defects account for 57.1%, 27.3%, and 15.6%, respectively. Notably, among the small defects, tiny defects make up 25.2%, while regular small defects constitute 31.9%.

2) Performance across defect sizes focusing on small defects: Based on the defined defect size subsets, we conducted further evaluations on datasets containing all defects, regular small defects, and tiny defects. As shown in Fig. 10, image-level performance generally declines with smaller defect sizes due to the removal of easily detectable large defects. However, pixel-level results reveal that regular small defects are often easier to localize than large or tiny ones. Despite the overall performance drop with decreasing defect size, our method consistently outperforms others, demonstrating strong robustness across scales.

The LSB module enhances small defect detection by modeling global normality, while the LNP aids fine-grained reconstruction via pattern memory. Furthermore, integrating SSB and SOAIL modules boosts both image- and pixel-level performance, validating the effectiveness of our multi-branch design for multi-scale anomaly detection.

3) Performance across diverse defect types and industrial conditions: To provide a more comprehensive overview of the variety of defects our method can handle, we conducted a statistical analysis of the defect categories in the MVTec AD dataset and visualized the results as a histogram (Fig. 11). This visualization distinguishes between common defect types, such as color, scratch, crack, cut, and contamination, which frequently occur across various industrial scenarios, and special defect types, such as flip, squeeze, broken, oil, and fold, which are more specific to certain industrial domains.

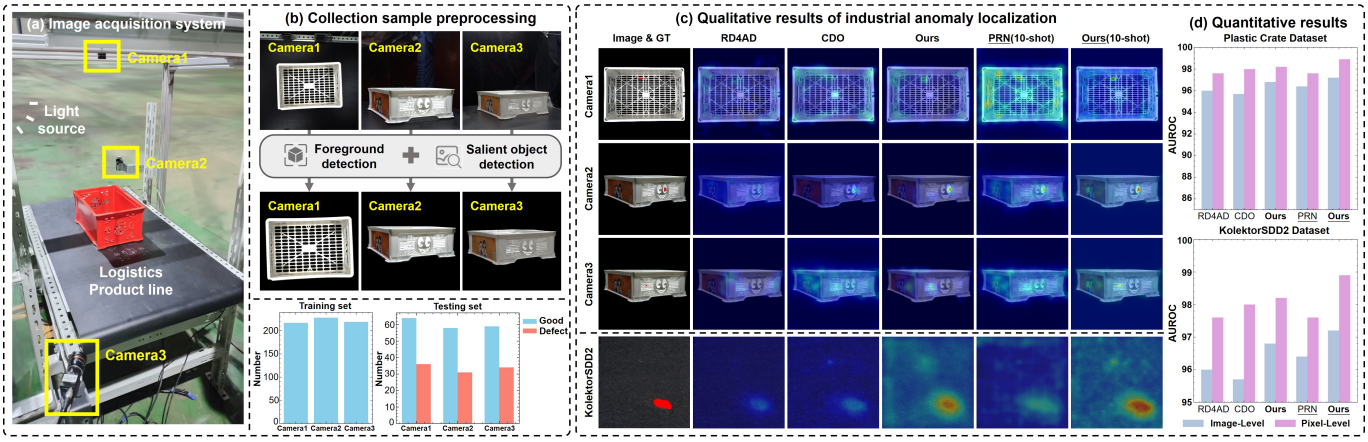


Fig. 13. Experimental validation in a real-world industrial environment under unsupervised and 10-shot semi-supervised scenarios. (a) A multi-view automatic image acquisition system installed on the logistics production line. (b) Image preprocessing workflow and the distribution of the plastic crate dataset. (c)-(d) Comparison of qualitative and quantitative results of different methods, with underlined approaches indicating those evaluated in the semi-supervised setting.

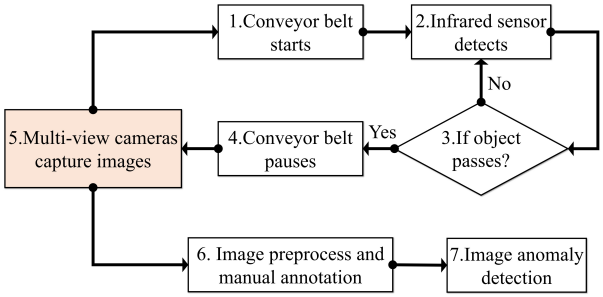


Fig. 14. Multi-view image capturing and processing system Flowchart.

TABLE VII
IMAGE-LEVEL / PIXEL-LEVEL AUROC RESULTS UNDER UNSUPERVISED AND SEMI-SUPERVISED (TEN ABNORMAL SAMPLES) SCENARIOS ON KOLEKTORSDD2 AND OUR REAL-WORLD BENCHMARKS

Category	Unsupervised scenario				Semi-supervised scenario	
	ATSN (TIM 24)	RD4AD (CVPR 22)	CDO (TII 23)	Ours	PRN (CVPR 23)	Ours*
Camera1	74.5/94.3	90.7/96.8	89.1/96.3	**91.5**/97.1	90.4/**97.2**	**91.8**/**97.5**
Camera2	76.8/95.1	87.7/96.0	86.2/97.1	**88.1**/**97.7**	87.6/96.9	**88.5**/**97.9**
Camera3	76.2/94.8	86.4/95.3	**88.9**/96.5	87.7/95.9	**91.2**/**97.0**	88.4/**96.7**
Average	75.8/94.7	88.3/96.0	88.1/96.6	89.1/96.9	**89.7**/**97.0**	**89.6**/**97.4**
KolektorSDD2	94.6/97.1	96.0/97.6	95.7/98.0	**96.8**/**98.2**	96.4/97.6	**97.2**/**98.9**

The prevalence of common defects demonstrates the strong generalization capability of our method across diverse real-world applications, while the inclusion of special defect types further highlights its adaptability to domain-specific industrial tasks. Fig. 6 further demonstrates anomaly localization on bottle, cable, and tire classes under varying lighting, reflecting typical industrial settings. These results confirm the robustness and adaptability of our method across different environments and applications.

F. Semi-supervised real world application

In this section, we present a real-world application of industrial plastic crate anomaly detection. First, we introduce a multi-view image acquisition and processing system. Then, we analyze the characteristics of the collected dataset. Finally, we evaluate the performance of our proposed RIAD method under both unsupervised and semi-supervised settings within this industrial environment.

1) Multi-view image acquisition setup: As illustrated in Fig. 12, the object (plastic crate) is placed on a conveyor belt that moves in a fixed direction. Three cameras (Camera1, Camera2, and Camera3) are strategically positioned around the object to capture comprehensive views from different angles. This configuration ensures full coverage of the crate's surfaces, allowing the system to detect anomalies that may not be visible from a single viewpoint.

2) Image capturing, processing, and annotation: Based on the schematic in Fig.12, we built a real-world industrial image acquisition system (Fig.13) and conducted data collection and annotation following the workflow in Fig.14. Specifically, objects are first detected by infrared sensors as they move along the conveyor belt. Upon detection, the belt pauses, and multi-view cameras capture images from different angles. As shown in the upper part of Fig. 13(b), the raw images captured from the production line contain cluttered backgrounds, including conveyor belt edges and other distracting elements. To mitigate the interference caused by such background noise, we preprocess the collected data using foreground detection [33] and salient object detection [38] methods. The resulting background-removed images are subsequently manually annotated and utilized for anomaly detection.

3) Collected dataset analysis: Following the steps described above, we collected a Multi-view Automated Plastic Crate Dataset (MVAPCD), which includes data from three different camera perspectives. Each view contains approximately 210 normal images for training, along with 60 normal and 35 anomalous images for testing. Compared to existing datasets such as MVTec AD, MVTec-3D AD, and BTAD, which primarily consist of small objects like capsules, screws, and carrots, the challenge posed by our plastic crate dataset lies in the relatively large product size (approximately 25cm*40cm*20cm) and the subtle nature of defect information, which results in a high rate of missed anomalies. Moreover, performance varies across views. As shown in Fig. 13(b), side views (Camera2 and Camera3) are more

challenging than the top view (Camera1) due to higher visual variability, structural complexity, and interference from printed labels. Side-view defects often appear along narrow edges or seams, making them harder to detect. Additionally, in practice, the conveyor's motion causes crates to shift forward after stopping, making objects appear larger in Camera2's view than in Camera3's, resulting in better performance for Camera2.

4) Performance analysis on real world application: Fig. 13 (c) qualitatively presents the anomaly localization results under unsupervised and semi-supervised scenarios. The localization results of RD4AD [15] and CDO [16] on the Plastic Crate Dataset and KolektorSDD2 dataset reveal that the detection task on the real dataset in this paper is more challenging. In comparison, our proposed RIAD method demonstrates the ability to localize various anomalies in the unsupervised scenario. In the semi-supervised scenario, we simulate human-machine interaction using 10 abnormal samples, where the PRN [19] method shows significantly better performance than unsupervised methods; however, it performs poorly on the Camera1 and KolektorSDD2 datasets. Owing to the SOAIL method proposed in this paper, the interaction with labeled abnormal samples enables more effective incremental learning, enhancing the model's fusion mechanism of S-T discrepancy maps and achieving more accurate localization results. Table VII and Fig. 13 (d) quantitatively illustrate the anomaly detection results of various methods under unsupervised and semi-supervised scenarios, demonstrating the efficiency of the proposed method in anomaly detection and localization across multiple datasets in both scenarios.

G. Limitation and future work

Despite the strong performance of the proposed RIAD method, certain limitations remain. In particular, the Logical Student Branch (LSB) and Structural Student Brankch (SSB) struggle to restore and detect extremely large or tiny defects due to two key factors. First, the ResNet backbone used for lightweight knowledge distillation primarily captures local features, limiting its ability to represent large-scale anomalies. Second, the low input resolution (256, 256) leads to the loss of fine-grained details, making tiny defects difficult to detect—as illustrated by the failure case in Fig. 7 and the camera3 example in Fig. 13 (c), where anomalies closely resemble the background. To address these challenges, we consider to integrate global-aware feature extractors such as Vision Transformers [61], Diffusion models [62], and few-shot large vision-language models [63] (e.g., CLIP) to better capture long-range dependencies, improve semantic understanding, and enhance reconstruction quality of normal patterns. In parallel, we plan to explore high-resolution detection strategies [64], including super-resolution, multi-scale analysis, and adaptive input resolutions, to boost sensitivity to subtle anomalies and further improve the method's practical applicability.

V. Conclusion

In conclusion, this paper has introduced the RIAD method, an efficient Knowledge distillation-based model applicable to unsupervised and semi-supervised human-machine interactive anomaly detection scenarios. By introducing style information for the first time in the knowledge distillation-based industrial anomaly detection field, RIAD significantly enhances the balance of generalization and robustness against out-of-distribution data via the bidirectional fusion of style information and content information. Extensive experiments on four widely used benchmark datasets validate the effectiveness of the proposed method and its components, showing strong performance across multi-size defects and various industrial conditions. At the same time, a real-world industrial application further validates its practical effectiveness and industrial applicability. Moving forward, we plan to further explore its applications in diverse industrial environments.

Acknowledgments: This work is supported by Shandong Natural Science Foundation of Outstanding Young Scientist Fund (ZR2024YQ071), National Natural Science Foundation of China (No.62172246), Youth Innovation and Technology Support Plan of Colleges and Universities in Shandong Province (2021KJ062), and Fundamental Research Funds for the Central Universities under the Youth Program (22CX06037A).

References

[1] A. G. Frank, L. S. Dalenogare, and N. F. Ayala, "Industry 4.0 technologies: Implementation patterns in manufacturing companies," *International journal of production economics*, vol. 210, pp. 15–26, 2019. 1

[2] H. Yang, Z. Zhu, C. Lin, W. Hui, S. Wang, and Y. Zhao, "Self-supervised surface defect localization via joint de-anomaly reconstruction and saliency-guided segmentation," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–10, 2023. 1

[3] S. Yuan, L. Li, H. Chen, and X. Li, "Surface defect detection of highly reflective leather based on dual-mask-guided deep-learning model," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–13, 2023. 1

[4] P. Li, F. Li, M. Liu, H. Bai, Y. Wei, A. Wang, S. Ma, and Y. Zhao, "Progressive complementary knowledge aggregation for cdznte defect segmentation," *IEEE Transactions on Industrial Informatics*, 2024. 1

[5] K. Shen, X. Zhou, and Z. Liu, "Minet: Multiscale interactive network for real-time salient object detection of strip steel surface defects," *IEEE Transactions on Industrial Informatics*, 2024. 1

[6] K. Roth, L. Pemula, J. Zepeda, B. Schölkopf, T. Brox, and P. Gehler, "Towards total recall in industrial anomaly detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022, pp. 14 318–14 328. 1, 3, 8

[7] Z. Liu, Y. Zhou, Y. Xu, and Z. Wang, "Simplenet: A simple network for image anomaly detection and localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 20 402–20 411. 1, 3, 8

[8] W. Liu, H. Chang, B. Ma, S. Shan, and X. Chen, "Diversity-measurable anomaly detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2023, pp. 12 147–12 156. 1, 3, 8

[9] N. Madan, N.-C. Ristea, R. T. Ionescu, K. Nasrollahi, F. S. Khan, T. B. Moeslund, and M. Shah, "Self-supervised masked convolutional transformer block for anomaly detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 1, pp. 525–542, 2024. 1, 3, 8

[10] R. Zhang, H. Wang, M. Feng, Y. Liu, and G. Yang, "Dual-constraint autoencoder and adaptive weighted similarity spatial attention for unsupervised anomaly detection," *IEEE Transactions on Industrial Informatics*, 2024. 1, 3

[11] Y. Liang, J. Zhang, S. Zhao, R. Wu, Y. Liu, and S. Pan, "Omni-frequency channel-selection representations for unsupervised anomaly detection," *IEEE Transactions on Image Processing*, 2023. 1, 3

[12] F. Lu, X. Yao, C.-W. Fu, and J. Jia, "Removing anomalies as noises for industrial defect localization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 16 166–16 175. 1, 3, 8

[13] M. Fučka, V. Zavrtanik, and D. Skočaj, "Transfusion–a transparency-based diffusion model for anomaly detection," in *European conference on computer vision*. Springer, 2025, pp. 91–108. 1, 3, 8

[14] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, "Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings," in *IEEE Conf. Comput. Vis. Pattern Recog.* IEEE, 2020, pp. 4182–4191. 1, 2, 3

[15] H. Deng and X. Li, "Anomaly detection via reverse distillation from one-class embedding," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022, pp. 9727–9736. 1, 2, 3, 4, 8, 13

[16] Y. Cao, X. Xu, Z. Liu, and W. Shen, "Collaborative discrepancy optimization for reliable image anomaly localization," *IEEE Transactions on Industrial Informatics*, pp. 1–10, 2023. 1, 3, 8, 13

[17] X. Tao, C. Adak, P.-J. Chun, S. Yan, and H. Liu, "Vitalnet: Anomaly on industrial textured surfaces with hybrid transformer," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–13, 2023. 1

[18] G. Pang, C. Shen, and A. Van Den Hengel, "Deep anomaly detection with deviation networks," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 353–362. 1, 3

[19] H. Zhang, Z. Wu, Z. Wang, Z. Chen, and Y.-G. Jiang, "Prototypical residual networks for anomaly detection and localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16 281–16 291. 1, 3, 6, 13

[20] Y. Cao, X. Xu, C. Sun, L. Gao, and W. Shen, "Bias: Incorporating biased knowledge to boost unsupervised image anomaly localization," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2024. 1, 3

[21] S. Yuan, L. Li, N. Yu, T. Peng, X. Hu, and X. Pan, "Anomaly detection of industrial products considering both texture and shape information," in *Computer Graphics International Conference*. Springer, 2023, pp. 149–160. 1

[22] P. Bergmann, K. Batzner, M. Fauser, D. Sattlegger, and C. Steger, "The mvtec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection," *International Journal of Computer Vision*, vol. 129, no. 4, pp. 1038–1059, 2021. 1, 6, 7, 8

[23] S. Akcay, D. Ameln, A. Vaidya, B. Lakshmanan, N. Ahuja, and U. Genc, "Anomalib: A deep learning library for anomaly detection," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 1706–1710. 1

[24] X. Zhang, S. Li, X. Li, P. Huang, J. Shan, and T. Chen, "Destseg: Segmentation guided denoising student-teacher for anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 3914–3923. 1, 3, 8

[25] Q. Zhou, S. He, H. Liu, T. Chen, and J. Chen, "Pull & push: Leveraging differential knowledge distillation for efficient unsupervised anomaly detection and localization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 5, pp. 2176–2189, 2023. 1, 3, 8

[26] T. D. Tien, A. T. Nguyen, N. H. Tran, T. D. Huy, S. Duong, C. D. T. Nguyen, and S. Q. Truong, "Revisiting reverse distillation for anomaly detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 24 511–24 520. 1, 3, 8

[27] J. Zhu, P. Yan, J. Jiang, Y. Cui, and X. Xu, "Asymmetric teacher–student feature pyramid matching for industrial anomaly detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–13, 2024. 1, 3, 8

[28] R. He, Z. Han, X. Lu, and Y. Yin, "Safe-student for safe deep semi-supervised learning with unseen-class unlabeled data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 585–14 594. 1

[29] R. Wu, S. Li, C. Chen, and A. Hao, "Improving video anomaly detection performance by mining useful data from unseen video frames," *Neurocomputing*, vol. 462, pp. 523–533, 2021. 1

[30] C. Wu, X. Liu, J. Wu, H. Zhang, and L. Wang, "Vertical-horizontal latent space with iterative memory review network for multi-class anomaly detection," *Knowledge-Based Systems*, vol. 292, p. 111594, 2024. 1

[31] Z. Gu, L. Liu, X. Chen, R. Yi, J. Zhang, Y. Wang, C. Wang, A. Shu, G. Jiang, and L. Ma, "Remembering normality: Memory-guided knowledge distillation for unsupervised anomaly detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 16 401–16 409. 2

[32] H. Guo, L. Ren, J. Fu, Y. Wang, Z. Zhang, C. Lan, H. Wang, and X. Hou, "Template-guided hierarchical feature restoration for anomaly detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 6447–6458. 2

[33] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 7464–7475. 3, 12

[34] Z. Yao, J.-N. Su, G. Fan, M. Gan, and C. P. Chen, "Gaca: A gradient-aware and contrastive-adaptive learning framework for low-light image enhancement," *IEEE Transactions on Instrumentation and Measurement*, 2024. 3

[35] L. Li, C. Chen, X. Yu, S. Pang, and H. Qin, "Siamtadt: A task-aware drone tracker for aerial autonomous vehicles," *IEEE Transactions on Vehicular Technology*, 2024. 3

[36] Z. Yao, G. Fan, J. Fan, M. Gan, and C. P. Chen, "Spatial-frequency dual-domain feature fusion network for low-light remote sensing image enhancement," *IEEE Transactions on Geoscience and Remote Sensing*, 2024. 3

[37] G. Shi, L. Li, and M. Song, "Beyond pixels: text-guided deep insights into graphic design image aesthetics," *Journal of Electronic Imaging*, vol. 33, no. 5, pp. 053 059–053 059, 2024. 3

[38] M. Song, W. Song, G. Yang, and C. Chen, "Improving rgb-d salient object detection via modality-aware decoder," *IEEE Transactions on Image Processing*, vol. 31, pp. 6124–6138, 2022. 3, 12

[39] M. Song, L. Li, D. Wu, W. Song, and C. Chen, "Rethinking object saliency ranking: A novel whole-flow processing paradigm," *IEEE Transactions on Image Processing*, 2023. 3

[40] Y. Ge, Q. Zhang, T.-Z. Xiang, C. Zhang, and H. Bi, "Tcnet: Co-salient object detection via parallel interaction of transformers and cnns," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 6, pp. 2600–2615, 2022. 3

[41] C. Chen, M. Song, W. Song, L. Guo, and M. Jian, "A comprehensive survey on video saliency detection with auditory information: the audio-visual consistency perceptual is the key!" *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 2, pp. 457–477, 2022. 3

[42] S. Zhang, M. Song, and L. Li, "Assisting rgb and depth salient object detection with nonconvolutional encoder: an improvement approach," *Journal of Electronic Imaging*, vol. 33, no. 2, pp. 023 036–023 036, 2024. 3

[43] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016. 4

[44] M.-M. Cheng, X.-C. Liu, J. Wang, S.-P. Lu, Y.-K. Lai, and P. L. Rosin, "Structure-preserving neural style transfer," *IEEE Transactions on Image Processing*, vol. 29, pp. 909–920, 2019. 4

[45] Y. Chen, Y. Wang, Y. Pan, T. Yao, X. Tian, and T. Mei, "A style and semantic memory mechanism for domain generalization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9164–9173. 4

[46] Y. Li, D. Zhang, M. Keuper, and A. Khoreva, "Intra-& extra-source exemplar-based style synthesis for improved domain generalization," *International Journal of Computer Vision*, vol. 132, no. 2, pp. 446–465, 2024. 4

[47] L. Xu, Q. Yan, Y. Xia, and J. Jia, "Structure extraction from texture via relative total variation," *ACM transactions on graphics (TOG)*, vol. 31, no. 6, pp. 1–10, 2012. 4

[48] M. Yang, P. Wu, and H. Feng, "Memseg: A semi-supervised method for image surface defect detection using differences and commonalities," *Engineering Applications of Artificial Intelligence*, vol. 119, p. 105835, 2023. 5

[49] A. Hatamizadeh, Z. Xu, D. Yang, W. Li, H. Roth, and D. Xu, "Unetformer: A unified vision transformer model and pre-training framework for 3d medical image segmentation," *arXiv preprint arXiv:2204.00631*, 2022. 5

[50] Y. Pu, Y. Wang, Z. Xia, Y. Han, Y. Wang, W. Gan, Z. Wang, S. Song, and G. Huang, "Adaptive rotated convolution for rotated object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 6589–6600. 5

[51] Z. Gu, J. Zhang, L. Liu, X. Chen, J. Peng, Z. Gan, G. Jiang, A. Shu, Y. Wang, and L. Ma, "Rethinking reverse distillation for multi-modal anomaly detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 8, 2024, pp. 8445–8453. 5, 6, 8

[52] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2414–2423. 6

[53] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 694–711. 6

[54] P. Mishra, R. Verk, D. Fornasier, C. Piciarelli, and G. L. Foresti, "Vt-adl: A vision transformer network for image anomaly detection and localization," in *International Symposium on Industrial Electronics*, 2021, pp. 01–06. 6, 7

[55] P. Bergmann, X. Jin, D. Sattlegger, and C. Steger, "The MVTec 3D-AD Dataset for Unsupervised 3D Anomaly Detection and Localization," *arXiv e-prints*, p. arXiv:2112.09045, 2021. 6, 7

[56] J. Božič, D. Tabernik, and D. Skočaj, "Mixed supervision for surface-defect detection: From weakly to fully supervised learning," *Computers in Industry*, vol. 129, p. 103459, 2021. 6, 7

[57] F. Zhang, H. Zhu, Y. Cen, S. Kan, L. Zhang, P. Vadakkepat, and T. H. Lee, "Low-shot unsupervised visual anomaly detection via sparse feature representation," *IEEE transactions on neural networks and learning systems*, 2024. 8

[58] Y. Zhou, Z. Huang, D. Zeng, Y. Qu, and Z. Wu, "Dual-branch knowledge distillation via residual features aggregation module for anomaly segmentation," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1–11, 2025. 8

[59] Y.-M. Chu, C. Liu, T.-I. Hsieh, H.-T. Chen, and T.-L. Liu, "Shape-guided dual-memory learning for 3d anomaly detection," in *Proceedings of the 40th International Conference on Machine Learning*, 2023, pp. 6185–6194. 8

[60] X. Zhang, M. Xu, and X. Zhou, "Realnet: A feature selection network with realistic synthetic anomaly for anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 699–16 708. 8

[61] H. Yao, Y. Cao, W. Luo, W. Zhang, W. Yu, and W. Shen, "Prior normality prompt transformer for multiclass industrial image anomaly detection," *IEEE Transactions on Industrial Informatics*, 2024. 13

[62] H. Yao, M. Liu, Z. Yin, Z. Yan, X. Hong, and W. Zuo, "Glad: towards better reconstruction with global and local adaptive diffusion models for unsupervised anomaly detection," in *European Conference on Computer Vision*. Springer, 2024, pp. 1–17. 13

[63] Y. Cao, J. Zhang, L. Frittoli, Y. Cheng, W. Shen, and G. Boracchi, "Adaclip: Adapting clip with hybrid learnable prompts for zero-shot anomaly detection," in *European Conference on Computer Vision*. Springer, 2024, pp. 55–72. 13

[64] Y. Cao, H. Yao, W. Luo, and W. Shen, "Varad: Lightweight high-resolution image anomaly detection via visual autoregressive modeling," *IEEE Transactions on Industrial Informatics*, 2025. 13