Fast Dynamic Prototypes for Unsupervised Anomaly Detection and Localization

Mingliang Li[1,3 †], Hanxi Li[1,3 †✉], Lin Yuanbo Wu[2 ✉], Changhong Liu[1], and Xiaowei Zhao[2]

[1] Jiangxi Normal University, China
[2] University of Warwick, Coventry, CV4 7AL, United Kingdom
[3] Yuefan Automation, China

Abstract. Unsupervised Anomaly Detection (UAD) excels in real-world applications especially with rare anomalies due to the elimination of labeled defective samples. Most existing UAD algorithms adopt a comparison process where the features of the test image are compared against their normal "prototypes". Such a paradigm relies primarily on raw features from pre-trained models and is inefficient due to the excessive number of prototypes. To enhance UAD performance, recent algorithms aim to establish a mapping from the test features to their respective prototypes . However, the proposed mapping models, typically based on distillation or attention mechanisms, perform the "reconstruction" implicitly, making the process unnecessarily complex and difficult to explain. In this paper, we propose a simple yet effective scheme, termed Fast Dynamic Prototype (FDP), to rapidly generate high-quality reconstructed prototypes for Multi-class UAD. First, a raw prototype bank is obtained via simply clustering the deep features of normal samples. Secondly, within the proposed Dynamic Bank Adapter module, the raw bank is dynamically condensed and adapted based on the information of the current test image. The reconstructed prototype of a given test feature is then defined as the weighted average of its nearest bank members. Finally, an auxiliary FDP utilizing a different backbone model is introduced to successfully mitigate the impact of artifacts present in the primary FDP. The study's extensive experiments demonstrate the superiority of the proposed algorithm, which outperforms all compared SOTA methods on three well-known UAD datasets (MVTec-AD, MVTec-3D, and ViSA), while attaining a speed up to 477 fps. Our implementation is publicly available at `https://github.com/lmlpy/FDP.git`.

Keywords: UAD · Fast Dynamic Prototype · Hybrid Residual

1 Introduction

Unsupervised anomaly detection (UAD) aims to identify abnormal patterns using only normal training data. It plays a crucial role in industrial inspection and

[1] † These authors contributed equally to this work.
[2] ✉ Corresponding authors.

safety-critical applications, where anomalous samples are scarce and expensive to annotate. While early studies focused on single-class settings, practical deployments increasingly demand a unified multi-class anomaly-detection framework with high efficiency and generalization.

Currently, the dominant paradigm for UAD is based on prototypes where a memory bank of normal features are constructed and then anomalies are detected via feature matching at test time. A representative method, PatchCore [21], directly stores dense raw patch features and performs nearest-neighbor matching. Although highly effective, this dense storage strategy introduces significant redundancy and computational overhead as the number of classes grows. Building on this design, WeakREST [17] and PRN [28] adopt a similar feature-matching procedure and improve localization by learning an advanced segmentation head. However, it still relies on large raw feature banks during inference.

Recent works instead focus on learning better prototype representations via feature reconstruction. UniAD [25] introduces a unified framework that learns refined prototypes for multi-class settings. More recently, Dinomaly [10] and INP-Former [19] employ transformer architectures and attention mechanisms to dynamically generate improved prototypes conditioned on test features. In particular, INPFormer [19] leverages information from the test image to further enhance prototype adaptation. While these transformer-based strategies improve representation quality, they introduce additional architectural complexity and computational cost, and prototype refinement is largely implicit through attention mapping.

To leverage the merits of both feature-matching and reconstruction-based strategies, we propose Fast Dynamic Prototype (FDP), an explicit and efficient framework for unsupervised anomaly detection. Rather than relying on dense storage or on transformer-based implicit reconstruction, FDP generates adaptive prototypes for test features in four clear, interpretable steps. We first construct a clustered raw prototype bank to eliminate redundancy while preserving representative diversity. A lightweight Dynamic Bank Adapter (DBA) then dynamically refines the bank based on information from the current test image. Thirdly, a weighted nearest-neighbor reconstruction produces the prototypes for the test features. Fourthly, to mitigate artifacts inherited from the self-taught backbone model, we incorporate an auxiliary bank based on a supervised backbone model, and the auxiliary FDPs are then fused with the primary FDPs using zero-convolution. As illustrated in Fig. 1, the proposed FDP lies between the two popular UAD methodologies and inherits the advantages of both. In addition to the advanced FDP, this work proposes a novel hybrid residual calculation method to improve the accuracy of anomaly detection and localization with minimal computational overhead.

Extensive experiments on three widely used anomaly detection benchmarks, *i.e.*, MVTec-AD, MVTec-3D, and VisA, demonstrate that FDP achieves state-of-the-art detection performance while maintaining substantially faster inference speed than competing UAD approaches. These results verify that explicit dynamic prototype reconstruction offers a simple, interpretable, and computation-

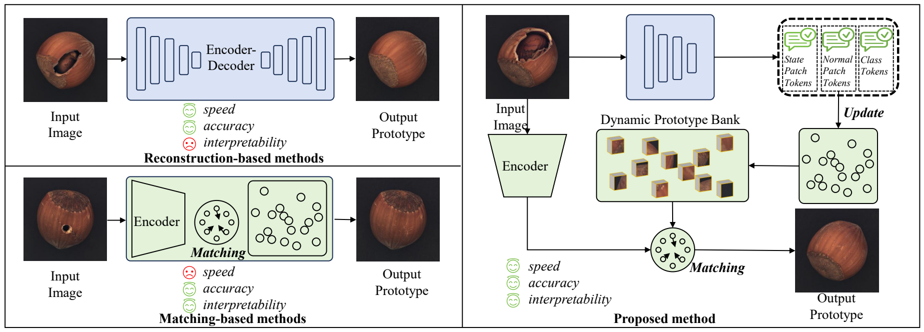

Fig. 1: The conceptual illustration of the proposed method. One can see that our algorithm inherits the merits from both reconstruction-based methods and the matching-based approaches. The prototype bank is adjusted by the compensation factor predicted based on a learned encoder and the current test image.

ally efficient alternative to dense memory banks and complex attention-based prototype learning strategies. In summary, our contributions are fourfold:

- We propose Fast Dynamic Prototype (FDP), an explicit, adaptive prototype-generation framework for efficient unsupervised anomaly detection.
- We develop a lightweight Dynamic Bank Adapter coupled with a complementary prototype reconstruction strategy to generate high-quality prototype representations for test features with minimal computational overhead.
- We design a simple yet effective residual-learning scheme that combines a novel hybrid residual with an embarrassingly simple segmentation head.
- We achieve state-of-the-art performance while significantly improving inference efficiency across multiple benchmark datasets.

2 Related Work

Two types of UAD algorithms Despite being grounded in diverse concepts from computer vision and machine learning, UAD algorithms share a common comparison process in which test features are matched against their normal "prototypes." Based on how this comparison is conducted, existing approaches can be broadly categorized into two types.

The first category comprises matching-based methods, which leverage a pre-trained encoder to extract raw feature representations and detect anomalies by measuring the distance between each test feature and its nearest neighbor in a precomputed bank. Representative works include PatchCore [21], PRN [28], and WeakREST [17]. With the exception of the pioneering PatchCore framework, most subsequent methods incorporate complex segmentation heads that operate on matching residuals to enhance detection performance. Despite their relatively conventional design, matching-based methods continue to achieve state-of-the-art results [17], albeit at the cost of high computational overhead from exhaustive memory bank comparisons.

The second category, reconstruction-based methods [1], takes a different approach by employing encoder-decoder architectures to dynamically generate normal prototypes for test features. In this paradigm, prototypes are not retrieved from a fixed bank but are instead rendered on-the-fly by a parameterized model, with anomalies identified through reconstruction errors. Notable examples include DesTSeg [30], RealNet [29], Dinomaly [10], INP-former [19], and the recently proposed DictAS [20]. By avoiding nearest-neighbor searches in large memory banks, reconstruction-based methods typically offer faster inference. However, they often suffer from implicit, overly complex reconstruction objectives that can hinder interpretability and training stability.

In this work, we strike a balance between these two paradigms. Our approach maintains a pre-determined prototype bank but introduces a dynamic compensation mechanism that updates the bank during inference based on the test image. This design inherits the efficiency of reconstruction-based methods while preserving the explicitness and robustness of matching-based strategies. As a result, we achieve superior anomaly detection performance with significantly reduced complexity.

Pre-trained Encoder Almost all UAD methods, including OneNIP [8], FoundAD [27], and UniADet [9], employ a pretrained backbone for raw feature extraction, and the employed models can be classified into three types: the conventional supervised models, such as ResNet [13] and DenseNet [14]; the recently emerged self-taught models like the DINO series [22]; and finally the AD-oriented models that are tailored specifically for UAD tasks [24,32]. In general, self-taught models achieve higher UAD performance than supervised models, especially when no segmentation head is used [10,19]. However, as reported in many studies [6,22], the "artifact" arising from the intrinsic attention nature of transformer models harms pixel-wise predictions, potentially leading to inferior UAD performance in certain cases. In this work, to address this issue, we propose to use both the supervised and self-taught backbones to compose the "complementary bank pair" and the final prototype is fused from the two sources.

3 Methods

3.1 Problem Definition

Given a training set $\mathcal{D}_{\text{train}} = \{\mathcal{X}^i_{\text{train}}\}_{i=1}^{C}$ comprising C categories of normal industrial products, where each category $\mathcal{X}^i_{\text{train}} = \{\mathbf{x}_j\}_{j=1}^{N_i}$ contains only anomaly-free samples, the objective is to learn a unified model $f : \mathbf{x} \to (\mathbf{s}, \mathbf{m})$ without any exposure to anomaly samples that can handle all categories simultaneously. For any test sample $\mathbf{x} \in \mathcal{X}_{\text{test}}$, the model should predict an image-level anomaly score $\mathbf{s} \in [0, 1]$ and a pixel-wise anomaly heatmap $\mathbf{m} \in \mathbb{R}^{\mathrm{h}\times\mathrm{w}}$ to accurately identify and localize anomalies, while maintaining cross-category compatibility and real-time efficiency.

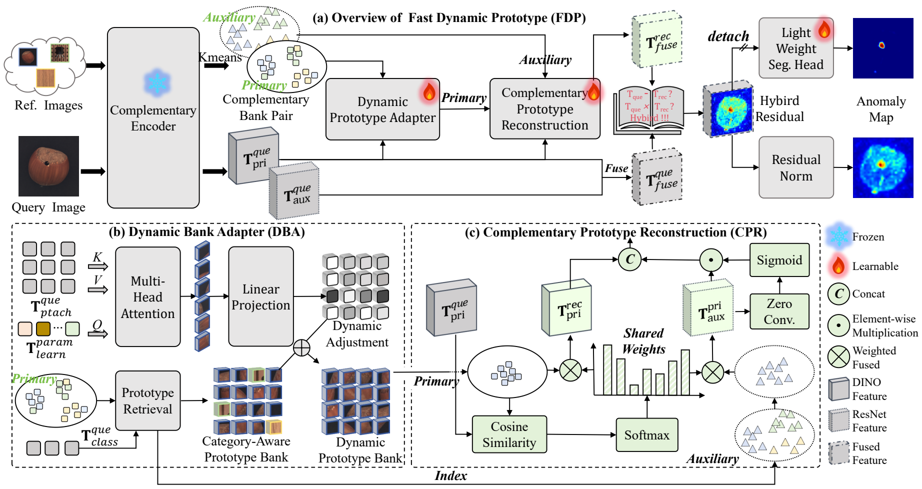

Fig. 2: The main workflow of the proposed method is shown in (a); section (b) illustrates the details of the dynamic prototype adapter, which includes prototype retrieval and dynamic refinement; (c) demonstrates the Complementary Prototype Reconstruction (CPR) module; the HRL module is shown in the right end of (a).

3.2 Overview

Our method comprises three key components: the Dynamic Prototype Adapter (DPA), the Complementary Prototype Reconstruction (CPR), and the Hybrid Residual Learning (HRL). The model's structure is depicted in Fig. 2.

To begin with, given a normal reference set $\mathcal{D}_{\text{ref}}$ and a query image $\text{I}_{\text{que}} \in \mathbb{R}^{\text{h}\times\text{w}\times 3}$, we employ two complementary encoders to extract the primary patch tokens T_{pri}, the auxiliary patch tokens T_{aux} and the class tokens T_{cls} based on which we build Primary Prototype Bank $\mathcal{B}_{\text{pri}}$ and its index-aligned Auxiliary Prototype Bank $\mathcal{B}_{\text{aux}}$ via K-Means clustering. Subsequently, the primary patch tokens $\text{T}_{\text{pri}}^{que}$ and class tokens $\text{T}_{\text{cls}}^{que}$ from the query image are then fed into the Dynamic Prototype Adapter (DPA), which dynamically extracts query-relevant information to modify the Primary Prototype Bank $\mathcal{B}_{\text{pri}}$. This leads to a compact yet discriminative Dynamic Prototype Bank $\mathcal{B}_{\text{dyn}}$. The Complementary Prototype Reconstruction (CPR) module is then fed with $\text{T}_{\text{pri}}^{que}$, $\mathcal{B}_{\text{dyn}}$ and $\mathcal{B}_{\text{aux}}$ and output the reconstructed primary tokens $\text{T}_{\text{pri}}^{rec}$ and the auxiliary tokens $\text{T}_{\text{aux}}^{rec}$. Next, the primary and auxiliary tokens (for both the reconstructed ones and the query ones) are fused in a zero-convolution manner, and the outcomes are denoted as $\text{T}_{\text{fuse}}^{que}$ and $\text{T}_{\text{fuse}}^{rec}$. Finally, the Hybrid Residual $\mathcal{R}_{\text{hyb}}$ is calculated based on the various discrepancies between $\text{T}_{\text{fuse}}^{que}$ and $\text{T}_{\text{fuse}}^{rec}$, and the pixel-level anomaly predictions $\mathbf{m}^{pred}$ can be obtained via the lightweight segmentation head.

3.3 Dynamic Prototype Adapter

During the inference stage, the query's patch tokens $\mathrm{T}_{\text{pri}}^{que} \in \mathbb{R}^{N\times C}$ (primary), $\mathrm{T}_{\text{aux}}^{que} \in \mathbb{R}^{N\times C}$ (auxiliary), and $\mathrm{T}_{\text{cls}}^{que} \in \mathbb{R}^{1\times C}$ (class) are obtained by using the two complementary encoders. The Complementary Prototype Bank $\mathcal{B}_{\text{cmp}} = \{\mathcal{B}_{\text{cls}},\ \mathcal{B}_{\text{pri}},\ \mathcal{B}_{\text{aux}}\}$ is also pre-generated. Here, $\mathcal{B}_{\text{cls}} \in \mathbb{R}^{M\times C}$, $\mathcal{B}_{\text{pri}} \in \mathbb{R}^{M\times C}$, $\mathcal{B}_{\text{aux}} \in \mathbb{R}^{M\times C}$ denote the class prototype bank, primary prototype bank, and auxiliary prototype bank, respectively. Taking these as input, DPA produces a compact Dynamic Prototype Bank $\mathcal{B}_{\text{dyn}} \in \mathbb{R}^{K\times C}$ where $K \ll M$ through the following two steps.

Step 1: Category-Aware Prototype Retrieval. The cosine similarity between the query's class token $\mathrm{T}_{\text{cls}}^{que}$ and all reference tokens in $\mathcal{B}_{\text{cls}}$ is calculated as

$$s_i = \frac{\mathrm{T}_{\text{cls}}^{que} \cdot \mathcal{B}_{\text{cls}}^{i}}{\|\mathrm{T}_{\text{cls}}^{que}\|\|\mathcal{B}_{\text{cls}}^{i}\|}, \qquad i = 1, 2, \ldots, M \tag{1}$$

where s_i denotes the similarity score of the i-th prototype. We then select the top-K indices with the highest scores as Eq. 2.

$$\mathrm{I}_{\text{idx}} = \text{top-}K(\{s_i\}_{i=1}^{M}), \qquad \mathcal{B}_{\text{ca}} = \mathcal{B}_{\text{pri}}[\mathrm{I}_{\text{idx}}] \in \mathbb{R}^{K\times C}. \tag{2}$$

Where $\mathcal{B}_{\text{ca}}$ contains K primary prototypes most relevant to the query's semantic category, the indices I_{idx} are used to select the corresponding primary prototypes from $\mathcal{B}_{\text{pri}}$ to form the Category-Aware Prototype Bank $\mathcal{B}_{\text{ca}}$.

Step 2: Adaptive Dynamic Refinement. To exploit the useful information from the test image, DPA introduces a multi-head attention process. Specifically, a set of learnable tokens $\mathrm{T}_{\text{learn}} \in \mathbb{R}^{L\times C}$ serves as queries (Q), the query's primary patch tokens $\mathrm{T}_{\text{pri}}^{que}$ act as keys (K) and values (V), enabling the extraction of dynamic information conditioned on the query image as

$$\mathrm{F}_{\text{dyn}} = \mathrm{MHA}(\mathrm{T}_{\text{learn}}, \mathrm{T}_{\text{pri}}^{que}, \mathrm{T}_{\text{pri}}^{que}) \in \mathbb{R}^{L\times C}, \tag{3}$$

where MHA denotes multi-head attention and F_{dyn} encodes the query-specific information and is then passed through a linear projection layer to compute the dynamic compensation matrix, and the original bank $\mathcal{B}_{\text{ca}}$ is adjusted via element-wise addition, which writes:

$$\Delta\mathcal{B} = \mathrm{Linear}(\mathrm{F}_{\text{dyn}}^{\mathsf{T}})^{\mathsf{T}} \in \mathbb{R}^{K\times C}, \qquad \mathcal{B}_{\text{dyn}} = \mathcal{B}_{\text{ca}} + \Delta\mathcal{B}. \tag{4}$$

3.4 Complementary Prototype Reconstruction

Observed that the self-taught (DINOv3) prototype bank introduces some artifacts, as evidenced in Fig. 4, we design the Complementary Prototype Reconstruction (CPR) module that leverages both self-taught and supervised backbone models.

Firstly, given the query primary patch tokens $\mathrm{T}_{\mathrm{pri}}^{que}$ and the dynamic prototype bank $\mathcal{B}_{\mathrm{dyn}}$, the cosine distance between each query token and each prototype is computed as:

$$d_{i,j} = 1 - \frac{\mathrm{T}_{\mathrm{pri}}^{que}[i] \cdot \mathcal{B}_{\mathrm{dyn}}[j]}{\|\mathrm{T}_{\mathrm{pri}}^{que}[i]\| \|\mathcal{B}_{\mathrm{dyn}}[j]\|}, \tag{5}$$

where $d_{i,j}$ denotes the distance between the i-th query token and j-th dynamic prototype. The reconstruction weights are then obtained by taking the reciprocal of distances, followed by softmax normalization:

$$w_{i,j} = \frac{\exp(1/d_{i,j})}{\sum_{k=1}^{K} \exp(1/d_{i,k})}, \tag{6}$$

where $\mathrm{W} \in \mathbb{R}^{N \times K}$ denotes the yielded weight matrix. The reconstructed primary tokens are then computed as a weighted sum of the dynamic prototypes as $\mathrm{T}_{\mathrm{pri}}^{rec} = \mathrm{W}\mathcal{B}_{\mathrm{dyn}} \in \mathbb{R}^{N \times C}$.

Second, the reconstruction is also conducted for the auxiliary query tokens. To reduce the model complexity, we reuse the weights W from the primary reconstruction. Formally, suppose that the members in the primary bank $\mathcal{B}_{\mathrm{pri}}$ and auxiliary bank $\mathcal{B}_{\mathrm{aux}}$ are index-aligned, we first select the category-aware auxiliary bank as:

$$\mathcal{B}_{\text{aux-ca}} = \mathcal{B}_{\mathrm{aux}}[\mathrm{I}_{\mathrm{idx}}] \in \mathbb{R}^{K \times C}, \tag{7}$$

where $\mathcal{B}_{\mathrm{aux}} \in \mathbb{R}^{M \times C}$ and $\mathrm{I}_{\mathrm{idx}}$ is defined in Eq. 2. The reconstructed auxiliary tokens are obtained via the same shared weights: $\mathrm{T}_{\mathrm{aux}}^{rec} = \mathrm{W}\mathcal{B}_{\text{aux-ca}} \in \mathbb{R}^{N \times C}$.

Ultimately, we fuse the reconstructed primary and auxiliary tokens through a lightweight zero-convolution module:

$$\mathrm{T}_{\mathrm{fuse}}^{rec} = \mathrm{T}_{\mathrm{pri}}^{rec} \oplus (\mathrm{T}_{\mathrm{aux}}^{rec} \odot Conv_{\mathrm{zero}}(\mathrm{T}_{\mathrm{aux}}^{rec}).\sigma()) \in \mathbb{R}^{N \times c}, \tag{8}$$

where $\oplus$ denotes channel-wise concatenation and $Conv_{\mathrm{zero}}$ is a zero-convolution layer that adaptively introduces auxiliary tokens. The fused tokens $\mathrm{T}_{\mathrm{fuse}}^{rec}$ are the final prediction of the prototype in this work.

3.5 Hybrid Residual Learning

To calculate the counterpart tokens for $\mathrm{T}_{\mathrm{fuse}}^{rec}$, $\mathrm{T}_{\mathrm{fuse}}^{que}$ is calculated via fusing the primary query tokens $\mathrm{T}_{\mathrm{pri}}^{que}$ and auxiliary query tokens $\mathrm{T}_{\mathrm{aux}}^{que}$, by using the identical mechanism defined in Eq. 8. These tokens are then reshaped back to the spatial feature tensors and apply L2 normalization along the channel dimension, leading to the fused query features $\mathrm{F}_{\mathrm{fuse}}^{que} \in \mathbb{R}^{h \times w \times c}$ and fused reconstructed features $\mathrm{F}_{\mathrm{fuse}}^{rec} \in \mathbb{R}^{h \times w \times c}$. We then compute the conventional cosine residual as:

$$\mathcal{R}_{\cos} = -\mathrm{F}_{\mathrm{fuse}}^{que} \odot \mathrm{F}_{\mathrm{fuse}}^{rec} \in \mathbb{R}^{h \times w \times c}, \tag{9}$$

where $\odot$ denotes element-wise multiplication. Conventionally, the anomaly map can be directly obtained via channel-wise summation:

$$\mathbf{m}_{i,j}^{norm} = \Sigma_{k=1}^{c} \mathcal{R}_{\cos}^{i,j}[k]. \tag{10}$$

When residual learning is adopted for better performance, $\mathcal{R}_{\cos}$ can be fed into a lightweight segmentation head $\mathcal{H}_{\text{seg}}$ to predict the anomaly map. However, due to the multiplication nature, the conventional cosine residual $\mathcal{R}_{\cos}$ tends to retain substantial object structure information that distracts the segmentation head from focusing on genuine anomalies (see Fig. 5). Conversely, the L1 residual defined as:

$$\mathcal{R}_{\text{L1}} = \|\mathrm{F}_{\text{fuse}}^{que} - \mathrm{F}_{\text{fuse}}^{rec}\|_{\text{L1}} \tag{11}$$

exhibits superior capability in highlighting anomalous regions through its subtractive paradigm. Nevertheless, computing L1 residual brings significant computational overhead, limiting its practical applicability. Compared with L1 residual, cosine residual enjoys a higher computational speed (Table VI, supplement), thanks to its highly optimized CUDA implementations of inner-product computation.

In this work, we propose a hybrid residual $\mathcal{R}_{\text{hyb}} \in \mathbb{R}^{h \times w \times c}$ that synergistically combines the complementary strengths of both multiplicative and subtractive paradigms. Following Eq. 12, our hybrid residual is computed as:

$$\begin{aligned}
\mathcal{R}_{\text{hyb}} &= \mathcal{R}_{\cos} - \mathcal{R}_{\cos} \odot \mathrm{S} \\
&= \left(-\mathrm{F}_{\text{fuse}}^{que} \odot \mathrm{F}_{\text{fuse}}^{rec}\right) - \left(-\mathrm{F}_{\text{fuse}}^{que} \odot \mathrm{F}_{\text{fuse}}^{rec}\right) \odot \mathrm{S} \\
&= \left(\mathrm{S} - \mathbf{1}\right) \odot \left(\mathrm{F}_{\text{fuse}}^{que} \odot \mathrm{F}_{\text{fuse}}^{rec}\right).
\end{aligned} \tag{12}$$

where $\mathrm{S} \in \mathbb{R}^{h \times w \times c}$ denotes the channel-wise replicated cosine similarity map between the query token and reconstructed tokens at corresponding spatial locations, expanded to match the number of channels of $\mathcal{R}_{\text{hyb}}$. Finally, the hybrid residual $\mathcal{R}_{hyb}$ is then passed to the lightweight segmentation head $\mathcal{H}_{\text{seg}}$ to generate the final anomaly map $\mathbf{m}^{seg}$, as shown in Fig. 3.

3.6 Training strategy

The training pipeline comprises two sequential phases, *i.e.*, the Dynamic Prototype Learning and the Hybrid Residual Learning. In the first phase, we only use a number of normal samples to train, with the loss of cosine similarity denoted as $\mathcal{L}_{\cos}$ as calculated in Eq. 13.

$$\mathcal{L}_{\cos} = \frac{1}{N} \sum_{i=1}^{N} \left(1 - \frac{\mathbf{T}_{\text{fuse}}^{que}[i] \cdot \mathbf{T}_{\text{fuse}}^{rec}[i]}{\|\mathbf{T}_{\text{fuse}}^{que}[i]\| \|\mathbf{T}_{\text{fuse}}^{rec}[i]\|}\right). \tag{13}$$

Subsequently, the trained model without segmentation head $\mathcal{H}_{\text{seg}}$ produce high-quality query fused tokens $\mathrm{T}_{\text{fuse}}^{que}$ and reconstructed fuse tokens $\mathrm{T}_{\text{fuse}}^{rec}$. Then

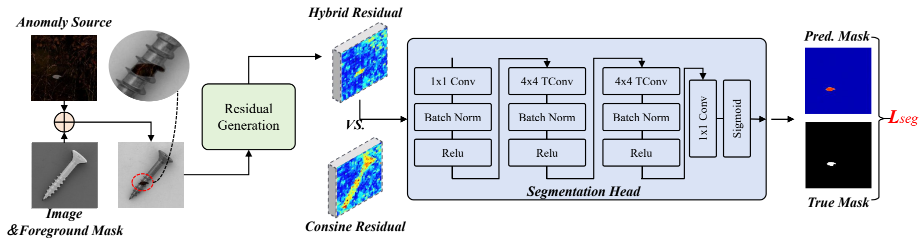

Fig. 3: The overview of residual learning. It can be seen that the Hybrid Residual shows better normal-abnormal distinction than the traditional cosine residual. The foreground mask is predicted by a pretrained BiRefNet [31].

calculate hybrid residual $\mathcal{R}_{\text{hyd}}$ as shown in Eq. 12, which is then gradient-clipped and fed into the segmentation head for the second phase. In the second phase, we generate synthetic anomaly samples as supplementary training data and adopt a composite loss function $\mathcal{L}_{\text{seg}} = \mathcal{L}_1 + \mathcal{L}_{\text{focal}}$ with the two sub-losses defined as:

$$\mathcal{L}_1 = \|\mathbf{m}^{seg} - \mathbf{m}^{\text{gt}}\|_1, \qquad \mathcal{L}_{\text{focal}} = \alpha_t \cdot \text{BCE}(\mathbf{m}^{seg},\ \mathbf{m}^{\text{gt}}) \cdot (1 - p_t)^{\gamma}, \tag{14}$$

where $\mathbf{m}^{gt}$ is the true mask of the synthetic anomaly sample, and p_t is the predicted probability for the positive class.

4 Experiments

Experimental Setting To evaluate the performance of the proposed method, we conducted experiments on four datasets: MVTec-AD [2], MVTec-3D [3], VisA [34], and Real-IAD [23], under both multi-class, single-class and cross-class experimental settings. During the training phase 1 of the FDP, we only considered the normal samples from the training set. Since the proposed FDP involves hybrid residual learning, following DRAEM [26], we used the DTD dataset [5] to create artificially anomaly images. In terms of evaluation metrics, we used P-AP, P-AUROC, and P-PRO to assess the model's pixel-level performance and I-AUROC to evaluate its image-level performance. Furthermore, we assessed the model's efficiency using Parameters, FLOPs, and FPS.

Competitors and Baseline We compared the proposed method with several UAD methods, including Patchcore [21], UniAD [25], DiAD [12], SimpleNet [18], Dinomaly [10], INP-Former [19], CPR [16], DeSTSeg [30], WeakREST [17]. We designed the baseline to directly perform nearest neighbor matching after constructing a prototype bank using the $\mathcal{D}_{\text{ref}}$. Then, the cosine distance between the matching tokens and the tokens to be detected is calculated as the final predicted $\mathbf{m}$, and the subsequent image prediction label $\mathbf{s}$ is calculated as same as the method we proposed.

Implementation Details In our FDP framework, DINOv3-small and ResNet-18 are adopted as the primary and auxiliary backbones, respectively. For data preprocessing, all input images are first resized to 512×512, then center-cropped to 448×448, and finally resized back to 512×512 using bilinear interpolation. For the hyperparameter of loss, the values of α and γ are -1 and 4, respectively. The model is optimized using AdamW with a batch size of 16. The training process is divided into two stages: an initial stage with a learning rate of 10^{-4} for 1000 iterations, and a second stage with a learning rate of 5×10^{-4} for 5000 iterations. A cosine annealing scheduler is employed in both stages to gradually reduce the learning rate to one-tenth of its initial value. The prototype bank is configured with a size of 8192. During evaluation, both ground-truth and predicted masks are resized to a uniform resolution of 256×256 prior to metric computation.

4.1 Main Results

The multi-class results are reported in Table 1. The proposed method FDP achieves the best performance on both image-level and pixel-level. In MVTec-AD dataset, FDP outperforms the state-of-the-art methods [10, 17, 19] by achieving **99.7/79.2/96.7** with respect to (I-AUROC/P-AP/P-PRO). Similarly, FDP obtains competitive detection accuracy in the other two datasets, and the **Average** value (calculated as the average of all 12 metrics) reaches **89.7**. In addition, we replace the backbone of some competitors to DINOv3 for the sake of in-depth comparative experiments. The experimental results are shown in Table 5.

We hypothesize that leveraging information from the test images to adaptively optimize the prototype bank can also improve performance in the single-class setting. To validate this hypothesis, we conduct experiments under the single-class setting, and the results are reported in Table 2. Our method achieves strong performance across three datasets, obtaining an average I-AUROC of **97.7**, P-AP of **70.9**, P-PRO of **96.6**, and P-AUROC of **98.8**. Furthermore, we evaluate the proposed method under the cross-class setting. Specifically, the model is trained on all categories except category c, and then directly tested on category c without retraining. The experimental results are presented in Table 3. FDP consistently outperforms competing methods, achieving P-AP scores of **79.5/67.4/56.1** and I-AUROC scores of **99.6/90.4/96.6** across the three datasets. These results demonstrate that the proposed method remains highly competitive even in the challenging cross-class setting.

4.2 Efficiency Evaluation

We conducted an efficiency comparison with several methods, and the results are presented in Table 4. As shown, our method achieves state-of-the-art performance with an **mAD** of **93.5** while also significantly improving inference efficiency, reaching **192 FPS**. Compared with INP-Former [19], the proposed FDP is nearly **5**$\times$ faster and requires only **30%** of its parameters. Comparing with the traditional embedding-based method PatchCore (13 FPS) [21], our approach achieves an **over 10**$\times$ speedup.

Table 1: Performance comparison of anomaly detection methods across three benchmark datasets in the **multi-class** setting. The best and second-best results are highlighted in red and blue, respectively. **Average(%)** denotes the average of all metrics across three datasets.

Method	MVTec-AD				MVTec-3D				Real-IAD				Average
	I-AUROC	P-AP	P-PRO	P-AUROC	I-AUROC	P-AP	P-PRO	P-AUROC	I-AUROC	P-AP	P-PRO	P-AUROC	
RD4AD [33]	94.6	48.6	91.1	96.1	77.9	29.8	93.5	98.4	82.4	25.0	89.6	97.3	77.0
UniAD [25]	96.5	43.4	90.7	96.8	78.9	21.2	88.1	96.5	83.0	21.1	86.7	97.3	75.0
SimpleNet [18]	95.3	45.9	86.5	96.9	72.5	18.3	77.6	93.5	57.2	2.8	39.0	75.7	63.4
DeSTSeg [30]	89.2	54.3	64.8	93.1	79.6	38.1	46.4	95.1	82.3	37.9	40.6	94.6	68.0
DiAD [12]	97.2	52.6	90.7	96.8	84.6	25.3	87.8	96.4	75.6	2.9	58.1	88.0	71.3
MambaAD [11]	98.6	56.3	93.1	97.7	86.2	37.5	93.6	98.6	86.3	33.0	90.5	98.5	80.8
CPR [16]	95.7	63.3	93.1	97.2	80.9	37.6	95.2	98.4	–	–	–	–	–
Dinomaly [10]	99.6	68.7	94.7	98.3	90.6	55.0	96.5	99.2	89.4	39.8	93.9	99.0	85.4
INP-Former [19]	99.7	71.0	94.9	98.5	92.6	55.9	97.0	99.2	90.5	44.2	94.9	99.1	86.5
WeakREST [17]	98.5	77.1	95.4	98.3	83.8	48.8	95.4	98.6	91.7	55.6	93.5	98.8	86.7
FDP	99.7	79.1	96.6	98.4	91.9	70.0	96.8	98.9	91.8	60.6	94.1	98.1	89.7

Average is computed as the average of all 12 metrics (4 metrics × 3 datasets). The (-) denotes missing values of some methods.

Table 2: Performance comparison of anomaly detection methods across three benchmark datasets in the **single-class** setting.

Method	MVTec-AD				MVTec-3D				VisA				Average
	I-AUROC	P-AP	P-PRO	P-AUROC	I-AUROC	P-AP	P-PRO	P-AUROC	I-AUROC	P-AP	P-PRO	P-AUROC	
RD4AD [33]	98.0	58.0	93.9	97.8	–	–	–	–	96.0	27.7	70.9	–	–
SimpleNet [18]	99.6	54.8	90.0	97.7	–	–	–	–	96.8	36.3	88.7	–	–
Dinomaly [10]	99.7	68.9	95.0	–	–	–	–	–	98.9	50.7	95.1	–	–
INP-Former [19]	99.7	70.2	95.4	98.3	93.0	56.6	97.0	99.2	98.5	49.2	93.8	98.4	87.4
FDP	99.8	81.5	97.4	98.7	94.6	71.5	97.3	99.1	98.7	59.6	95.0	98.6	91.0

Furthermore, we replaced the backbones of several state-of-the-art methods and conducted additional evaluations. Specifically, we adopted DINOv3-small and DINOv3-base as the common backbones and selected Dinomaly [10] and INP-Former [19] as representative comparison methods. The results in Table 5 show that, under both DINOv3-small and DINOv3-base [22], our method achieves the best mAD scores of **93.5** and **93.6**, respectively.

4.3 Ablation Study

To validate the effectiveness of each module, we conducted a systematic ablation study, with the results shown in Table 6. Baseline (10^5), which uses a large-scale prototype bank, achieved an mAD of **90.9**. In contrast, significantly reducing the bank size for Baseline(8192) led to a sharp deterioration in detection performance (I-AUROC: 98.8, P-AP: 65.4, mAD: 87.4). Based on this, the proposed modules were gradually introduced. Firstly, DPA improved I-AUROC, P-AP, and mAD by **0.8**, **7.1**, and **3.9**, respectively. Secondly, CPR further optimized pixel-level detection, with P-AP reaching **73.0**. Thirdly, followed by the enhancement of segmentation performance through Residual learning using a simple segmentation head. Finally, with the addition of HRL to form the complete FDP, the mAD reached the best performance of **93.5**, achieving a significant improve-

Table 3: Performance comparison of anomaly detection methods across three benchmark datasets in the **cross-class** setting.

Method	**MVTec-AD**				**MVTec-3D**				**VisA**				**Average**
	I-AUROC	P-AP	P-PRO	P-AUROC	I-AUROC	P-AP	P-PRO	P-AUROC	I-AUROC	P-AP	P-PRO	P-AUROC	
PaDiM [7]	95.5	53.6	92.2	97.3	76.6	23.7	91.0	97.3	89.9	31.3	83.6	97.6	77.5
PatchCore [21]	99.1	57.1	93.1	98.1	82.5	26.3	89.1	97.0	90.4	39.5	86.2	97.8	79.7
CPR [16]	92.8	61.4	91.4	96.8	71.3	23.6	87.6	96.3	87.8	30.8	77.5	92.7	75.8
HETMM [4]	99.2	68.0	95.8	98.6	83.4	37.7	94.8	98.4	89.0	38.4	89.1	96.3	82.4
Dinomaly [10]	63.6	15.2	39.5	73.8	46.9	4.2	59.5	87.8	59.5	9.8	51.5	86.3	49.8
FDP	99.6	79.5	96.9	98.5	90.4	67.4	95.8	98.6	96.6	56.1	93.5	98.4	89.3

Table 4: Efficienc comparison of anomaly detection methods. Metrics include model parameters (M), FLOPs (G), inference speed (FPS) and mean Anomaly Detection Score (mAD) on MVTec-AD. All experiments about inference speed were conducted on an NVIDIA RTX 4090 GPU with 24GB memory.

Method	**Efficiency**			**Performance**
	Params(M)↓	FLOPs(G)↓	FPS↑	mAD(%)↑
MambaAD [11]	25.7	8.3	59	86.4
SimpleNet [18]	72.8	16.1	62	81.2
DeSTSeg [30]	35.2	122.7	125	75.4
Dinomaly [10]	132.8	104.7	59	90.3
INP-Former [19]	139.8	98.0	38	91.0
$\mathbf{FDP}_{392\times392}$	38.7	42.4	358	91.6
$\mathbf{FDP}^{\dagger}_{392\times392}$	38.7	42.4	477	91.6
$\mathbf{FDP}_{512\times512}$	38.7	73.9	192	93.5
$\mathbf{FDP}^{\dagger}_{512\times512}$	38.7	73.9	240	93.5

FDP† denotes the TensorRT-accelerated model, while FDP refers to the PyTorch implementation. Additionally, considering that the input size of other methods is 392×392, we also provide a faster 392×392 version. It is worth noting that since DINOv3 requires the size of the input image to be a multiple of 16 and 392 does not meet this requirement, it will be automatically cropped to the nearest multiple of 16 (384).

ment of $\varDelta$**mAD : 2.6** compared to the baseline(10^5) while also providing faster inference speed. The experimental results strongly demonstrate the effectiveness and synergistic contributions of each module.

4.4 Validation of Feature Complementarity and Hybrid Residuals

Verification Feature's Complementarity The semantic artifact problem in this DINO series for real-object segmentation tasks has been greatly mitigated by the gram loss proposed by DINOv3 [22]. However, in industrial anomaly segmentation tasks, there are still many semantic artifacts, as shown in Fig. 4. On top, the residuals from the supervised encoder, ResNet-18, show fuzzy localization but fewer artifacts. On the bottom, residuals extracted from the self-supervised encoder DIN0v3-small exhibit precise localization but more artifacts. The two approaches exhibit complementary characteristics in the residual space, suggesting potential integration to improve localization accuracy and suppress artifacts.

Table 5: Performance comparison using same backbone architectures across different methods on MVTec-AD dataset. The experimental results of this INP-Former [19] and Dinomaly [10] were obtained after we conducted experiments by replacing the backbone according to the official code provided.

DINOv3-Small				**DINOv3-Base**			
Method	I-AUROC	P-AP	mAD	**Method**	I-AUROC	P-AP	mAD
Dinomaly [10]	99.0	70.4	90.4	Dinomaly [10]	99.4	72.7	91.6
INP-Former [19]	99.2	70.8	90.8	INP-Former [19]	99.4	72.9	91.6
FDP	99.7	79.1	93.5	**FDP**	99.5	79.4	93.6

According to the data preprocessing of INP-Former and Dinomaly, the input images are resized to 448, then cropped to 392. However, since 392 does not meet the requirement of DINOv3 input size being a multiple of 16, we uniformly use bilinear sampling to resize the cropped 392 images to 512 × 512.

Table 6: Ablation Study Results of Different Modules on MVTec-AD dataset. These DPA, CPR, and HRL refer to Dynamic Prototype Adapter, Complementary Prototype Reconstruction, and Hybrid Residual Learning, respectively. Baseline(10^5) and Baseline(8192) denote the baseline whose size of prototype bank is 10^5 and 8192, respectively.

Model Variant	**Modules**				**I-AUROC (%)**	**P-AP (%)**	**mAD (%)**
	DPA	**CPR**	**Seg. Head**	**HRL**			
Baseline(10^5)	✗	✗	✗	✗	99.6	71.8	90.9
Baseline(8192)	✗	✗	✗	✗	98.8	65.4	87.4
+ **DPA**	✓	✗	✗	✗	99.6	72.5	91.3
+ **CPR**	✓	✓	✗	✗	99.7	73.0	91.4
+ **Seg. Head**	✓	✓	✓	✗	99.7	77.8	92.9
+ **HRL(Full)**	✓	✓	✓	✓	99.7	79.1	93.5

In addition, we computed the CKA scores [15] to measure the complementarity of two deep networks (Tab. 7). The lowest CKA score between DINOv3-S and ResNet18 indicates that the two networks are sufficiently complementary.

Cosine Residual vs. Hybrid Residual In the residual visualization experiments, we found that the residuals calculated directly using the cosine-similarity-based multiplication paradigm contain substantial real-world object information, as shown in the second column of Fig. 5. This type of information constitutes significant background clutter and semantic noise in residual-based segmentation training, severely interfering with anomaly localization. To address this issue, we propose a hybrid residual that improves the calculation of residuals. The hybrid residuals can effectively filter out real object information and significantly highlight the differences between abnormal and normal regions, as shown in the third column of Fig. 5, thereby enhancing the performance of industrial anomaly detection.

Table 7: CKA scores to quantify the feature complementarity encoded by the dual backbones.

Configuration	Pri. Encoder	Aux. Encoder	MVTec-AD	MVTec-3D	VisA
Dual SSL Encoder	DINOv3-B	DINOv3-S	0.8767	0.8616	0.8665
Dual SL Encoder	ResNet34	ResNet18	0.9963	0.9957	0.9961
Comp. Encoder	DINOv3-S	ResNet18	0.3786	0.3739	0.4139

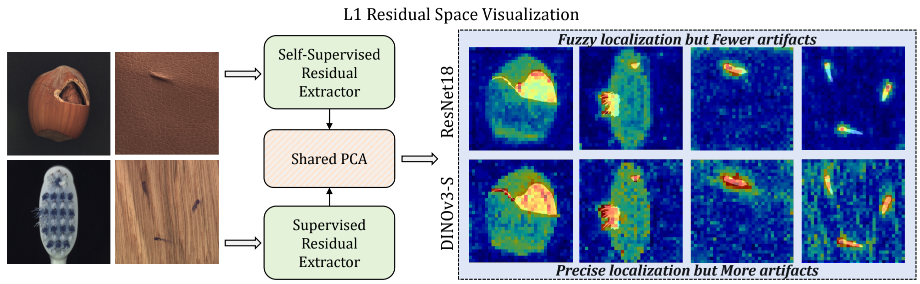

Fig. 4: Visualization results of the residuals obtained when self-supervised and supervised models are used individually as backbones. The Self-Supervised Residual Extractor and the Supervised Residual Extractor both use our method, but one uses DINOv3-small as the backbone and the other uses ResNet-18. The residuals from different backbones will be input into the shared PCA, which will be reduced to 1 dimension and visualized; the highlighted area indicates the anomaly.

5 Conclusion

This paper introduces Fast Dynamic Prototype (FDP), a novel framework for unsupervised anomaly detection that foregrounds both interpretability and computational efficiency. In contrast to prevailing methods that depend on dense memory banks or complex, implicit transformer-based refinements, FDP constructs adaptive prototypes for test features through a sequence of four explicit and modular operations: clustering-based bank construction, dynamic bank adaptation, complementary prototype fusion via a dual-backbone design, and a lightweight segmentation head with hybrid residuals. This architecture directly mitigates the redundancy and scalability limitations of traditional approaches while preserving a transparent and interpretable pipeline.

Complementing the core FDP mechanism, we proposed a hybrid residual calculation method that enhances both detection and localization accuracy with negligible additional computational cost. Extensive evaluations on the MVTec-AD, MVTec-3D, and VisA benchmarks demonstrate that FDP not only achieves state-of-the-art performance but also delivers substantially faster inference speeds. These results validate that explicit dynamic prototype reconstruction offers a powerful and practical alternative to both dense storage and attention-based methods.

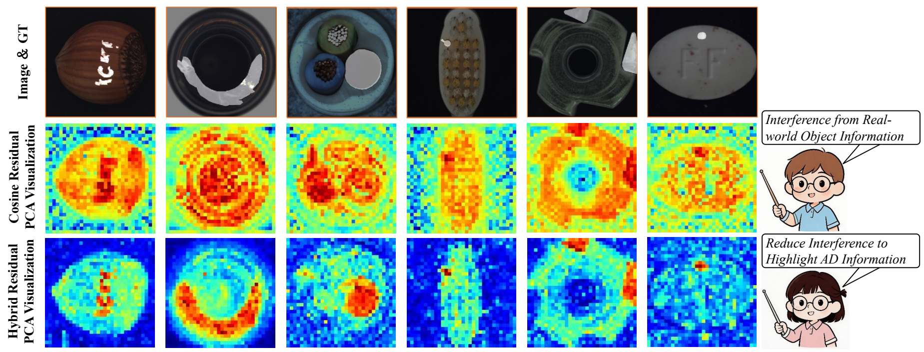

Fig. 5: Comparison of effects with different residuals. The first column shows the RGB images, with highlighted areas indicating the actual anomaly regions; the second and third columns show feature heatmaps obtained via PCA dimensionality reduction based on the traditional cosine residual and our proposed hybrid residual, respectively.

Table 8: Quantitative evaluation of the dynamic extraction mechanism across different anomaly area ratios.

Anomaly Area Ratio	Number	I-AUROC	P-AP	P-PRO	P-AUROC
0%–5%	1015	98.1/99.3/+1.2	75.2/77.0/+1.8	95.5/97.3/+1.8	99.0/99.2/+0.2
5%–20%	185	100.0/100.0/0	88.4/88.2/-0.2	97.6/97.8/+0.2	99.3/99.3/0
20%–100%	58	99.1/99.3/+0.2	77.2/76.1/-1.1	89.8/89.4/-0.5	95.4/95.5/+0.1

Metrics are presented in the format FDP-/FDP/Δ, where FDP- refers to the proposed FDP without the dynamic extraction mechanism, and Δ represents the relative improvement from FDP- to FDP.

Limitation and Future Work It is important to note that the proposed method leverages information from the test image itself to adapt the prototype bank. Both our ablation study and INP-Former [19] demonstrate that dynamic extraction mechanism to optimize reconstruction or prototype bank is generally effective. However, we observe that its effectiveness varies across samples with different anomalous regions. To this end, we supplement the test results stratified by defect area proportion. As shown in Table 8, while this strategy proves particularly effective for samples with small defects that retain abundant normal context, it may introduce excessive abnormal information into the normal feature bank when processing samples dominated by large defects. Nevertheless, the overall benefits of this strategy substantially outweigh this limitation.

Future work could explore a more defect-aware adaptation module that selectively exploits useful information from test images while suppressing contamination from anomalous regions, thereby improving UAD performance.

6 Acknowledgment

This project was partially supported by NSFC 62372150.

References

1. Deep industrial image anomaly detection: A survey. Machine Intelligence Research **21**(1), 104–135 (2024)
2. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Mvtec ad — a comprehensive real-world dataset for unsupervised anomaly detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9584–9592 (2019)
3. Bergmann, P., Jin, X., Sattlegger, D., Steger, C.: The mvtec 3d-ad dataset for unsupervised 3d anomaly detection and localization. ArXiv **abs/2112.09045** (2021)
4. Chen, Z., Xie, X., Yang, L., Lai, J.: Hard-normal example-aware template mutual matching for industrial anomaly detection. International Journal of Computer Vision (IJCV) (2024)
5. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. 2014 IEEE Conference on Computer Vision and Pattern Recognition pp. 3606–3613 (2013)
6. Darcet, T., Oquab, M., Mairal, J., Bojanowski, P.: Vision transformers need registers. In: The Twelfth International Conference on Learning Representations (2024)
7. 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)
8. Gao, B.B.: Learning to detect multi-class anomalies with just one normal image prompt. In: 18th European Conference on Computer Vision (ECCV 2024). pp. – (2024)
9. Gao, B.B., Wang, C.: One language-free foundation model is enough for universal vision anomaly detection. In: arXiv:2601.05552 (2026)
10. Guo, J., Lu, S., Zhang, W., Li, H.: Dinomaly: The less is more philosophy in multi-class unsupervised anomaly detection. 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 20405–20415 (2024)
11. He, H., Bai, Y., Zhang, J., He, Q., Chen, H., Gan, Z., Wang, C., Li, X., Tian, G., Xie, L.: Mambaad: Exploring state space models for multi-class unsupervised anomaly detection. Advances in Neural Information Processing Systems **37**, 71162–71187 (2024)
12. He, H., Zhang, J., Chen, H., Chen, X., Li, Z., Chen, X., Wang, Y., Wang, C., Xie, L.: A diffusion-based framework for multi-class anomaly detection. In: Proceedings of the AAAI conference on artificial intelligence. vol. 38, pp. 8472–8480 (2024)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 770–778 (2015)
14. Huang, G., Liu, Z., Weinberger, K.Q.: Densely connected convolutional networks. CoRR **abs/1608.06993** (2016)
15. Kornblith, S., Norouzi, M., Lee, H., Hinton, G.E.: Similarity of neural network representations revisited. ArXiv **abs/1905.00414** (2019)
16. Li, H., Hu, J., Li, B., Chen, H., Zheng, Y., Shen, C.: Target before shooting: Accurate anomaly detection and localization under one millisecond via cascade patch retrieval. IEEE Transactions on Image Processing **33**, 5606–5621 (2024)
17. Li, H., Wu, J., Liu, D., Wu, L.Y., Chen, H., Shen, C.: Accurate industrial anomaly detection and localization using weakly-supervised residual transformers. IEEE Transactions on Image Processing (2026)

18. Liu, Z., Zhou, Y., Xu, Y., Wang, Z.: Simplenet: A simple network for image anomaly detection and localization. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 20402–20411 (2023)
19. Luo, W., Cao, Y., Yao, H., Zhang, X., Lou, J., Cheng, Y., Shen, W., Yu, W.: Exploring intrinsic normal prototypes within a single image for universal anomaly detection. 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 9974–9983 (2025)
20. Qu, Z., Tao, X., Gong, X., Qu, S., Zhang, X., Wang, X., Shen, F., Zhang, Z., Prasad, M., Ding, G.: Dictas: A framework for class-generalizable few-shot anomaly segmentation via dictionary lookup. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 20519–20528 (October 2025)
21. Roth, K., Pemula, L., Zepeda, J., Scholkopf, B., Brox, T., Gehler, P.: Towards total recall in industrial anomaly detection. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 14298–14308 (2021)
22. Sim'eoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., Massa, F., Haziza, D., Wehrstedt, L., Wang, J., Darcet, T., Moutakanni, T., Sentana, L., Roberts, C., Vedaldi, A., Tolan, J., Brandt, J., Couprie, C., Mairal, J., J'egou, H., Labatut, P., Bojanowski, P.: Dinov3 (2025)
23. Wang, C., Zhu, W., Gao, B.B., Gan, Z., Zhang, J., Gu, Z., Qian, S., Chen, M., Ma, L.: Real-iad: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 22883–22892 (2024)
24. Yao, X., Luo, Y., Qian, Z., Zhang, C.: Adpretrain: Advancing industrial anomaly detection via anomaly representation pretraining. In: Thirty-Ninth Annual Conference on Neural Information Processing Systems, NeurIPS 2025 (2024)
25. You, Z., Cui, L., Shen, Y., Yang, K., Lu, X., Zheng, Y., Le, X.: A unified model for multi-class anomaly detection. Advances in Neural Information Processing Systems **35**, 4571–4584 (2022)
26. Zavrtanik, V., Kristan, M., Skoaj, D.: Drm – a discriminatively trained reconstruction embedding for surface anomaly detection. 2021 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 8310–8319 (2021)
27. Zhai, G., Zhou, Y., Deng, X., Heckler, L., Navab, N., Busam, B.: Foundation visual encoders are secretly few-shot anomaly detectors. ArXiv **abs/2510.01934** (2025)
28. 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 (CVPR). pp. 16281–16291 (June 2023)
29. Zhang, X., Xu, M., Zhou, X.: Realnet: A feature selection network with realistic synthetic anomaly for anomaly detection. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 16699–16708 (2024)
30. Zhang, X., Li, S., Li, X., Huang, P.C., Shan, J., Chen, T.: Destseg: Segmentation guided denoising student-teacher for anomaly detection. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 3914–3923 (2022)
31. Zheng, P., Gao, D., Fan, D.P., Liu, L., Laaksonen, J., Ouyang, W., Sebe, N.: Bilateral reference for high-resolution dichotomous image segmentation. CAAI Artificial Intelligence Research **3**, 9150038 (2024)
32. Zheng, Y., Wang, X., Deng, R., Bao, T., Zhao, R., Wu, L.: Focus your distribution: Coarse-to-fine non-contrastive learning for anomaly detection and localization. 2022 IEEE International Conference on Multimedia and Expo (ICME) pp. 1–6 (2021)

33. Zhou, Z., Wang, L., Fang, N., Wang, Z., Qiu, L., Zhang, S.: R3d-ad: Reconstruction via diffusion for 3d anomaly detection. In: European conference on computer vision. pp. 91–107. Springer (2024)
34. 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 (2022)