RcAE: Recursive Reconstruction Framework for Unsupervised Industrial Anomaly Detection

Rongcheng Wu[2, 6†], Hao Zhu[3†], Shiying Zhang[1†], Mingzhe Wang[1†*], Zhidong Li[2], Hui Li[1], Jianlong Zhou[2], Jiangtao Cui[1], Fang Chen[2], Pingyang Sun[4], Qiyu Liao[3], Ye Lin[2, 5, 6]

[1]School of Computer Science and Technology, Xidian University
[2]The Data Science Institute, University of Technology Sydney
[3]Data61, CSIRO
[4]School of Photovoltaic and Renewable Energy Engineering, University of New South Wales
[5]Department of Computing, The Hong Kong Polytechnic University
[6]Molly Wardaguga Institute for First Nations Birth Rights, Faculty of Health, Charles Darwin University

Abstract

Unsupervised industrial anomaly detection requires accurately identifying defects without labeled data. Traditional autoencoder-based methods often struggle with incomplete anomaly suppression and loss of fine details, as their single-pass decoding fails to effectively handle anomalies with varying severity and scale. We propose a recursive architecture for autoencoder (RcAE), which performs reconstruction iteratively to progressively suppress anomalies while refining normal structures. Unlike traditional single-pass models, this recursive design naturally produces a sequence of reconstructions, progressively exposing suppressed abnormal patterns. To leverage this reconstruction dynamics, we introduce a Cross Recursion Detection (CRD) module that tracks inconsistencies across recursion steps, enhancing detection of both subtle and large-scale anomalies. Additionally, we incorporate a Detail Preservation Network (DPN) to recover high-frequency textures typically lost during reconstruction. Extensive experiments demonstrate that our method significantly outperforms existing non-diffusion methods, and achieves performance on par with recent diffusion models with only 10% of their parameters and offering substantially faster inference. These results highlight the practicality and efficiency of our approach for real-world applications.

Introduction

Anomaly detection is a fundamental task in computer vision with broad applications, such as manufacturing quality control (Bergmann et al. 2021) and surveillance (Chandola, Banerjee, and Kumar 2009). In industry, accurate defect detection is critical for ensuring product quality and reducing cost. However, a major challenge arises from severe data imbalance: normal samples are abundant, while anomalies are rare, diverse, and difficult to annotate (Ruff et al. 2021).

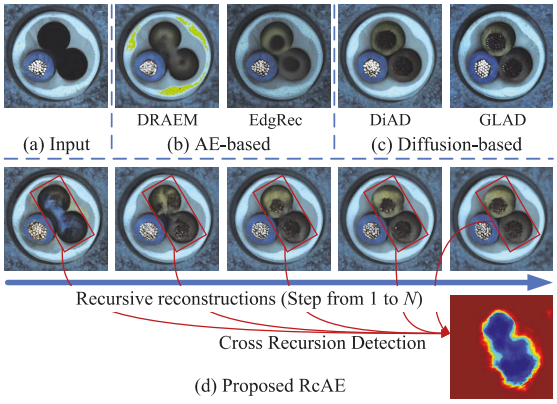


Figure 1: Reconstruction comparison of AE and diffusion-based methods. Our recursive design incrementally suppresses anomalies for high-fidelity reconstruction, then leverages cross-recursion dynamics for accurate detection.

To address this, unsupervised and semi-supervised methods have gained popularity, typically learning the distribution of normal data and detecting deviations at test time as anomalies (Tao et al. 2022). Among them, reconstruction-based methods, especially autoencoder (AE) variants, stand out for their simplicity and intuitive design: models trained on normal data are expected to poorly reconstruct unseen anomalies (Zavrtanik, Kristan, and Skočaj 2021; Gong et al. 2019). However, traditional AE-based methods face several limitations in industrial anomaly detection: (1) Overfitting to limited and homogeneous normal data (Kingma, Welling et al. 2019); (2) Expressive latent spaces may reconstruct anomalies (Bergmann et al. 2018); (3) Single-pass decoding that oversmooths fine details (Liu et al. 2020; Zavrtanik, Kristan, and Skočaj 2021); (4) Fixed-scale architectures struggle with varying size and severity anomalies (Zavrtanik, Kristan, and Skocaj 2021; Ristea et al. 2022).

These limitations lead to poor reconstruction quality in

[†]These authors contributed equally.
[*]Corresponding author: Mingzhe Wang (E-mail address: wangmingzhe@xidian.edu.cn).

traditional AEs. To address this, recent works have introduced GANs (Liang et al. 2023), transformers (You et al. 2022), diffusion models (He et al. 2024; Zhang et al. 2023), or augmentation-heavy strategies (Li et al. 2021; Zavrtanik, Kristan, and Skocaj 2021; Zhang, Xu, and Zhou 2024). While effective, these methods typically require high computational cost or intricate pre-processing pipelines, limiting their practicality in industrial settings.

In light of these challenges, we revisit a fundamental question: *Can high-quality anomaly reconstruction be achieved without the overhead of complex or resource-heavy designs?* To this end, we propose a lightweight and effective framework based on simple AEs called **Recursive Convolutional Autoencoder (RcAE)**. As shown in Fig. 1, unlike traditional single-pass AEs, RcAE performs iterative reconstruction, progressively suppressing anomalies and reconstructing normal structures across multiple steps. This recursive formulation enables residual evolution to naturally highlight inconsistent anomalies patterns over recursive steps. To leverage this reconstruction dynamics, we propose a novel **Cross Recursion Detection (CRD)** module, which monitors reconstruction inconsistencies of anomalies across recursion steps. CRD captures persistent deviations that reveal both micro-defects and large structural anomalies. Additionally, we introduce a lightweight **Detail Preservation Network (DPN)** to restore fine textures in normal regions that might otherwise be oversmoothed, further reducing false positives caused by detail loss.

Overall, this paper forms an efficient framework, achieving strong detection performance without significantly high computational overhead, making it practical for real-world deployment. **The main contributions are:**

- **Recursive Reconstruction Framework** that performs anomaly suppression and normal pattern enhancement over multiple iterations, enabling significantly robust reconstructions without high computational burden.
- **Cross Recursion Detection:** We introduce a novel module that exploits patterns of constantly changing anomalies across recursive steps. By analyzing persistent inconsistencies, Cross Recursion Detection enables unified detection of both subtle and large-scale anomalies.
- **Detail Refinement:** While our recursive architecture provide fine reconstructions, we further design a Detail Preservation Network that selectively restores fine textures in normal regions, preserving structural fidelity.
- **Efficiency:** Our method achieves state-of-the-art performance on par with recent diffusion models, while requiring $10\times$ fewer parameters and offering faster inference, making it suitable for industrial deployment.

Related Works

Unsupervised anomaly detection models are trained by only normal data. Reconstruction-based methods remain prominent due to their intuitive logic: a model that learns to reconstruct normal patterns should struggle to reconstruct unseen anomalies, thus exposing them by reconstruction errors.

Autoencoders: Foundations and Challenges. A classical realization of this paradigm is the autoencoder (AE) framework, which learns to compress and reconstruct normal data (Gong et al. 2019; Bergmann et al. 2022; Park, Noh, and Ham 2020). While simple and effective in low-data regimes, traditional AEs exhibit several persistent drawbacks: (1) They easily overfit on limited and homogeneous normal data, leading to poor generalization and high false negatives (Kingma, Welling et al. 2019); (2) Expressive latent spaces may reconstruct anomalous regions, diminishing detection contrast and accuracy (Bergmann et al. 2018); (3) Single-pass decoding often smooths out high-frequency details, causing false positives in normal areas (Liu et al. 2020; Zavrtanik, Kristan, and Skočaj 2021); (4) Fixed-scale architectures struggle with anomalies of varying size and severity, limiting robustness across diverse defect types (Zavrtanik, Kristan, and Skocaj 2021; Ristea et al. 2022). These limitations expose the fragility of single-pass reconstruction mechanisms, especially in high-precision use cases.

Beyond AEs: Better Reconstructions at a Cost. To address the reconstruction limitations of basic AEs, recent research has explored more expressive models: **GAN-based approaches** (Liang et al. 2023; Akçay, Atapour-Abarghouei, and Breckon 2019) introduce adversarial losses to generate visually sharper reconstructions. While enhancing realism, they often suffer from training instability and require large-scale data (Akcay, Atapour-Abarghouei, and Breckon 2019), limiting their practicality in sparse-label industrial scenarios. **Transformer-based models** (You et al. 2022; Mishra et al. 2021; Pirnay and Chai 2022) leverage self-attention to capture long-range dependencies. For instance, UniAD (You et al. 2022) unifies multiple object categories under a single framework. However, their high memory cost and complex optimization hinder deployment in real-time applications (Xu et al. 2021). **Diffusion-based frameworks** (He et al. 2024; Yao et al. 2025; Zhang et al. 2023; Wyatt et al. 2022) have recently achieved state-of-the-art reconstruction quality by learning denoising trajectories from noise to clean samples. Their iterative nature enables gradual normalization of anomalies, making them powerful but computationally prohibitive, often requiring dozens to hundreds of denoising steps per image (Rombach et al. 2022). Moreover, diffusion models may still suffer from semantic inconsistencies and detail loss (Batzner, Heckler, and König 2024; Zavrtanik, Kristan, and Skočaj 2021). These reflects a clear trend: improvements in reconstruction typically comes at the cost of inference latency, training complexity, or resource requirements, which is a tradeoff that undermines deployment in resource-constrained industrial settings.

Limitations and Our Perspective. Despite ongoing progress, reconstruction-based methods still face several persistent challenges: (1) Loss of fine details in normal regions, leading to false positives (Denouden et al. 2018); (2) Incomplete suppression of anomalies, causing false negatives (Cai, Chen, and Cheng 2024); (3) Loose boundary modeling of normal distributions, allowing reconstruction of anomalies (Pang et al. 2021); (4) High computational demands in high-performing models, leading to poor deployment feasibility (Fučka, Zavrtanik, and Skočaj 2024). Thus, improving reconstruction quality without incurring high cost

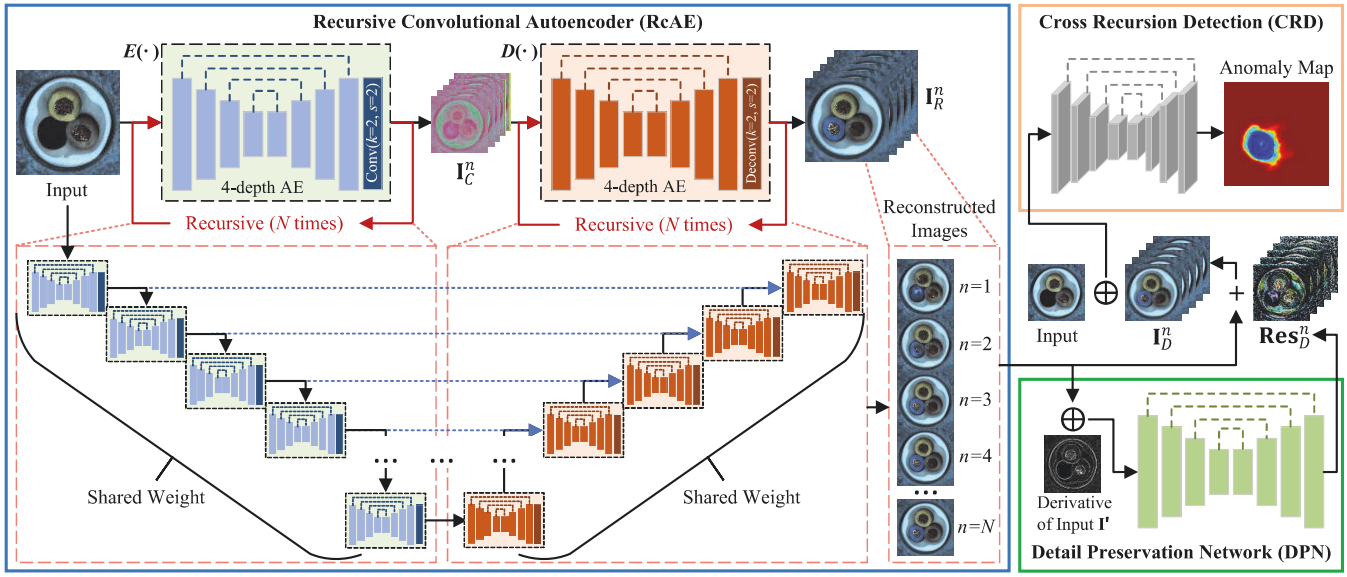


Figure 2: Overview of the proposed anomaly detection framework. RcAE performs iterative reconstruction to progressively suppress anomalies and stabilize normal structures; DPN selectively restores high-frequency textures; CRD leverages reconstruction dynamics across recursion steps to produce robust pixel-level anomaly maps.

is still a challenging issue, and we aim to propose a framework that is both high-performing and computationally efficient, striking a better balance for practical deployment.

Methodology

As illustrated in Fig. 2, the proposed framework is a lightweight yet expressive reconstruction-based pipeline with three key components: (1) a **Recursive Autoencoder (RcAE)** that reformulates reconstruction as a multi-step refinement process, progressively suppressing anomalies while stabilizes normal structures across iterations; (2) a **Detail Preservation Network (DPN)** that restores high-frequency details in normal regions, reducing false positives while preserving structural fidelity; and (3) a **Cross Recursion Detection (CRD)** that exploits reconstruction dynamics across recursion steps to highlight persistent inconsistencies, generating robust anomaly detection.

Preliminaries

Let input space $\mathcal{X} \subset \mathbb{R}^{H\times W\times C}$ contain images with $H \times W$ resolution and C channels. The training set $\mathcal{D}_{\text{train}} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_S\}$ contains only normal images $\mathbf{x}_i \in \mathcal{X}$. The goal of unsupervised anomaly detection is to learn a mapping $f : \mathcal{X} \to \mathcal{X}$ such that reconstruction errors $|\mathbf{x}_i - f(\mathbf{x}_i)|$ expose anomalies in unseen test images. During inference, an anomaly score is assigned to each pixel of a test image $\mathbf{x}_{\text{test}} \in \mathcal{X}$ based on deviations from learned normal patterns.

Standard Autoencoder Framework. A traditional autoencoder consists of an encoder $E_{\boldsymbol{\theta}_E} : \mathcal{X} \to \mathcal{Z}$ and a decoder $D_{\boldsymbol{\theta}_D} : \mathcal{Z} \to \mathcal{X}$, where $\mathcal{Z} \subset \mathbb{R}^d$ is a compact latent space with dimensionality $d \ll H \times W \times C$, $\boldsymbol{\theta}_E$ and $\boldsymbol{\theta}_D$ are the parameters. The network is trained to minimize the reconstruction loss on normal samples:

$$\mathcal{L}_{\text{rec}} = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - D_{\boldsymbol{\theta}_D}(E_{\boldsymbol{\theta}_E}(\mathbf{x}_i))\|_p, \quad (1)$$

where $\|\cdot\|_p$ denotes the ℓ_p norm. The reconstruction of an input image x is denoted as $\hat{x} = D_{\boldsymbol{\theta}_D}(E_{\boldsymbol{\theta}_E}(\mathbf{x}))$. The AE-based methods are simple and efficient, but often fail to fully suppress subtle anomalies and preserve fine details.

Recursive Convolutional Autoencoder

Traditional deep convolutional autoencoders (ConvAEs) improve reconstruction capacity by stacking N distinct encoder and decoder blocks. Formally, the encoder blocks $\{E_1, E_2, ..., E_N\}$ with parameters $\{\boldsymbol{\theta}_{E_1}, \boldsymbol{\theta}_{E_2}, ..., \boldsymbol{\theta}_{E_N}\}$ and decoder blocks $\{D_1, D_2, ..., D_N\}$ with parameters $\{\boldsymbol{\theta}_{D_1}, \boldsymbol{\theta}_{D_2}, ..., \boldsymbol{\theta}_{D_N}\}$, define a mapping:

$$f_{\text{deep}}(\mathbf{x}) = D_1 \circ D_2 \circ ... \circ D_N \circ E_N \circ ... \circ E_2 \circ E_1(\mathbf{x}), \quad (2)$$

where each block has it own parameters. While effective but significantly increases model size and training complexity.

To achieve high-quality reconstruction without inflating the parameter count, we propose a Recursive Convolutional Autoencoder (RcAE) that replaces these stacked layers with a compact, recursive design. Instead of using multiple independent blocks, RcAE reuses two shared autoencoders in an iterative pipeline to perform progressive compression and reconstruction, mimicking the depth and semantic abstraction of a deep ConvAE with far fewer parameters.

Recursive Structure. As shown in Fig. 2, RcAE consists of two phases: a recursive compression phase with encoder $E(\cdot)$ and a recursive reconstruction phase with decoder $D(\cdot)$, both adopt a standard 4-layer ConvAE with skip connections and parameter sharing within each phase. To simulate the behavior of a deeper AE while preserving the scale-space encoding and decoding, we insert a downsampling

convolution layer (kernel size = 2, stride = 2) after $E(\cdot)$, reducing spatial resolution by a factor of 2. Likewise, $D(\cdot)$ ends with a deconvolutional upsampling layer (kernel size = 2, stride = 2) to restore resolution. This allows each recursion operates at progressively coarser semantic levels.

Compression Phase. Given input image $\mathbf{x}$, the compression phase progressively compresses it through N recursions using a shared encoder E with parameters $\boldsymbol{\theta}_E$:

$$\mathbf{I}_C^i = E(\mathbf{I}_C^{i-1}; \boldsymbol{\theta}_E), \quad i \in \{1, 2, ..., N\}, \tag{3}$$

where $\mathbf{I}_C^0 = \mathbf{x}$. Each recursion step reduces spatial resolution by the shared encoder $E : \mathcal{X}_{i-1} \to \mathcal{X}_i$ maps from one resolution level to a lower one, where $\mathcal{X}_i \subset \mathbb{R}^{H/2^i \times W/2^i \times C}$, enabling hierarchical abstraction of visual content.

Reconstruction Phase. Starting from the compressed representation $\mathbf{I}_C^N$ from N-times recursion, the reconstruction phase progressively reconstructs the image through N recursions using a shared decoder D with parameters $\boldsymbol{\theta}_D$:

$$\mathbf{I}_R^j = D(\mathbf{I}_R^{j-1}; \boldsymbol{\theta}_D), \quad j \in \{1, 2, ..., N\}, \tag{4}$$

where $\mathbf{I}_R^0 = \mathbf{I}_C^N$, and each iteration restores the spatial resolution by the shared decoder $D : \mathcal{X}_{N-j+1} \to \mathcal{X}_{N-j}$ which maps from one resolution level to a higher one. The final reconstruction $\mathbf{I}_R^N$ represents the full-resolution anomaly-normalized output.

Progressive Reconstruction Benefits. Together, these components implement a recursive architecture where deeper iteration compresses and reconstructs the image at a coarser scale, effectively performing reconstruction at multiple semantic levels: early iterations retain low-level details but may still contain residual anomalies, while later iterations better suppress anomalies but may over-smooth fine structures. Such progressive refinement allows RcAE to simultaneously suppress anomalies and preserve normal structures, improving robustness without extra parameters.

Training. To encourage generalization across varying recursive depths and adapt to anomalies of different intensities, the number of recursions is randomly sampled from $[1, N]$ during training, which also help avoiding shortcut learning. We supervise both intensity and edge fidelity as follows:

$$\mathcal{L}_{\text{rec}} = \|\mathbf{I} - \mathbf{I}_R^N\|_1 + \|\mathbf{I}' - \mathbf{I}_R'^N\|_1, \tag{5}$$

where $\mathbf{I}'$ and $\mathbf{I}_R'^N$ denote the first-order gradients of input and the final reconstruction, respectively.

Detail Preservation Network

While RcAE effectively suppresses anomalies through multi-step reconstruction, the recursive nature of the process can also lead to the accumulation of detail loss in normal regions, leading to false positives.

To address this, we propose a Detail Preservation Network (DPN), which selectively restores fine details in normal areas while avoiding the reintroduction of anomalies. As illustrated in the 2nd stage of Fig. 2, DPN is a lightweight 4-layer convolutional autoencoder with skip connections, which takes the recursive reconstructions $\{\mathbf{I}_R^n\}$ concatenated with the first-order derivative of the input $\mathbf{I}'$, to predict residual maps $\{\mathbf{Res}_D^n\}$ contains missing details:

$$\mathbf{Res}_D^n = f_{\text{DPN}}((\mathbf{I}_R^n \oplus \mathbf{I}'); \boldsymbol{\theta}_{\text{DPN}}), \quad \mathbf{I}_D^n = \mathbf{I}_R^n + \mathbf{Res}_D^n, \tag{6}$$

where $\oplus$ denotes channel-wise concatenation, and $\boldsymbol{\theta}_{\text{DPN}}$ is the parameters. The output $\mathbf{I}_D^n$ represents the detail-enhanced reconstruction for downstream anomaly scoring.

DPN is supervised by a dual-term ℓ_1 loss to recover both intensity and edge details:

$$\mathcal{L}_{\text{DPN}} = \|(\mathbf{Res}_D^n + \mathbf{I}_R^n) - \mathbf{I}\|_1 + \|(\mathbf{Res}_D^n + \mathbf{I}_R^n)' - \mathbf{I}'\|_1, \tag{7}$$

where $\mathbf{I}'$ and $(\mathbf{Res}_D^n + \mathbf{I}_R^n)'$ denote the gradient maps of $\mathbf{I}$ and $\mathbf{I}_D^n$, respectively.

Importantly, RcAE is frozen during DPN training, and only clean normal samples are used. This forces the network to focus on learning residuals caused by recursive detail degradation rather than anomaly-related deviations. In inference, since anomalies generate unfamiliar residuals outside the learned distribution, DPN naturally fails to restore them, preserving anomaly suppression. This selective recovery mechanism effectively enhances fine details in normal regions while reducing false positives, thereby improving the reliability of pixel-wise anomaly detection.

Cross Recursion Detection

Our recursive design naturally produces a sequence of reconstructions, and the differences between steps reflect region-wise stability, i.e., normal regions stabilize quickly, while anomalous regions fluctuate due to reconstruction difficulty. Early iterations may retain residual defects but preserve fine details, whereas later iterations suppress anomalies better but may lose subtle textures. To leverage this reconstruction dynamics, we introduce the Cross Recursion Detection (CRD) for robust anomaly localization.

As shown in the 3rd stage of Fig. 2, CRD is a 4-depth 3D ConvAE with skip connections that jointly models spatial features and reconstruction dynamics across recursion steps. It takes the concatenation of the input $\mathbf{I}$ and the detail-enhanced reconstructions $\mathbf{I}_D^n$ to predict anomaly map $\mathbf{M}_A$:

$$\mathbf{M}_A = f_{\text{CRD}}((\mathbf{I}_D^n \oplus \mathbf{I}); \boldsymbol{\theta}_{\text{CRD}}), \quad n \in \{1, 2, ..., N\}, \tag{8}$$

where $\boldsymbol{\theta}_{\text{CRD}}$ are the learnable parameters. 3D convolutions allow CRD to extract cross-recursion temporal patterns, highlighting regions that remain unstable across iterations.

During training, both RcAE and DPN are frozen. We use only normal images, and generate pseudo anomaly masks $\mathbf{M}_P$ via simple augmentations (e.g., color patches, random lines, copy-paste). CRD is optimized using a dual-term ℓ_2 loss for spatial and edge consistency:

$$\mathcal{L}_{\text{CRD}} = \|\mathbf{M}_A - \mathbf{M}_P\|_2 + \|\mathbf{M}_A' - \mathbf{M}_P'\|_2, \tag{9}$$

where $\mathbf{M}'$ denotes the gradient map. At test time, CRD outputs the final pixel-wise anomaly map $\mathbf{M}_A$. For image-level anomaly detection, we follow standard practice of averaging the top-k pixel scores.

In contrast to prior methods that rely solely on a single reconstruction, our CRD module fully exploits cross-recursion dynamics of RcAE, offering reliable detection of anomalies at multiple scales and varying intensities.

Category			Non-Diffusion Method					Diffusion-based			Flow-based	Diffusion+DINO
			DRAEM	PatchCore	RD4AD	EfficientAD	**Ours**	D3AD	DiAD	DiffAD	MSFlow	GLAD
From Scratch?			✓	✗	✗	✗	✓	✗	✗	✓	✗	✗
MVTec AD Dataset	Objects	Bottle	99.2/**99.1**	**100**/98.6	**100**/99.0	99.9/98.7	**100**/**99.1**	**100**/98.6	99.7/98.4	**100**/98.8	**100**/99.0	**100**/98.9
		Cable	91.8/94.7	99.5/98.4	95.0/**99.4**	95.2/98.8	97.4/97.1	97.8/93.3	94.8/96.8	94.6/96.8	99.5/98.5	**99.9**/98.1
		Capsule	98.5/94.3	98.1/98.8	96.3/97.3	97.9/**99.2**	94.4/97.9	96.6/97.9	89.0/97.1	97.5/98.2	99.2/98.1	**99.5**/98.5
		Hazelnut	**100**/**99.7**	**100**/98.7	99.9/98.2	99.4/98.8	**100**/99.5	98.0/98.8	99.5/98.3	**100**/99.4	**100**/98.7	**100**/99.5
		Metal Nut	98.7/99.5	**100**/98.4	**100**/**99.6**	99.6/98.5	99.8/98.8	98.9/96.1	99.1/97.3	**100**/99.4	**100**/99.3	**100**/98.8
		Pill	98.9/97.6	96.6/97.4	96.6/95.7	98.6/98.7	98.4/**98.9**	99.2/98.2	95.7/95.7	97.7/97.7	**99.6**/98.8	98.1/97.9
		Screw	93.9/97.6	97.0/**99.1**	97.0/**99.1**	97.0/98.7	95.8/98.7	83.9/99.0	90.7/97.9	97.2/99.0	**97.8**/**99.1**	96.9/**99.1**
		Toothbrush	**100**/98.1	**100**/98.7	99.5/93.0	**100**/97.7	**100**/**99.4**	**100**/99.0	99.7/99.0	**100**/99.2	**100**/98.5	**100**/**99.4**
		Transistor	93.1/90.9	**100**/96.3	96.7/95.4	99.9/97.2	98.6/96.6	96.8/95.6	99.8/95.1	96.1/93.7	**100**/**98.3**	98.3/96.2
		Zipper	**100**/98.8	99.4/98.8	98.5/98.2	99.7/96.3	**100**/**99.6**	98.2/98.3	95.1/96.2	**100**/99.0	**100**/99.2	98.5/97.9
	Texture	Carpet	97.0/95.5	98.7/99.0	98.9/98.8	99.3/96.3	**100**/**99.6**	94.2/97.6	99.4/98.6	98.3/98.1	**100**/99.4	99.0/98.5
		Grid	99.9/**99.7**	98.2/98.7	**100**/97.0	99.9/94.1	**100**/**99.7**	**100**/99.2	98.5/96.6	**100**/**99.7**	99.8/99.4	**100**/99.6
		Leather	**100**/98.6	**100**/99.3	**100**/98.6	**100**/97.7	**100**/**99.8**	98.5/99.4	99.8/98.8	**100**/99.1	**100**/99.7	**100**/**99.8**
		Tile	99.6/99.2	98.7/95.6	99.3/98.9	99.9/91.5	99.2/97.8	95.5/94.7	96.8/92.4	**100**/**99.4**	**100**/98.2	**100**/98.7
		Wood	99.1/96.4	99.2/95.0	99.2/**99.3**	99.5/90.9	**100**/97.9	99.7/95.9	99.7/93.3	**100**/96.7	**100**/97.1	99.4/98.4
	Avg.		98.0/97.3	99.1/98.1	98.5/97.8	99.1/96.9	98.9/98.7	97.2/97.4	97.2/96.8	98.7/98.3	**99.7**/**98.8**	99.3/98.6
VisA Dataset		Candle	89.6/91.0	98.7/99.2	94.3/98.7	98.4/99.1	**99.9**/**99.3**	95.6/-	92.8/97.3	90.4/-	97.7/98.3	**99.9**/94.8
		Capsules	89.2/99.0	68.8/96.5	90.8/99.4	93.5/98.2	98.7/**99.6**	88.5/-	58.2/97.3	87.6/-	98.0/96.2	**99.1**/**99.6**
		Cashew	88.3/85.0	97.7/**99.2**	97.4/94.1	97.2/**99.2**	96.9/96.2	94.2/-	91.5/90.9	81.4/-	94.9/98.7	**98.4**/97.0
		Chewinggum	96.4/97.7	99.1/98.9	98.4/97.4	**99.9**/99.2	99.8/99.4	99.7/-	99.1/94.7	94.0/-	93.6/**99.7**	99.6/99.1
		Fryum	94.7/82.5	91.6/95.9	96.2/96.7	96.5/96.5	**99.9**/97.6	96.5/-	89.8/97.6	87.1/-	88.2/**99.6**	99.4/96.9
		Macaroni1	93.9/99.4	90.1/98.5	98.6/99.6	99.4/**99.9**	99.7/99.3	94.3/-	85.7/94.1	87.6/-	97.6/97.6	**99.9**/99.8
		Macaroni2	88.3/99.7	63.4/93.5	89.5/99.2	96.7/99.8	97.1/99.5	92.5/-	62.5/93.6	90.7/-	98.0/89.5	**98.9**/**99.8**
		Pcb1	84.7/98.4	96.0/**99.8**	97.1/99.7	98.5/**99.8**	**99.7**/99.7	97.7/-	88.1/98.7	75.0/-	96.0/98.9	99.6/99.6
		Pcb2	96.2/94.0	95.1/98.4	97.0/98.6	99.5/**99.3**	99.5/98.1	98.3/-	91.4/95.2	94.6/-	93.5/97.8	**100**/98.6
		Pcb3	97.4/94.3	93.0/98.9	96.4/99.2	98.9/**99.4**	99.5/98.3	97.4/-	86.2/96.7	94.7/-	94.4/98.9	**99.9**/98.9
		Pcb4	98.9/97.6	99.5/98.3	**99.9**/97.7	98.9/99.1	**99.9**/98.7	99.8/-	99.6/97.0	97.7/-	93.0/**99.5**	**99.9**/**99.5**
		Pipe fryum	94.7/65.8	99.0/99.3	94.6/98.7	99.7/99.3	**99.9**/97.5	96.9/-	96.2/**99.4**	92.7/-	97.0/98.9	98.9/**99.4**
	Avg.		92.4/92.0	91.0/98.1	95.8/98.3	98.1/**99.1**	99.2/98.6	96.0/97.9	86.8/96.0	89.5/-	95.2/97.8	**99.5**/98.6
Avg. All			95.6/94.9	95.4/98.0	97.3/98.0	98.6/97.8	99.0/**98.7**	96.6/97.4	92.5/96.4	94.6/-	97.7/98.3	**99.4**/98.6

Table 1: Comparison of anomaly detection and localization performance on MVTec AD and VisA. Each entry reports I-AUROC / P-AUROC (%). **Bold** and underlined numbers indicate the **best** and second-best results, respectively.

Training Strategy

Our framework is trained in three independent stages on normal data to ensure stability and modular effectiveness. Note that all components are trained from scratch:

- **Stage 1:** Train RcAE with $\mathcal{L}_{\text{rec}}$. To prevent shortcut learning and overfitting to a fixed recursion depth, the recursion depth is randomly selected from $[1, N]$ per batch.
- **Stage 2:** Freeze RcAE, train DPN with loss $\mathcal{L}_{\text{DPN}}$ to restore high-frequency details via $\mathbf{I}_D^n = \mathbf{I}_R^n + \mathbf{Res}_D^n$.
- **Stage 3:** Freeze RcAE and DPN, train CRD with $\mathcal{L}_{\text{CRD}}$ using $\mathbf{I}$ and $\{\mathbf{I}_D^n\}$ to predict anomaly map $\mathbf{M}_A$. Pseudo masks $\mathbf{M}_P$ are generated via lightweight augmentations.

Augmentation: We adopt simple perturbations in Stages 1 and 3, including: (1) Random color blocks, (2) Copy & paste patches, (3) Random lines (1–4) of length 50–150 forming crack-like structures. These are applied to random blocks of size 32, 64, or 128 with varying coverage (0–100%).

Experiments

We comprehensively evaluate the proposed method on both image-level and pixel-level anomaly detection tasks, and analyze computational efficiency. Comparisons are made on the MVTec AD (Bergmann et al. 2021) and VisA (Zou et al. 2022) datasets with representative approaches across three categories: (1) Non-diffusion methods: DRAEM (Zavrtanik, Kristan, and Skocaj 2021), PatchCore (Roth et al. 2022), RD4AD (Deng and Li 2022), EfficientAD (Batzner, Heckler, and König 2024); (2) Flow-based methods: MSFlow (Zhou et al. 2025); (3) Diffusion-based methods: D3AD (Tebbe and Tayyub 2024), DiAD (He et al. 2024), DiffAD (Zhang et al. 2023), and GLAD (Yao et al. 2025).

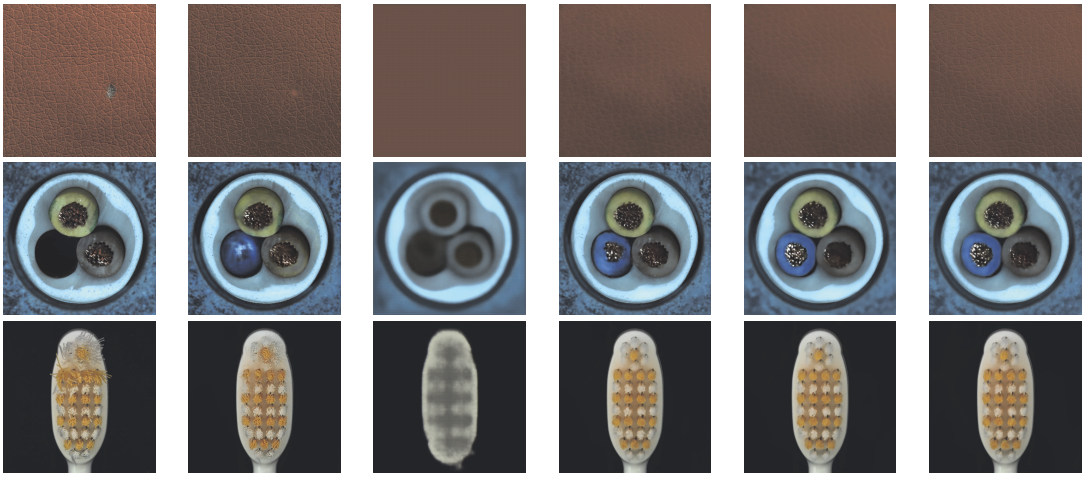
Evaluation Metrics: Following common practice, we report image-level AUROC (I-AUROC) and pixel-level AUROC (P-AUROC) as primary evaluation metrics.

Training Setup: We train all components from scratch using Adam optimizer (η=10^{-4}, β_1=0.9, β_2=0.999, ϵ=10^{-8}). Stage 1/2/3 are trained for 1500/400/300 epochs, respectively, with recursion depth N=5 and input size 1024×1024. Experiments are conducted on NVIDIA RTX4090 with Python 3.10, and all baselines follow their official settings.

Anomaly Detection and Localization

Table 1 summarizes the anomaly detection and localization performance on the MVTec AD and VisA datasets. Our method consistently delivers strong results in both image and pixel level metrics while maintaining high efficiency.

On MVTec AD, our method achieves an average of 98.9% I-AUROC and 98.7% P-AUROC, outperforming



(a) Inputs (b) ConvAE (c) RcAE w/o shared weight (d) RcAE (N=3) w/o DPN (e) RcAE (N=5) w/o DPN (f) RcAE (N=5) w/ DPN

Figure 3: Qualitative ablation on reconstruction. (a) Input. (b) ConvAE: blurry outputs with residual anomalies. (c) RcAE w/o weight sharing: partial anomaly memorization and detail loss. (d) RcAE (N=3): significantly improved reconstruction but anomalies remain. (e) RcAE (N=5): stronger suppression with slight texture erosion. (f) RcAE (N=5) + DPN: fine textures restored while anomalies remain suppressed.

non-diffusion baselines such as RD4AD (98.5/97.8%) and DRAEM (98.0/97.3%), and also surpassing diffusion models like D3AD (97.2/97.4%), DiAD (97.2/96.8%), and DiffAD (98.7/98.3%). Notably, we achieve higher pixel-level accuracy than GLAD (98.7% vs. 98.6%), despite GLAD utilizing a large vision model (DINO (Caron et al. 2021)) with latent diffusion. This highlights the effectiveness of our pipeline, which achieves high performance without relying on pretrained models or heavy designs.

On the more challenging VisA dataset, which features complex object structures and diverse anomaly types, many methods exhibit performance degradation. Our method achieves 99.2% I-AUROC and 98.6% P-AUROC, tying GLAD for the second-best localization performance and ranking second in detection. It ranks among the top-2 in 10 out of 12 categories, highlighting strong robustness under realistic and challenging conditions.

Overall, our method achieves the **highest avg. P-AUROC (98.7%)** and the **second-highest I-AUROC (99.0%)** across both datasets. These results validate the effectiveness of our recursive reconstruction pipeline, highlighting our approach is highly suitable for practical industrial applications.

Ablation Study

Effectiveness of Core Components. Table 2 presents a step-wise ablation. Starting from a plain ConvAE baseline (82.4/90.8% I-/P-AUROC), introducing RcAE already lifts performance to 94.1/95.8% by progressively suppressing anomalies and refining normal structures. Adding the DPN improves results to 95.7/96.6% by selectively restoring high-frequency details that may be smoothed during recursive reconstruction. Finally, integrating CRD yields the full model with 98.9/98.7%, leveraging cross-recursion residual dynamics for robust anomaly localization. Fig. 3 provides qualitative evidence: ConvAE outputs are blurry with residual defects, RcAE progressively normalizes anomalies, and DPN restores textures without reintroducing them.

RcAE	DPN	CRD	Accuracy
✗	✗	✗	82.4/90.8
✓	✗	✗	94.1/95.8
✓	✓	✗	95.7/96.6
✓	✓	✓	**98.9/98.7**

Table 2: Ablation on core components of our method. Each entry shows I-AUROC / P-AUROC (%) on MVTec AD.

Impact of Recursion Depth of RcAE. We analyze the effect of recursion depth N in Table 3. At N=1, RcAE equals to two ConvAEs with limited capacity (86.2/87.4%). Increasing N strengthens anomaly suppression and semantic refinement, with the largest gain between N=1 and N=3, and a peak at N=5. Beyond this, performance plateaus or slightly declines, suggesting diminishing returns and minor over-smoothing. This validates the benefit of recursive architecture in reconstruction quality and exposing anomalies. Fig. 3 illustrates these trends qualitatively, and more reconstruction results can be seen in Supplementary Materials.

Dataset	Number of Iterations (N)					
	N=1	N=2	N=3	N=4	N=5	N=6
MVTec	86.2/87.4	90.8/89.9	96.3/96.8	98.0/98.3	**98.9/98.7**	98.7/98.4
VisA	89.3/88.7	93.2/92.1	97.3/96.4	98.9/98.1	**99.2/98.6**	**99.2**/98.1

Table 3: Effect of RcAE recursion depth N on anomaly detection (I-AUROC / P-AUROC (%)).

Impact of RcAE Architecture. Table 4 shows how skip connections and parameter sharing affect RcAE performance. Replacing the basic ConvAE (65.2/72.4%) with a recursive design significantly improves accuracy, and skip connections further boosts it to 98.9/98.7%. Conversely, removing weight sharing drops performance to 71.3/74.2%, highlighting the importance of both recursion and constrained parameterization. Notably, for anomaly detection, skip connections often lead to shortcut learning in ConvAE. In RcAE, the repeated compression–reconstruction suppress shortcuts, while skip connections enhance shallow feature propagation, improving reconstruction quality.

Model	Skip Conn.	Shared Weights	Accuracy	
			MVTec AD	VisA
ConvAE	✓	✗	65.2 / 72.4	68.5 / 70.1
ConvAE	✗	✗	82.4 / 90.8	79.7 / 78.5
RcAE	✗	✓	92.2 / 95.2	94.3 / 95.9
RcAE	✓	✗	71.3 / 74.2	73.5 / 75.0
RcAE	✓	✓	**98.9 / 98.7**	**99.2 / 98.6**

Table 4: Effect of skip connections and parameter sharing in RcAE (I-AUROC / P-AUROC (%)).

Data Efficiency. Table 5 shows that RcAE consistently outperforms ConvAE across all training data ratios. With only 10% of the training data, RcAE already surpasses the full-data ConvAE, demonstrating superior data efficiency, which is important for industrial scenarios with scarce samples.

Method	Training Data Percentage				
	10%	25%	50%	75%	100%
ConvAE	62.3/61.5	68.9/77.2	71.4/87.8	79.1/88.9	82.4/90.8
RcAE	**84.1/93.4**	**91.2/93.7**	**95.5/95.1**	**97.2/96.8**	**98.9/98.7**

Table 5: Impact of training data size on anomaly detection performance on MVTec AD (I-AUROC / P-AUROC (%)).

Effectiveness of Detail Preservation Network. Detail Preservation Network mitigates detail loss that can lead to false positives. Table 2 shows that adding DPN improves I-/P-AUROC from 94.1/95.8% to 95.7/96.6%. It also improves reconstruction quality with average SSIM and PSNR gains of 0.059 and 0.61 dB (up to 0.32 and 2.61 dB).

As shown in Fig. 3(e-f), DPN selectively restores details while maintaining anomaly suppression, especially on highly textured samples (e.g., leather), confirming that DPN preserves details without reintroducing anomalies.

N_R	Recon steps fed to CRD	I-AUROC / P-AUROC
1	{5}	95.7 / 96.6
3	{1, 3, 5}	98.0 / 97.1
5	**{1–5}**	**98.9 / 98.7**

Table 6: Impact of the number of RcAE reconstructions used by CRD (I-AUROC / P-AUROC (%) on MVTec AD).

Effectiveness of Cross Recursion Detection. We fix $N{=}5$ in RcAE and vary the number of reconstructions N_R fed into CRD (Table 6). Using only the final reconstruction ($N_R{=}1$, step 5) provides a strong baseline (95.7/96.6%). Incorporating intermediate reconstructions ($N_R{=}3$, steps 1/3/5) improves results to 98.0/97.1%, and using all steps ($N_R{=}5$, steps 1–5) achieves 98.9/98.7%. This demonstrates that leveraging reconstruction dynamics across recursion steps yields more reliable anomaly localization than a single residual map.

Computational Complexity

As shown in Fig. 4, our method achieves a favorable balance between accuracy and efficiency. The recursive architecture slightly increases inference time compared to single-pass ConvAEs , but it maintains a compact parameter count with high performance and remains much faster than diffusion. Despite this lightweight design without pretraining or external priors, the accuracy is on par with GLAD, which requires both latent diffusion and DINO. This combination of high performance, small size, and good latency makes our method practical for real-world applications.

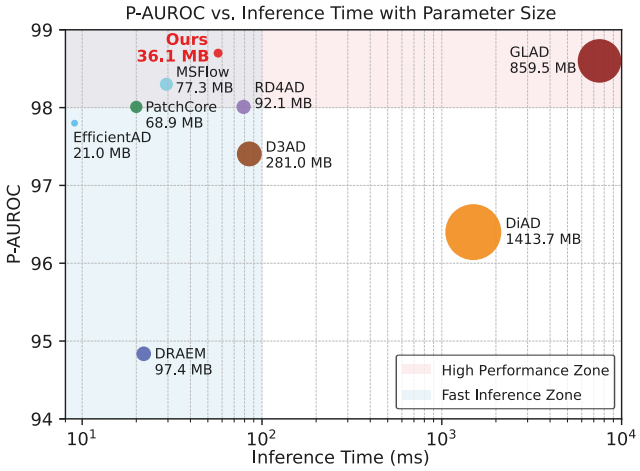


Figure 4: Trade-off between computation cost and accuracy. Circle size indicates model parameter count.

Conclusion

We proposed an efficient recursive autoencoder framework for industrial anomaly detection. By sharing parameters across iterations, the model progressively refines reconstructions without increasing model size and leverages cross-recursion dynamics for robust anomaly localization. Trained entirely from scratch without external priors, our method achieves state-of-the-art performance with practical computational efficiency. Despite these advantages, the current design may be less effective for high-level logical anomalies that require semantic reasoning. Future work will explore integrating lightweight prior knowledge or hybrid architectures to address such cases, and extending the recursive paradigm to broader industrial vision tasks facing similar efficiency and data limitations.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62306223), the China Postdoctoral Science Foundation (No. 2024M752533), the Young Talent Fund of Xi'an Association for Science and Technology (No. 959202413053), the Fundamental Research Funds for the Central Universities (No. XJSJ24020), the Postdoctoral Science Foundation of Shaanxi Province, the Xiaomi Young Talents Program, and the Digital Finance CRC (supported by the Cooperative Research Centres program, an Australian Government initiative).

References

Akcay, S.; Atapour-Abarghouei, A.; and Breckon, T. P. 2019. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *ACCV*, 622–637. Springer.

Akçay, S.; Atapour-Abarghouei, A.; and Breckon, T. P. 2019. Skip-ganomaly: Skip connected and adversarially trained encoder-decoder anomaly detection. In *International Joint Conference on Neural Networks (IJCNN)*, 1–8.

Batzner, K.; Heckler, L.; and König, R. 2024. EfficientAD: Accurate visual anomaly detection at millisecond-level latencies. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 128–138.

Bergmann, P.; Batzner, K.; Fauser, M.; Sattlegger, D.; and Steger, C. 2021. The MVTec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. *IJCV*, 129(4): 1038–1059.

Bergmann, P.; Batzner, K.; Fauser, M.; Sattlegger, D.; and Steger, C. 2022. Beyond dents and scratches: Logical constraints in unsupervised anomaly detection and localization. *IJCV*, 130(4): 947–969.

Bergmann, P.; Löwe, S.; Fauser, M.; Sattlegger, D.; and Steger, C. 2018. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. *arXiv preprint arXiv:1807.02011*.

Cai, Y.; Chen, H.; and Cheng, K.-T. 2024. Rethinking autoencoders for medical anomaly detection from a theoretical perspective. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 544–554. Springer.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9650–9660.

Chandola, V.; Banerjee, A.; and Kumar, V. 2009. Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 41(3): 1–58.

Deng, H.; and Li, X. 2022. Anomaly detection via reverse distillation from one-class embedding. In *CVPR*, 9737–9746.

Denouden, T.; Salay, R.; Czarnecki, K.; Abdelzad, V.; Phan, B.; and Vernekar, S. 2018. Improving reconstruction autoencoder out-of-distribution detection with mahalanobis distance. *arXiv preprint arXiv:1812.02765*.

Fučka, M.; Zavrtanik, V.; and Skočaj, D. 2024. TransFusion–a transparency-based diffusion model for anomaly detection. In *ECCV*, 91–108. Springer.

Gong, D.; Liu, L.; Le, V.; Saha, B.; Mansour, M. R.; Venkatesh, S.; and Hengel, A. v. d. 2019. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *ICCV*, 1705–1714.

He, H.; Zhang, J.; Chen, H.; Chen, X.; Li, Z.; Chen, X.; Wang, Y.; Wang, C.; and Xie, L. 2024. A diffusion-based framework for multi-class anomaly detection. In *AAAI*, volume 38, 8472–8480.

Kingma, D. P.; Welling, M.; et al. 2019. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4): 307–392.

Li, C.-L.; Sohn, K.; Yoon, J.; and Pfister, T. 2021. Cutpaste: Self-supervised learning for anomaly detection and localization. In *CVPR*, 9664–9674.

Liang, Y.; Zhang, J.; Zhao, S.; Wu, R.; Liu, Y.; and Pan, S. 2023. Omni-frequency channel-selection representations for unsupervised anomaly detection. *IEEE TIP*.

Liu, W.; Li, R.; Zheng, M.; Karanam, S.; Wu, Z.; Bhanu, B.; Radke, R. J.; and Camps, O. 2020. Towards visually explaining variational autoencoders. In *CVPR*, 8642–8651.

Mishra, P.; Verk, R.; Fornasier, D.; Piciarelli, C.; and Foresti, G. L. 2021. VT-ADL: A vision transformer network for image anomaly detection and localization. In *IEEE International Symposium on Industrial Electronics*, 01–06.

Pang, G.; Shen, C.; Cao, L.; and Hengel, A. V. D. 2021. Deep Learning for Anomaly Detection: A Review. *ACM Comput. Surv.*, 54(2).

Park, H.; Noh, J.; and Ham, B. 2020. Learning memory-guided normality for anomaly detection. In *CVPR*, 14372–14381.

Pirnay, J.; and Chai, K. 2022. Inpainting transformer for anomaly detection. In *International Conference on Image Analysis and Processing*, 394–406.

Ristea, N.-C.; Madan, N.; Ionescu, R. T.; Nasrollahi, K.; Khan, F. S.; Moeslund, T. B.; and Shah, M. 2022. Self-supervised predictive convolutional attentive block for anomaly detection. In *CVPR*, 13576–13586.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *CVPR*, 10684–10695.

Roth, K.; Pemula, L.; Zepeda, J.; Schölkopf, B.; Brox, T.; and Gehler, P. 2022. Towards total recall in industrial anomaly detection. In *CVPR*, 14318–14328.

Ruff, L.; Kauffmann, J. R.; Vandermeulen, R. A.; Montavon, G.; Samek, W.; Kloft, M.; Dietterich, T. G.; and Müller, K.-R. 2021. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5): 756–795.

Tao, X.; Gong, X.; Zhang, X.; Yan, S.; and Adak, C. 2022. Deep learning for unsupervised anomaly localization in industrial images: A survey. *IEEE Transactions on Instrumentation and Measurement*, 71: 1–21.

Tebbe, J.; and Tayyub, J. 2024. Dynamic Addition of Noise in a Diffusion Model for Anomaly Detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 3940–3949.

Wyatt, J.; Leach, A.; Schmon, S. M.; and Willcocks, C. G. 2022. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *CVPR*, 650–656.

Xu, J.; Wu, H.; Wang, J.; and Long, M. 2021. Anomaly transformer: Time series anomaly detection with association discrepancy. *arXiv preprint arXiv:2110.02642*.

Yao, H.; Liu, M.; Yin, Z.; Yan, Z.; Hong, X.; and Zuo, W. 2025. GLAD: Towards better reconstruction with global and local adaptive diffusion models for unsupervised anomaly detection. In *ECCV*, 1–17.

You, Z.; Cui, L.; Shen, Y.; Yang, K.; Lu, X.; Zheng, Y.; and Le, X. 2022. A Unified Model for Multi-class Anomaly Detection. In *NeurIPS*, volume 35, 4571–4584.

Zavrtanik, V.; Kristan, M.; and Skocaj, D. 2021. DRAEM - A Discriminatively Trained Reconstruction Embedding for Surface Anomaly Detection. In *ICCV*, 8330–8339.

Zavrtanik, V.; Kristan, M.; and Skočaj, D. 2021. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112: 107706.

Zhang, X.; Li, N.; Li, J.; Dai, T.; Jiang, Y.; and Xia, S.-T. 2023. Unsupervised surface anomaly detection with diffusion probabilistic model. In *ICCV*, 6782–6791.

Zhang, X.; Xu, M.; and Zhou, X. 2024. RealNet: A Feature Selection Network with Realistic Synthetic Anomaly for Anomaly Detection. In *CVPR*, 16699–16708.

Zhou, Y.; Xu, X.; Song, J.; Shen, F.; and Shen, H. T. 2025. MSFlow: Multiscale Flow-Based Framework for Unsupervised Anomaly Detection. *IEEE TNNLS*, 36(2): 2437–2450.

Zou, Y.; Jeong, J.; Pemula, L.; Zhang, D.; and Dabeer, O. 2022. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *ECCV*, 392–408.