Unsupervised Multi-View Visual Anomaly Detection via Progressive Homography-Guided Alignment

Xintao Chen[1*], Xiaohao Xu[2*], Bozhong Zheng[1], Yun Liu[1], Yingna Wu[1†]

[1] ShanghaiTech University
[2]University of Michigan, Ann Arbor

Abstract

Unsupervised visual anomaly detection from multi-view images presents a significant challenge: distinguishing genuine defects from benign appearance variations caused by viewpoint changes. Existing methods, often designed for single-view inputs, treat multiple views as a disconnected set of images, leading to inconsistent feature representations and a high false-positive rate. To address this, we introduce ViewSense-AD (VSAD), a novel framework that learns viewpoint-invariant representations by explicitly modeling geometric consistency across views. At its core is our Multi-View Alignment Module (MVAM), which leverages homography to project and align corresponding feature regions between neighboring views. We integrate MVAM into a View-Align Latent Diffusion Model (VALDM), enabling progressive and multi-stage alignment during the denoising process. This allows the model to build a coherent and holistic understanding of the object's surface from coarse to fine scales. Furthermore, a lightweight Fusion Refiner Module (FRM) enhances the global consistency of the aligned features, suppressing noise and improving discriminative power. Anomaly detection is performed by comparing multi-level features from the diffusion model against a learned memory bank of normal prototypes. Extensive experiments on the challenging RealIAD and MANTA datasets demonstrate that VSAD sets a new state-of-the-art, significantly outperforming existing methods in pixel, view, and sample-level visual anomaly detection, proving its robustness to large viewpoint shifts and complex textures. Our code will be released to drive further research.

1 Introduction

Industrial anomaly detection is a critical task in modern manufacturing, where even minuscule defects can compromise product quality, lead to costly recalls and pose safety risks(Cao et al. 2024). While most unsupervised anomaly detection methods rely on single-view imagery, complex 3D objects often feature occlusions or intricate geometries that a single viewpoint cannot fully capture. Consequently, multi-view imaging systems, which capture an object from several fixed perspectives, have become a practical and effective solution for ensuring comprehensive surface inspection.

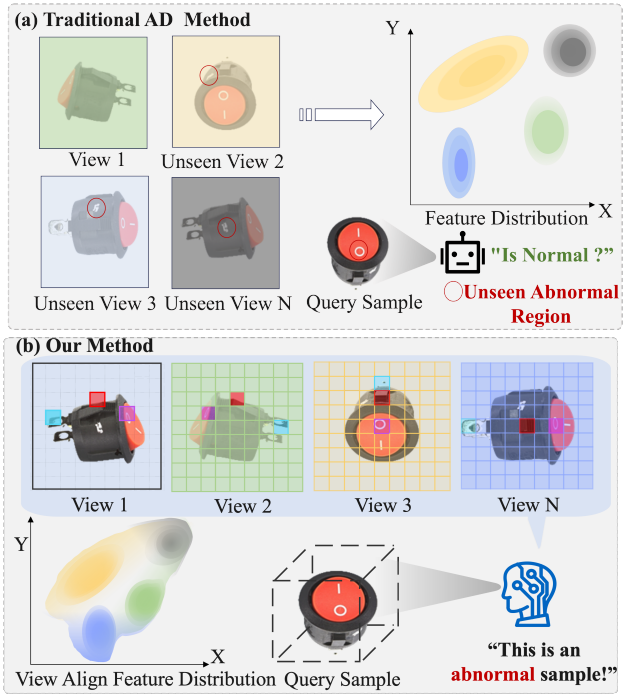

Figure 1: **(a) Conventional methods** process views independently, yielding discrete and inconsistent features that struggle to differentiate viewpoint changes from true defects. **(b) Our method (VSAD)** employs homography-based alignment to establish correspondences between views, learning a continuous and consistent representation that enables robust anomaly detection.

However, transitioning from single-view to multi-view settings introduces a fundamental challenge: distinguishing true anomalies from appearance shifts induced purely by changes in viewpoint (Fig. 1a). Existing unsupervised methods, whether reconstruction-based (e.g., DRAEM (Zavrtanik, Kristan, and Skočaj 2021)) or embedding-based (e.g., PatchCore (Roth et al. 2022)), typically process each view independently. This 'bag-of-views' approach ignores the underlying geometric relationships between images, resulting in feature representations that are fragmented and

*These authors contributed equally.
†Corresponding authors

unaligned. As a result, these models are prone to misinterpreting normal geometric variations as anomalies, leading to poor performance and reliability in real-world scenarios.

To overcome these limitations, we propose **ViewSense-AD (VSAD)**, an unsupervised framework designed to learn continuous and consistent cross-view representations (Fig. 1b). Our work is inspired by how human inspectors naturally operate: they mentally align different views of an object to build a holistic understanding of its surface, effortlessly distinguishing surface texture from geometric perspective shifts. VSAD mimics this reasoning process through a synergistic design. First, to determine where to look for corresponding information, we introduce a Multi-View Alignment Module (MVAM). It uses homography-based projection to explicitly match related feature patches across adjacent views. Second, to learn how to integrate this aligned information, the MVAM is embedded within a View-Align Latent Diffusion Model (VALDM). By performing alignment progressively during the denoising process, VALDM constructs a viewpoint-invariant representation. Finally, to refine the holistic alignment representation, a lightweight Fusion Refiner Module (FRM) explicitly models cross-view consistency to suppress noise and sharpen the distinction between normal and anomalous patterns.

During inference, we extract multi-level features from the DDIM inversion process and compare them against a memory bank of normal prototypes for fine-grained, multi-scale anomaly localization. Our extensive experiments on the RealIAD and MANTA datasets show that VSAD consistently outperforms state-of-the-art baselines across all evaluation levels. These results validate that by explicitly modeling geometric consistency, our framework effectively bridges the gap between fragmented image-level processing and holistic, human-like perception in multi-view anomaly detection.

Our contributions are summarized as follows:

- We propose **VSAD**, a novel unsupervised multi-view anomaly detection framework that learns continuous and consistent cross-view representations through homography-guided alignment.
- We design a homography-based **MVAM** and embed it into a **VALDM** for progressive inter-view alignment, whose output is further enhanced by a lightweight **FRM** that refines global consistency.
- VSAD achieves new **state-of-the-art performance** on the large-scale RealIAD and MANTA benchmarks, demonstrating superior robustness and generalization in challenging multi-view industrial scenarios.

2 Related Work

2.1 Unsupervised Anomaly Detection

Unsupervised anomaly detection methods learn from anomaly-free data and are categorized as reconstruction-based or embedding-based (Fan et al. 2025b; Zheng et al. 2025; Li et al. 2025). Reconstruction-based methods, like autoencoders, VAE and GANs, identify anomalies as regions with high reconstruction error. More recent works have improved reconstruction fidelity using memory modules (Gong et al. 2019; Cai et al. 2021), pseudo-anomaly augmentation (Zavrtanik, Kristan, and Skočaj 2021; Hu et al. 2024; Sun et al. 2025), and diffusion models (Zavrtanik, Kristan, and Skočaj 2023; Kim et al. 2024; Yao et al. 2025). However, when applied to multi-view data, they typically reconstruct each view in isolation, failing to enforce cross-view consistency.

Embedding-based methods leverage powerful features from models pre-trained on large datasets like ImageNet (Deng et al. 2009). They model the distribution of normal features using memory banks (Roth et al. 2022; Bae, Lee, and Kim 2023; Liu et al. 2025), normalizing flows (Gudovskiy, Ishizaka, and Kozuka 2022; Yao et al. 2024), or student-teacher networks (Bergmann et al. 2020; Liu et al. 2024; Wang et al. 2025a). While highly effective for single-view tasks, these methods inherently lack a mechanism to account for the geometric transformations between views, making them susceptible to feature misalignment and inconsistency in multi-view settings.

2.2 Multi-view Feature Alignment

Aligning features across multiple views is fundamental in computer vision, with applications in novel view synthesis (Gao et al. 2024; Zhang et al. 2025), 3D perception (Banerjee et al. 2025), and autonomous driving. Common strategies include Transformer-based fusion using self- or cross-attention (Wu et al. 2023; Daryani et al. 2025), epipolar geometry constraints (Sun et al. 2021; Chang et al. 2024; Wang et al. 2025b), and homography-based alignment (He et al. 2020; Hwang, Benz, and Kim 2024; Ni et al. 2025). These techniques aim to create a unified representation by establishing spatial or semantic correspondences. However, their direct application to unsupervised anomaly detection is non-trivial. Most existing AD frameworks do not explicitly align multi-view data. Our work addresses this gap by introducing a lightweight and effective homography-based alignment mechanism tailored for industrial inspection scenarios, where objects are often captured from fixed viewpoints or on a turntable.

2.3 Diffusion-based Models for Anomaly Detection

Diffusion models (Ho, Jain, and Abbeel 2020) have demonstrated state-of-the-art performance in image generation. Their ability to produce high-fidelity reconstructions has been leveraged for anomaly detection. AnoDDPM (Wyatt et al. 2022) and other methods (He et al. 2024; Akshay et al. 2025) use the denoising process to reconstruct an anomaly-free version of a test image, with anomalies detected from the reconstruction residual. Others use diffusion models to synthesize diverse defects for training more discriminative models (Zhang et al. 2023b; Hu et al. 2024; Song et al. 2025). While effective, these methods primarily focus on single-image reconstruction. In contrast, our work extracts multi-level decoder features during the DDIM (Song, Meng, and Ermon 2020) inversion process, not for reconstruction, but as rich descriptors for a fine-grained, embedding-based anomaly detection approach.

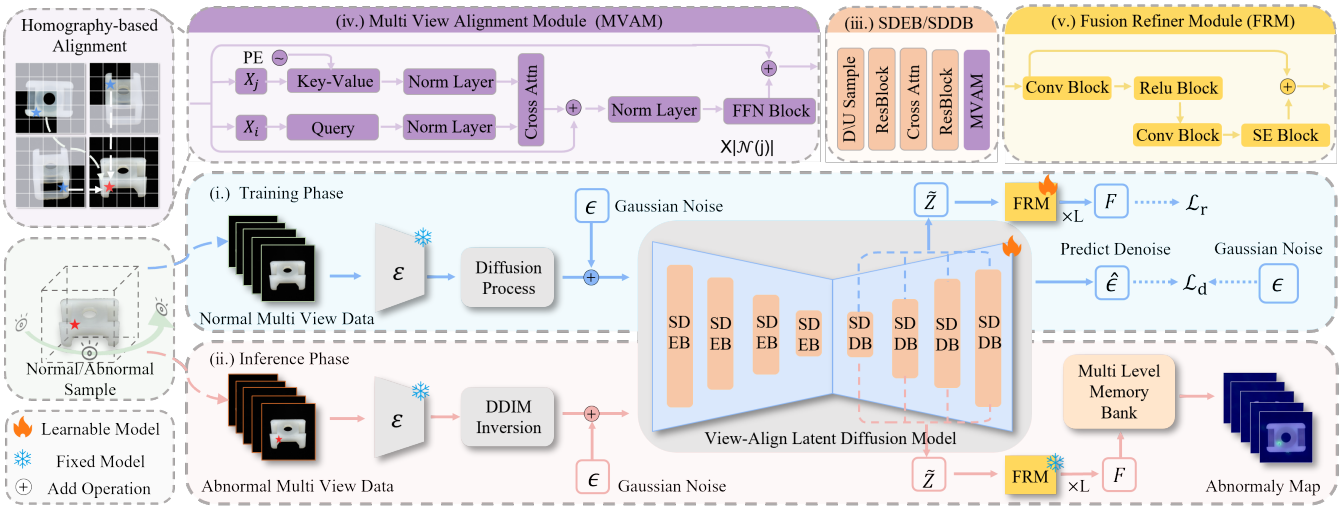

Figure 2: **Overall architecture of ViewSense-AD (VSAD)**. (i.) During training, multi-view images are encoded into latent space. The **View-Align Latent Diffusion Model (VALDM)** performs progressive denoising, where at each Unet layer, the **Multi-View Alignment Module (MVAM)** aligns features from neighboring views using homography. A **Fusion Refiner Module (FRM)** then enhances global consistency. The model is trained with a denoising loss $\mathcal{L}_{\text{d}}$ and a refinement loss $\mathcal{L}_{\text{r}}$. (ii.) At inference, multi-level refined features are extracted via DDIM inversion and compared against a normal memory bank for anomaly scoring. (iii.) The architecture of the UNet encoder/decoder block used in Stable Diffusion. The proposed MVAM module is integrated after the ResBlock. (iv.) Detailed architecture of the MVAM. (v.) Detailed architecture of the FRM.

3 Method

3.1 Problem Formulation

In unsupervised multi-view anomaly detection, we are given a training set $\mathcal{D}_C = \{S_n\}_{n=1}^{N}$ for an object category C. Each sample $S_n = \{I_m\}_{m=1}^{M}$ consists of M RGB images captured from different viewpoints, where $I_m \in \mathbb{R}^{3\times H\times W}$. The training set contains only anomaly-free samples. The goal is to learn a function that can identify and localize anomalies in a test sample S_q. This involves generating a pixel-wise anomaly map for each view, an anomaly score for each view, and an overall score for the sample.

3.2 Overall Architecture

We propose VSAD, an unsupervised framework designed around the principles of explicit alignment, progressive understanding, and global refinement. As illustrated in Figure 2, the framework is composed of several key components. The core is a View-Align Latent Diffusion Model (VALDM) that learns the distribution of normal multi-view samples. To enable viewpoint-invariant learning, we embed our Multi-View Alignment Module (MVAM) into each layer of the model's U-Net backbone. This module uses homography to explicitly align features from neighboring views. Following alignment at each decoder stage, a lightweight Fusion Refiner Module (FRM) refines the fused features by enhancing global consistency. For detection, we use a multi-level memory bank. At inference, features are extracted from the decoder via DDIM inversion, refined, and compared against a memory bank of normal prototypes to enable robust, fine-grained anomaly scoring.

3.3 Multi-View Alignment Module (MVAM)

Industrial multi-view capture setups, often using fixed cameras or turntables, produce images with significant spatial overlap and smooth appearance transitions. To exploit this, we propose MVAM for patch-level feature alignment.

Given a set of multi-view feature maps $\mathbf{X} \in \mathbb{R}^{M\times C\times H\times W}$ from a U-Net layer, we define neighboring view pairs for each view i as $\mathcal{P}(X_i) = \{(X_i, X_j) \mid j \in \mathcal{N}(i)\}$, where $\mathcal{N}(i)$ is the set of adjacent view indices. For each patch centered at position p_i in the query view X_i, we project its location into each neighboring view X_j using a pre-computed homography matrix $H_{i\to j}$. Around the projected location $p_j = H_{i\to j} \cdot p_i$, we sample a local search window of size $R \times R$ to find the best-matching patch.

For each patch at location p_j in the search window, we compute its relative displacement from the query patch p_i after projection: $\Delta\mathbf{p}_j = p_j - H_{i\to j} \cdot p_i$. This offset is encoded using a standard 2D frequency-based positional embedding (Vaswani et al. 2017) to form $\gamma(\Delta\mathbf{p}_j)$. We then construct query, key, and value representations for attention-based aggregation:

$$q_i = W_q \cdot X_i(p_i) \tag{1}$$

$$k_j = W_k \cdot (X_j(p_j) + \gamma(\Delta\mathbf{p}_j)) \tag{2}$$

$$v_j = W_v \cdot (X_j(p_j) + \gamma(\Delta\mathbf{p}_j)) \tag{3}$$

where W_q, W_k, W_v are learnable projection matrices. An attention mechanism computes weights α_j to aggregate the value vectors, producing an aligned feature $\tilde{X}_i(p_i)$:

$$\alpha_j = \frac{\exp\left(q_i^\top k_j/\sqrt{d}\right)}{\sum_{j'\in\mathcal{N}(i)} \exp\left(q_i^\top k_{j'}/\sqrt{d}\right)} \tag{4}$$

$$\tilde{X}_i(p_i) = \sum_{j \in \mathcal{N}(i)} \alpha_j \cdot v_j \tag{5}$$

This process is applied to all patches, yielding an aligned feature map $\tilde{\mathbf{X}} \in \mathbb{R}^{M \times C \times H \times W}$ with enhanced cross-view continuity.

3.4 View-Align Latent Diffusion Model (VALDM)

To achieve progressive understanding, we embed MVAM into a latent diffusion model based on the DDIM formulation. Given multi-view images $I = \{I_m\}_{m=1}^{M}$, a VAE encoder $\mathcal{E}_{\text{VAE}}$ produces latent representations $Z_0 = \mathcal{E}_{\text{VAE}}(I)$. The DDIM forward process adds Gaussian noise to produce a noisy latent Z_t at timestep t:

$$Z_t = \sqrt{\bar{\alpha}_t} Z_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbf{I}) \tag{6}$$

The reverse process is handled by a U-Net which predicts the noise $\hat{\epsilon}_t$ from Z_t. We modify the U-Net decoder. At each decoder layer l, the feature map $X^{(l)}$ is first processed by the standard ResNet and attention blocks. The output is then passed to our MVAM to produce an aligned feature map $\tilde{Z}_t^{(l)}$, which is then passed to the next layer.

$$\tilde{Z}_t^{(l)} = \text{MVAM}^{(l)}(\text{U-NetBlock}^{(l)}(Z_t^{(l)})) \tag{7}$$

This multi-stage alignment strategy allows the model to build a coherent representation by progressively aligning features at different semantic levels during the denoising process. The model is trained with a standard denoising objective:

$$\mathcal{L}_{\text{d}} = \mathbb{E}_{Z_0, \epsilon, t}\left[\|\epsilon - \hat{\epsilon}_\theta(Z_t, t)\|_2^2\right] \tag{8}$$

where $\hat{\epsilon}_\theta$ is the noise predicted by our modified U-Net.

3.5 Fusion Refiner Module (FRM)

While MVAM provides explicit local alignment, we introduce the FRM to further enhance global consistency and suppress noise from the fusion process. After the MVAM at each decoder layer l, the aligned features $\tilde{Z}^{(l)}$ are fed into the FRM. For each view m, FRM applies a small convolutional network $f(\cdot)$ followed by a Squeeze-and-Excitation (SE) attention block $\mathcal{A}(\cdot)$ to produce a refinement residual, which is added back to the input:

$$Z_m'^{(l)} = f(\tilde{Z}_m^{(l)}) \odot \mathcal{A}(f(\tilde{Z}_m^{(l)})) \tag{9}$$

$$F_m^{(l)} = \tilde{Z}_m^{(l)} + Z_m'^{(l)} \tag{10}$$

where $F_m^{(l)}$ is the final refined feature for view m at layer l.

To explicitly enforce consistency, we introduce a refinement loss $\mathcal{L}_{\text{r}}$ that minimizes the L2 distance between the refined features of neighboring view pairs (i, j):

$$\mathcal{L}_{\text{r}} = \frac{1}{L}\sum_{l=1}^{L}\frac{1}{|\mathcal{P}|}\sum_{(i,j)\in\mathcal{P}}\|F_i^{(l)} - F_j^{(l)}\|_2^2 \tag{11}$$

where L is the number of decoder layers and $\mathcal{P}$ is the set of all neighboring pairs. The total training loss is $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{d}} + \lambda\mathcal{L}_{\text{r}}$, where λ is a balancing hyperparameter.

3.6 Multi-Level Memory Bank Detection

At inference, we use the trained model as a feature extractor. For a test sample, we perform DDIM inversion for a fixed number of steps. From each of the L decoder layers, we extract the refined features $\{F^{(l)}\}_{l=1}^{L}$. During training, these features from all normal samples are stored in a multi-level memory bank $\mathcal{M} = \{\mathcal{M}^{(l)}\}_{l=1}^{L}$.

For a test sample, its refined features $\{F_q^{(l)}\}_{l=1}^{L}$ are extracted. The anomaly score for a patch at spatial location (u, v) is calculated by finding its minimum distance to the corresponding memory bank, aggregating scores across levels:

$$S_{\text{pixel}}(u, v) = \sum_{l=1}^{L} w_l \min_{m \in \mathcal{M}^{(l)}} \|F_q^{(l)}(u, v) - m\|_2 \tag{12}$$

where w_l are weights for each level. The view-level anomaly score S_{view} is the maximum score in the pixel-level anomaly map, and the sample-level score S_{sample} is the maximum score across all views. This multi-level approach enables robust detection of anomalies at various scales.

4 Experiments

4.1 Experiments Setting

Datasets. We evaluate our method on two challenging multi-view anomaly detection datasets: **Real-IAD** (Wang et al. 2024) consist of 151,050 RGB images across 30 classes with 5 views. **MANTA** (Fan et al. 2025a) contains 137,338 images from 38 categories, each with 5 viewpoints. Training images are normal, while test images include both normal and defective cases.

Evaluation metrics. We evaluate performance using the Area Under the Receiver Operating Characteristic Curve across three levels: 1) **Pixel-level AUROC (P-AUROC)** measures fine-grained anomaly localization, 2) **View-level AUROC (V-AUROC)** evaluates whether an individual view contains anomalies, and 3) **Sample-level AUROC (S-AUROC)** assesses multi-view detection by taking the maximum view-level score across all views within a sample. More comprehensive metric comparisons are provided in the appendix of the supplementary material.

Baseline methods. We compare our method with other unsupervised approaches,including reconstruction based methods Draem(Zavrtanik, Kristan, and Skočaj 2021), CK-AAD(Fang et al. 2025), RealNet(Zhang, Xu, and Zhou 2024) and embedding based methods PatchCore(Roth et al. 2022), CFlow(Gudovskiy, Ishizaka, and Kozuka 2022), DeSTSeg(Zhang et al. 2023a), RDPP(Tien et al. 2023), FiCo(Chen et al. 2025b). All results are reproduced from official code or cited from original papers.

Implementation details. Our model is based on Stable Diffusion v2 from the Diffusers library (von Platen et al. 2022). We train it using AdamW with a learning rate of 1×10^{-4} and weight decay 1×10^{-2}. The MVAM patch sampling radius R is set to 3. Training runs for 80 epochs on four NVIDIA A6000 GPUs. The memory bank uses features from the 3rd and 4th U-Net decoder blocks. Additional details are provided in the supplementary appendix.

Class	Reconstruction Based Method			Embeding Based Method					
	DRAEM	CKAAD	Realnet	Patchcore	CFlow	DeSTSeg	RDPP	FiCo	VSAD(Ours)
Audiojack	91.4/83.2/92.8	87.8/89.2/95.3	91.3/79.8/91.4	**93.7**/81.4/97.6	86.5/81.1/89.9	88.5/81.9/95.6	86.4/85.4/86.3	92.3/89.5/98.3	90.3/**89.7/98.7**
Bottle Cap	96.3/67.6/88.1	95.1/92.8/97.6	97.3/92.8/98.5	94.1/91.7/94.2	**98.9**/86.8/95.3	96.6/85.3/95.1	96.4/95.0/98.5	98.8/**97.7**/98.6	98.2/97.5/**98.6**
Button Battery	94.6/84.4/93.1	91.1/85.7/96.8	90.6/82.5/96.6	81.3/79.2/96.7	93.3/77.8/92.0	92.0/90.8/97.0	94.4/89.0/95.9	81.9/77.4/91.2	**96.4/93.8/97.6**
End Cap	75.4/64.8/86.3	**94.8/90.0**/96.4	81.4/72.8/92.0	86.7/80.6/96.2	83.2/75.2/89.8	87.2/81.5/94.0	93.7/86.7/96.0	89.0/82.7/95.9	94.7/89.3/**96.8**
Eraser	69.4/70.4/81.3	92.3/91.9/96.3	88.9/87.2/98.2	90.6/90.7/98.3	92.1/90.3/98.9	**95.6/96.3**/98.7	93.6/90.4/96.2	89.4/90.1/97.2	93.8/95.0/**99.1**
Fire Hood	83.9/72.0/83.6	84.7/82.0/97.9	85.6/77.9/98.1	87.2/81.6/94.3	88.2/83.6/97.4	**93.4**/88.1/95.6	89.5/83.8/96.7	89.2/85.6/96.1	90.4/**88.3/98.4**
Mint	78.6/70.2/83.1	79.4/73.3/95.4	73.1/66.6/93.6	75.8/70.8/95.3	74.5/69.6/94.2	80.0/70.6/93.4	84.7/82.2/95.9	65.5/65.9/92.6	**85.7/85.8/97.0**
Mounts	89.3/73.3/84.0	97.0/85.9/98.6	97.2/85.0/**99.0**	**98.9**/85.4/96.4	98.3/85.5/98.1	96.1/83.4/96.9	95.7/**88.9**/97.1	96.4/83.1/96.2	96.9/88.5/98.9
Pcb	90.5/87.6/95.5	93.9/93.1/97.3	86.4/77.0/94.5	94.5/94.2/97.3	77.4/75.8/94.3	91.0/87.7/97.5	91.8/91.9/97.7	89.7/89.5/96.2	**94.6/94.5/98.2**
Phone Battery	97.8/76.6/85.4	93.9/91.2/98.2	86.5/81.4/97.3	93.1/89.6/98.1	91.3/86.3/97.1	93.2/83.6/87.3	96.4/**91.3**/98.3	92.7/91.0/**98.4**	94.3/91.1/97.5
Plastic Nut	89.8/71.2/80.8	92.1/89.1/98.5	87.1/80.3/95.6	95.9/88.7/96.4	86.7/77.1/95.3	94.2/88.1/96.1	97.0/92.6/98.5	94.0/88.9/95.9	**97.8/95.2/98.6**
Plastic Plug	85.3/71.8/76.9	91.5/87.2/98.5	86.0/80.3/94.2	87.7/85.2/96.6	90.6/84.9/94.6	93.8/83.9/95.6	95.3/92.0/98.6	93.7/89.2/96.1	**95.5/92.1/99.3**
Porcelain Doll	94.0/75.7/86.2	96.8/86.8/97.1	90.2/80.1/96.9	90.8/79.9/96.1	95.2/83.4/96.3	94.4/81.6/95.8	**97.2/90.4**/98.2	93.9/83.5/**98.3**	94.6/89.3/96.2
Regulator	85.6/72.1/86.0	83.6/82.4/98.1	74.5/65.2/95.5	75.7/71.3/96.6	66.4/59.6/90.9	92.2/87.9/97.5	91.1/**90.1/98.2**	**93.8**/88.6/97.9	91.5/89.5/98.1
Strip Base*	80.0/87.8/94.9	99.5/97.9/99.0	99.0/96.8/98.1	**99.8**/99.4/98.8	99.0/97.5/98.2	98.9/98.4/98.8	99.4/99.3/**99.6**	99.7/99.4/99.6	99.7/98.9/98.6
Sim Card Set	99.7/94.1/96.0	98.7/**96.8**/98.0	91.2/91.4/97.3	93.3/94.3/97.9	96.1/95.1/98.5	97.4/91.7/96.9	96.9/**95.9**/97.6	96.2/95.6/**98.7**	99.0/94.2/97.8
Switch	92.6/84.7/89.7	**98.6**/95.1/97.1	87.7/82.7/96.1	94.3/93.1/96.9	96.0/96.0/97.9	96.3/95.0/99.1	97.3/96.7/99.2	96.2/94.8/96.5	97.1/**96.9/99.3**
Tape	99.1/91.5/97.4	93.3/93.0/98.7	96.8/91.9/99.5	99.2/97.0/98.7	99.5/96.6/99.2	98.6/94.6/99.2	**99.8/97.9/99.6**	99.4/96.8/99.3	97.4/96.8/99.6
Terminalblock	68.9/55.9/83.1	98.4/96.3/99.4	92.3/82.9/91.3	96.6/90.5/99.6	95.7/88.8/97.5	96.8/93.1/97.5	93.7/96.9/98.4	97.3/93.5/99.2	**98.5/96.9/99.7**
Toothbrush	93.0/74.6/68.0	**95.7/88.7/97.2**	86.8/69.7/92.0	91.6/85.3/96.8	92.6/80.0/92.8	92.0/88.3/95.4	95.6/85.3/96.9	90.2/83.2/95.5	95.3/86.7/97.0
Toy	68.0/62.0/60.0	91.8/88.8/95.3	70.8/64.0/90.2	91.5/82.9/95.9	70.1/63.2/86.4	91.6/82.1/89.7	91.5/86.6/95.7	84.6/78.5/90.4	**92.6/88.9/96.1**
Toy Brick	67.6/65.7/91.2	79.5/77.1/92.3	82.4/78.1/94.2	79.9/69.8/92.4	82.1/**81.2**/96.4	84.8/79.0/95.3	82.6/78.4/95.8	88.5/78.4/96.1	**89.6**/80.4/**97.5**
Transistor1	93.8/83.1/88.0	98.7/93.8/98.1	82.4/78.1/92.3	**99.0**/94.8/98.3	98.1/92.6/96.9	97.2/95.0/96.7	97.8/**96.9**/97.9	97.6/96.3/98.4	97.4/95.2/99.1
U Block	89.7/73.9/88.6	98.8/92.3/97.5	90.8/86.4/96.5	96.8/90.1/96.6	94.8/87.0/95.6	98.1/89.3/98.8	**99.1/92.4/99.4**	95.7/89.5/97.4	98.4/89.5/99.4
Usb	82.2/72.6/95.7	**95.0/92.7**/96.9	90.0/83.3/97.7	88.7/82.4/96.4	84.3/80.5/96.0	92.5/87.4/97.8	94.1/90.3/96.4	94.5/90.7/97.2	93.3/91.3/**99.0**
Usb Adaptor	94.6/72.6/85.1	92.2/84.8/97.2	84.6/72.0/95.5	87.0/79.9/96.8	86.7/80.0/94.1	93.7/73.9/91.5	92.1/82.2/96.6	86.0/78.6/92.0	93.6/**85.7/98.1**
Vcpill	82.8/75.5/75.9	96.8/90.7/96.7	92.4/88.9/98.2	88.7/83.5/97.3	91.2/89.5/97.7	96.6/90.6/98.1	96.6/91.1/97.1	89.8/85.2/95.7	**97.3/92.8/98.8**
Wooden Beads	86.8/77.8/86.1	90.7/87.7/98.1	89.2/82.6/97.9	89.8/86.0/95.8	89.4/86.1/96.6	92.6/86.2/95.8	**93.0/89.3/98.1**	90.1/85.7/94.1	89.9/87.2/97.2
Woodstic	76.4/72.5/90.5	74.3/74.5/92.6	**92.4/90.6**/95.3	87.6/86.3/89.0	71.0/79.5/92.8	87.9/87.9/97.1	84.8/85.8/96.4	83.8/85.0/96.7	90.6/89.1/**97.7**
Zipper	97.8/90.8/82.5	99.8/98.8/98.8	99.4/94.0/97.2	98.8/97.9/98.1	98.1/95.7/96.6	99.4/98.5/96.2	84.8/85.8/98.3	99.8/97.5/98.0	99.9/98.9/99.1
Average	86.5/75.9/85.9	92.5/89.0/97.1	88.1/81.4/95.7	90.9/86.1/96.5	88.9/83.5/95.3	93.4/87.4/96.0	93.4/90.0/97.2	91.6/87.7/96.4	**94.8/91.7/98.3**

Table 1: Anomaly detection performance on the **RealIAD** dataset. Scores are reported as S-AUROC / V-AUROC / P-AUROC (%). The best result is in **bold**, second best is underlined. 'Scrip Base*' denotes 'Rolled Scrip Base'.

4.2 Comparison with State-of-the-Art Methods

Quantitative results on RealIAD. The experimental results on the RealIAD dataset are presented in Table 1. Our proposed method **VSAD** achieves the highest average scores across all three metrics: **98.3%** for P-AUROC, **91.7%** for V-AUROC, and **94.8%** for S-AUROC, surpassing the state-of-the-art baselines by **+1.1%**, **+1.8%**, and **+1.4%**, respectively. VSAD performs particularly well in categories with large view changes (e.g., Audiojack, Zipper, PCB), indicating its ability to tell real anomalies apart from differences caused by viewpoint variations. This demonstrates that our view-alignment strategy, which helps the model learn stable and consistent features across views, effectively enhances both fine-grained localization and overall anomaly detection accuracy.

Quantitative results on MANTA. Table 2 presents the quantitative results on the MANTA dataset, where VSAD achieves the best average performance across the dataset, with **96.8%** in P-AUROC, **93.98%** in V-AUROC, and **94.52%** in S-AUROC, surpassing the state-of-the-art baseline methods by **+1.3%**, **+1.1%**, and **+1.1%**. These results demonstrate the robustness and generalization ability of our model across different object categories. In challenging cases such as the "rotten core" defect in the maize category where normal and abnormal textures are very similar, VSAD ranks second among all methods. However, in categories with large viewpoint variations, such as shot button and thin register, VSAD achieves the best performance. Results from both datasets suggest that our model is more stable under viewpoint changes and gives more reliable detection results.

Qualitative results. To further validate the effectiveness of our model, we conduct qualitative experiments. As shown in Fig. 3, our method achieves more accurate localization of anomalous regions. In comparison, the embedding based method PatchCore produces more false positives, likely due to its sensitivity to large viewpoint changes. The reconstruction based method CKAAD performs poorly on subtle defects, especially in texture-rich objects like audiojack from RealIAD dataset, where it shows low true positive activation. Overall, our model better generalizes to viewpoint variations and complex textures, enabling finer and more precise anomaly localization in multi-view settings. Additional visualizations are included in appendix.

4.3 Ablation Studies

Effectiveness of different components. As shown in Table 3, we conduct ablation experiments on the RealIAD and MANTA datasets to evaluate the effectiveness of different components, with model performance assessed at the sample, view and pixel levels. We independently remove the MVAM and FRM modules from the architecture and observe consistent drops in all metrics. Specifically, removing the MVAM module causes a drop by **-8.32%/-8.47%/-7.09%** reaching **86.52%/83.24%/91.25%** on RealIAD dataset and on MANTA dataset by **-10.30%/-**

Class	Reconstruction Based Method			Embedding Based Method					
	DRAEM	CKAAD	Realnet	Patchcore	CFlow	DeSTSeg	RDPP	FiCo	VSAD(Ours)
Block Inductor	85.6/81.8/71.8	89.5/91.2/97.2	76.1/77.4/88.6	92.1/92.5/97.8	84.3/87.2/95.7	92.1/91.4/97.7	91.0/91.4/96.3	77.7/69.2/87.5	**93.2/93.7/98.6**
Copper Standoff	66.9/61.7/71.7	98.2/97.9/98.1	92.9/87.1/95.3	97.1/97.5/97.9	98.0/94.5/96.8	91.8/92.8/94.7	**98.5/98.0/98.4**	66.2/83.7/97.5	96.3/97.9/98.1
Flat Nut	60.1/62.5/69.1	89.8/92.6/96.2	83.8/81.4/94.5	95.5/95.6/98.2	84.8/88.7/97.3	84.7/83.2/96.2	95.0/96.4/98.8	83.0/86.2/95.9	**96.4/97.7/99.6**
Led	92.0/90.1/86.9	97.9/97.2/98.5	90.8/92.6/97.3	96.9/97.8/98.6	97.9/97.8/99.1	96.0/92.0/97.5	98.9/99.0/**99.4**	98.8/98.2/98.6	**99.0/99.8**/99.1
Led Pad	60.7/58.8/71.7	**98.9/97.2/98.8**	77.1/78.3/91.8	98.7/97.1/97.7	91.4/94.0/97.1	98.3/97.0/95.2	96.4/97.1/96.9	98.8/96.7/98.5	95.0/95.4/97.2
Long Button	83.8/75.8/79.4	98.2/97.8/98.5	85.5/83.9/93.1	96.0/96.5/98.0	92.1/92.5/95.8	95.4/97.3/98.5	96.3/96.9/98.1	95.5/96.1/97.8	**98.3/97.9/98.6**
Power Inductor	65.1/66.3/76.9	85.1/86.9/95.6	67.1/62.9/87.4	81.5/82.3/94.5	81.1/80.7/93.7	**87.9/88.7**/94.7	83.8/86.7/**97.0**	80.9/83.7/96.6	86.7/88.3/95.6
Short Button	75.5/67.0/76.1	95.2/95.1/98.1	75.2/74.2/92.8	95.4/94.8/98.9	87.9/87.4/96.9	96.2/95.5/99.2	94.7/95.3/97.0	93.1/92.8/98.6	**96.9/96.8/99.4**
Thin Resistor	90.1/87.8/76.2	97.0/97.5/98.1	89.2/88.6/95.5	97.2/96.0/97.6	95.0/95.6/98.0	98.3/94.8/96.9	98.1/97.0/98.0	97.8/90.3/94.3	**98.8/97.9/99.4**
Type C	89.3/81.9/82.7	96.8/97.0/98.1	84.0/83.7/93.3	97.3/95.6/96.8	92.6/92.8/97.7	97.1/97.2/98.3	98.0/**97.8/98.9**	**98.5**/97.4/98.9	95.3/96.8/97.6
Wafer Resistor	84.0/83.4/71.6	95.3/94.7/97.4	80.9/83.5/95.1	94.0/95.6/96.2	92.4/93.0/95.3	96.8/94.6/96.0	96.1/**96.0**/99.1	92.5/93.2/98.7	**97.6**/95.6/**99.2**
Maize	80.8/79.8/67.8	83.1/82.4/87.3	70.7/71.2/83.2	**85.6/84.1/92.5**	68.6/69.7/82.1	84.4/81.4/79.6	75.6/76.6/90.7	61.2/67.0/89.2	85.1/82.9/91.7
Paddy	67.3/66.8/79.7	86.8/84.3/79.6	88.0/80.9/75.4	**89.1/88.5/85.7**	86.3/81.6/73.7	86.6/80.6/63.1	87.3/84.8/78.3	75.1/73.3/77.3	84.3/88.4/83.2
Soybean	78.8/83.0/68.2	91.8/94.4/93.2	84.6/84.6/85.8	88.7/91.1/92.2	81.4/86.2/88.2	90.7/89.7/85.4	87.5/90.9/92.3	65.4/77.8/90.0	**92.7/94.5/93.3**
Wheat	61.0/67.7/82.0	85.2/84.7/93.0	80.1/76.9/91.5	86.1/86.9/94.6	85.6/86.2/94.4	86.0/82.7/89.0	86.4/85.8/93.5	78.7/84.7/94.1	**86.9/87.6/96.4**
Capsule	98.8/98.0/84.6	97.4/96.9/87.5	95.7/91.9/79.8	98.9/98.1/93.3	96.5/96.8/84.7	98.8/98.1/89.3	99.0/98.4/90.2	98.1/98.2/94.1	**99.4/98.9/95.5**
Coated Tablet	98.6/97.6/99.0	99.6/99.3/99.7	98.7/98.3/99.4	99.2/98.9/99.8	98.1/97.4/99.7	**99.7/99.5/99.8**	98.2/98.6/99.7	98.3/98.8/98.9	97.4/97.1/99.7
Embossed Tablet	83.2/79.6/77.8	96.0/95.9/97.7	86.5/87.2/89.9	95.3/96.1/97.6	90.0/88.9/92.3	95.3/94.7/93.5	96.7/**96.6/98.5**	95.6/95.0/98.3	**97.8**/96.1/98.5
Lettered Tablet	77.2/73.1/68.3	94.9/94.3/**97.5**	77.4/76.0/94.4	94.5/94.1/97.2	91.2/91.3/96.4	87.1/84.0/89.7	**95.1/94.6**/97.3	92.3/93.2/96.2	94.3/91.4/97.0
Oblong Tablet	72.3/70.5/76.0	93.2/91.6/97.1	76.1/75.8/89.1	87.5/89.7/94.5	77.9/81.5/92.4	93.5/89.7/93.0	87.3/90.5/92.6	81.6/77.5/95.2	**95.4/93.9/98.1**
Pink Tablet	94.3/91.8/92.7	98.3/97.5/98.3	95.4/94.2/98.2	97.7/97.3/98.2	96.4/95.5/98.8	97.4/96.5/96.9	98.5/97.2/98.2	**99.0/98.5**/98.0	98.8/98.3/**99.4**
Red Tablet	74.3/71.3/73.7	82.6/85.1/85.0	77.5/75.5/69.5	77.1/80.6/82.9	77.5/79.8/71.2	80.0/81.4/66.4	79.1/82.4/84.1	74.9/77.6/83.6	**87.3/89.6/89.7**
White Tablet	88.9/88.9/82.4	98.5/95.7/97.8	95.5/93.9/97.2	97.2/96.0/97.6	96.1/95.4/96.3	**99.1/96.6/99.2**	98.2/96.6/98.7	97.9/95.9/98.2	98.0/96.0/98.4
Yellow Tablet	93.8/92.7/94.5	98.1/97.0/97.1	98.3/96.5/97.6	98.6/98.3/97.7	98.7/96.3/98.5	**99.3**/97.4/98.4	98.9/**98.3**/98.9	98.7/95.5/98.3	98.4/95.0/98.6
Button	92.8/94.5/96.8	91.0/93.8/98.5	92.9/88.5/99.4	88.7/92.5/95.2	76.7/86.8/93.2	**93.9**/96.0/99.4	90.1/93.8/99.5	89.1/82.6/92.3	93.9/**96.8/99.6**
Gear	87.7/81.6/76.3	96.8/96.2/98.4	84.8/78.2/86.3	96.6/95.3/99.4	85.1/80.4/97.1	98.0/96.9/99.0	96.8/96.1/97.4	96.0/94.3/99.2	**98.1/97.0/99.7**
Nut	66.5/80.2/86.8	94.1/92.8/97.8	92.7/89.6/97.6	93.0/87.5/96.4	84.2/88.4/97.9	82.1/80.4/94.4	91.0/**94.0/99.0**	**96.6**/92.2/97.9	94.0/93.2/97.5
Nut Cap	75.0/65.3/55.7	93.8/89.8/97.6	89.4/79.7/91.4	94.3/89.0/97.2	86.2/80.3/95.0	87.3/77.1/93.0	93.0/86.3/96.1	92.2/87.2/96.5	**98.8/92.9/98.4**
Red Washer	96.7/94.7/91.5	96.9/95.0/98.4	94.3/91.9/98.8	97.5/97.2/98.3	93.0/95.0/98.6	96.7/95.8/97.7	96.6/95.9/98.2	**99.1/97.6/99.2**	97.7/96.6/98.6
Round Button*	82.2/82.4/85.2	98.9/98.1/**99.6**	88.1/87.9/96.7	97.3/97.9/98.4	89.9/93.0/92.8	**99.2**/98.5/99.2	97.9/98.6/99.5	97.2/98.0/99.5	95.7/96.8/98.3
Screw	65.7/67.1/81.7	94.1/91.7/95.6	74.5/72.2/80.6	90.8/88.8/94.6	71.1/76.6/91.8	90.9/88.4/96.3	90.6/89.0/96.4	96.6/89.0/**96.7**	**97.7/92.8**/96.4
Square Button*	97.0/91.9/85.4	97.0/96.9/99.3	92.2/90.3/97.9	**98.2/98.4/99.4**	92.3/93.7/98.4	97.2/96.9/98.7	97.6/97.2/99.3	97.7/97.1/99.3	95.0/95.0/98.3
Terminal	82.2/77.3/70.7	96.5/95.9/97.4	77.0/73.7/90.7	93.6/94.0/96.1	84.1/84.5/96.5	96.7/95.0/98.5	94.0/92.8/97.2	**97.1/98.2/98.9**	96.8/97.6/98.6
Wire Cap	93.3/87.9/88.4	94.6/93.9/96.6	82.6/80.8/95.6	90.6/93.0/95.0	83.7/84.8/94.0	**96.6/95.4**/98.3	91.0/91.1/98.5	88.5/92.9/98.3	95.1/94.2/**98.7**
Yellow Washer*	78.5/76.5/77.9	93.4/92.4/95.1	89.6/85.7/90.4	92.1/91.2/95.5	87.5/88.3/92.8	92.8/91.1/94.0	93.3/95.1/95.3	96.9/91.7/95.2	**97.1/96.1/96.5**
Coffee Beans	65.1/60.4/62.3	85.3/80.9/89.6	67.3/61.5/70.3	74.9/75.5/89.5	74.8/72.3/84.5	83.6/80.9/88.4	64.4/62.1/85.5	67.7/64.3/84.4	**88.9/86.8/91.5**
Goji Berries	78.4/77.2/81.9	92.4/87.5/93.7	77.7/75.1/88.6	89.0/85.0/91.2	87.8/83.7/93.5	86.4/79.7/87.7	89.2/84.3/93.1	92.2/81.3/90.4	**94.2/89.5/95.3**
Pistachios	57.2/62.8/67.5	78.4/77.0/86.7	79.4/76.6/76.7	78.1/77.3/83.6	74.7/75.9/77.8	61.2/60.8/69.1	74.6/72.9/81.5	71.7/67.0/78.4	**79.8/78.3/88.1**
Average	80.0/78.4/78.7	93.4/92.8/95.5	84.5/82.3/90.4	92.4/92.2/95.4	87.2/87.7/93.1	92.0/90.3/92.7	92.0/91.9/95.4	88.8/88.2/94.7	**94.5/93.9/96.8**

Table 2: Comprehensive anomaly detection results with S-AUROC / V-AUROC / I-AUROC(%) metrics on MANTA. The best and second-best results are mark in **bold** and underlined. 'Round Button*' denotes 'Round Button Cap', 'Square Button*' denotes 'Square Button Cap' and 'Yellow Washer*' denotes 'Yellow Green Washer'.

10.38%/-7.47% reaching **84.22%/83.56%/89.34%**, highlighting its importance to implicitly align multi view representation. Similarly, removing the FRM modules from each decoder layer results in drops of **-1.62%/-0.89%/-0.99%** on RealIAD dataset, with final scores of **93.22%/90.82%/97.35%**, and **-0.90%/-0.68%/-0.98%** on MANTA dataset, with scores of **93.62%/93.26%/95.83%** indicating that FRM effectively refines the learned features by explicitly enforcing consistency across views and reducing noise from the fusion process for better anomaly detection. Fig. 4 presents the localization results from various component ablation studies, showing that the proposed components significantly enhance anomaly discrimination. More visualizations are provided in the appendix.

Cross-view feature distribution analysis. As shown in Fig. 5 in arxiv version (Chen et al. 2025a), before passing through the MVAM module, features from different views are more scattered in the feature space (processed via t-SNE). After multi-stage alignment via MVAM, they form tighter clusters, which are further compacted by the FRM module. This process reduces boundary noise and improves cross-view consistency.

Dataset	Method	S-AUROC (↑)	V-AUROC(↑)	P-AUROC (↑)
RealIAD	w/o MVAM	86.52	83.24	91.25
	w/o FRM	93.22	90.82	97.35
	VSAD(Full)	**94.84**	**91.71**	**98.34**
MANTA	w/o MVAM	84.22	83.56	89.34
	w/o FRM	93.62	93.26	95.83
	VSAD(Full)	**94.52**	**93.94**	**96.81**

Table 3: Ablation study on the key components of VSAD on the RealIAD and MANTA datasets. Performance is measured in Sample / View / Pixel-AUROC (%).

Impact of different UNet decoder layer. To assess the impact of different UNet decoder levels on our embedding-based model, we conduct ablation studies using various

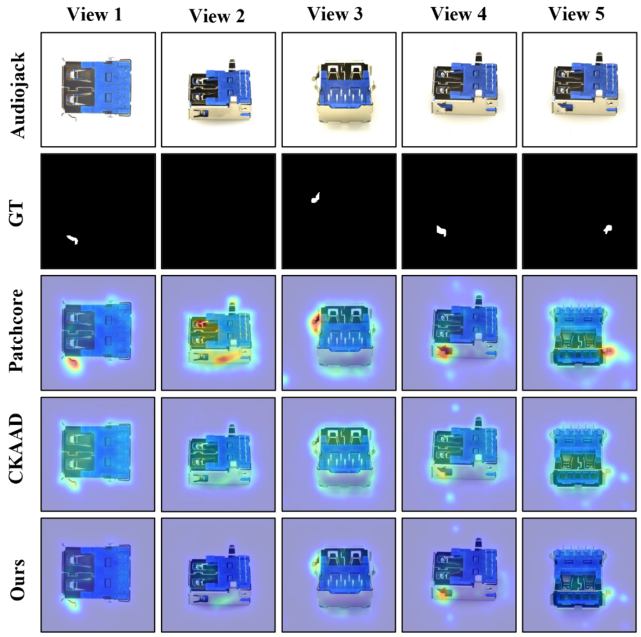

Figure 3: Qualitative anomaly localization results on RealIAD Compared to baselines like PatchCore and CKAAD, our method (VSAD) produces significantly more accurate and fine-grained localization maps with fewer false positives, demonstrating its superior ability to handle viewpoint variations and subtle defects.

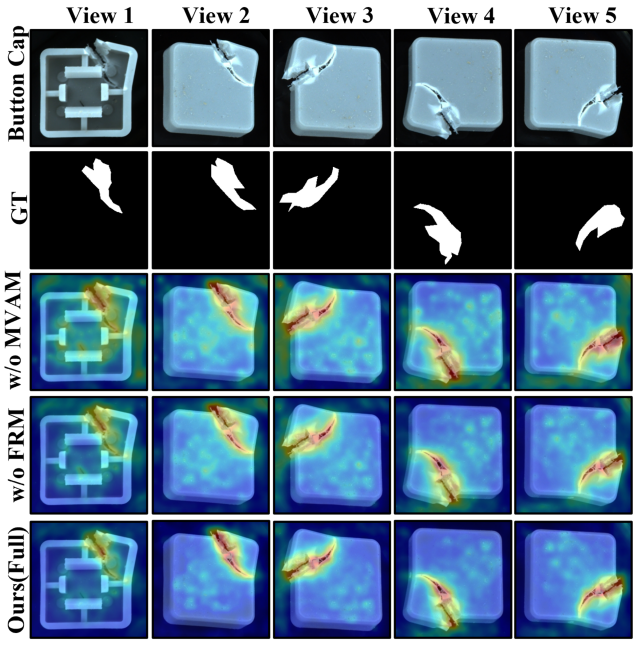

Figure 4: Visualization of the effect of component ablations on localization performance, shown on MANTA.

combinations of decoder layers on both datasets. As shown in Table 4, using features from both the 3rd and 4th decoder

Layers Used				S-AUROC(↑)	V-AUROC(↑)	P-AUROC(↑)
4	3	2	1			
✓	✗	✗	✗	93.7	91.1	97.8
✓	✓	✗	✗	**94.8**	**91.7**	**98.3**
✓	✓	✓	✗	92.3	90.9	97.4
✓	✓	✓	✓	89.4	86.9	95.5

Table 4: Impact of using features from different U-Net decoder layers on RealIAD performance (%). Using levels 4+3 provides the best overall performance.

$R \times R$	S-AUROC(↑)	V-AUROC(↑)	P-AUROC(↑)
1×1	91.52	91.03	95.21
2×2	94.32	93.67	96.15
3×3	**94.52**	**93.94**	**96.81**
4×4	93.95	93.54	95.95

Table 5: Ablation study performance (%) of patch sampling radius hyperparameter R on the MANTA benchmark.

blocks yields the best performance on RealIAD, with improvements of **+1.16%**(S-AUROC), **+0.66%**(V-AUROC), and **+0.58%**(P-AUROC) over using only the 4th block. High-level features provide rich semantic information, while mid-level features add structural details that help improve performance. In contrast, adding lower-level features from the 2nd and 1st decoder blocks leads to performance drops, likely due to the increased noise from multi-view variations.

Impact of hyperparamenters patch sampling radius. According to Table 5, an appropriate selection on hyperparamenters R greatly improves anomaly localization and detection for our method. When $R \times R = 9$, the best performance of the model is achieved. We consider that a small patch sampling radius may lead to insufficient alignment between multi view representations, while increasing the radius enhances robustness and improves detection accuracy. However, when $R = 4$, the model performance slightly decreases due to noise from excessive sampling.

5 Conclusion

We introduced ViewSense-AD, a framework tackles a key challenge in multi-view anomaly detection: distinguishing true defects from geometric variations. By embedding a homography-guided alignment module into a latent diffusion model, VSAD progressively learns viewpoint-invariant representations of object surfaces. Enhanced by a feature refinement module, this process achieves new state-of-the-art results on the RealIAD and MANTA benchmarks. Our work demonstrates that explicitly modeling cross-view geometric consistency is a robust and effective path forward for real-world industrial inspection.

Limitation and future work. Future work could replace rigid homographies with learnable deformation fields for non-rigid objects (e.g.,textiles) and learn alignment end-to-end, eliminating pre-calibrated cameras and improving adaptability in dynamic settings.

References

Akshay, S.; Narasimhan, N. L.; George, J.; and Balasubramanian, V. N. 2025. A Unified Latent Schrodinger Bridge Diffusion Model for Unsupervised Anomaly Detection and Localization. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 25528–25538.

Bae, J.; Lee, J.-H.; and Kim, S. 2023. Pni: industrial anomaly detection using position and neighborhood information. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6373–6383.

Banerjee, P.; Shkodrani, S.; Moulon, P.; Hampali, S.; Han, S.; Zhang, F.; Zhang, L.; Fountain, J.; Miller, E.; Basol, S.; et al. 2025. Hot3d: Hand and object tracking in 3d from egocentric multi-view videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 7061–7071.

Bergmann, P.; O'Mahony, M.; Batzner, K.; Fauser, M.; Sattlegger, D.; and Steger, C. 2020. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14435–14444.

Cai, R.; Zhang, H.; Liu, W.; Gao, S.; and Hao, Z. 2021. Appearance-Motion Memory Consistency Network for Video Anomaly Detection. In *AAAI Conference on Artificial Intelligence*.

Cao, Y.; Xu, X.; Zhang, J.; Cheng, Y.; Huang, X.; Pang, G.; and Shen, W. 2024. A Survey on Visual Anomaly Detection: Challenge, Approach, and Prospect. *arXiv preprint arXiv:2401.16402*.

Chang, J.; He, J.; Zhang, T.; Yu, J.; and Wu, F. 2024. EI-MVSNet: Epipolar-Guided Multi-View Stereo Network With Interval-Aware Label. *IEEE Transactions on Image Processing*, 33: 753–766.

Chen, X.; Xu, X.; Zheng, B.; Liu, Y.; and Wu, Y. 2025a. Unsupervised Multi-View Visual Anomaly Detection via Progressive Homography-Guided Alignment. arXiv:2511.18766.

Chen, Z.; Luo, X.; Wang, W.; Zhao, Z.; Su, F.; and Men, A. 2025b. Filter or compensate: Towards invariant representation from distribution shift for anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 2420–2428.

Daryani, A. E.; Bhutta, M.; Hernandez, B.; and Medeiros, H. 2025. CaMuViD: Calibration-Free Multi-View Detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 1220–1229.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Fan, L.; Fan, D.; Hu, Z.; Ding, Y.; Di, D.; Yi, K.; Pagnucco, M.; and Song, Y. 2025a. Manta: A large-scale multi-view and visual-text anomaly detection dataset for tiny objects. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 25518–25527.

Fan, L.; Huang, J.; Di, D.; Su, A.; Song, T.; Pagnucco, M.; and Song, Y. 2025b. Salvaging the Overlooked: Leveraging Class-Aware Contrastive Learning for Multi-Class Anomaly Detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 21419–21428.

Fang, Q.; Su, Q.; Lv, W.; Xu, W.; and Yu, J. 2025. Boosting Fine-Grained Visual Anomaly Detection with Coarse-Knowledge-Aware Adversarial Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 16532–16540.

Gao, R.; Holynski, A.; Henzler, P.; Brussee, A.; Martin-Brualla, R.; Srinivasan, P.; Barron, J. T.; and Poole, B. 2024. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv preprint arXiv:2405.10314*.

Gong, D.; Liu, L.; Le, V.; Budhaditya, S.; Rowe, D.; and Reid, I. 2019. Memorizing normality to detect anomaly in video: A memory-augmented autoencoder with dynamic online update. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1705–1714.

Gudovskiy, D.; Ishizaka, T.; and Kozuka, K. 2022. CFLOW-AD: Real-Time Unsupervised Anomaly Detection with Conditioned Normalizing Flows. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 1116–1126.

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 *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 8472–8480.

He, X.; Zhang, Y.; Zhang, Z.; and Fang, H. 2020. EPINET: A feature pyramid network for cross-view image matching. In *2020 International Conference on Culture-oriented Science & Technology (ICCST)*, 313–317. IEEE.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, 6840–6851.

Hu, T.; Zhang, J.; Yi, R.; Du, Y.; Chen, X.; Liu, L.; Wang, Y.; and Wang, C. 2024. Anomalydiffusion: Few-shot anomaly image generation with diffusion model. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 8526–8534.

Hwang, J.; Benz, P.; and Kim, P. 2024. Booster-shot: Boosting stacked homography transformations for multi-view pedestrian detection with attention. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 363–372.

Kim, S.; Jin, C.; Diethe, T.; Figini, M.; Tregidgo, H. F.; Mullokandov, A.; Teare, P.; and Alexander, D. C. 2024. Tackling structural hallucination in image translation with local diffusion. In *European Conference on Computer Vision*, 87–103. Springer.

Li, W.; Gu, Y.; Chen, X.; Xu, X.; Hu, M.; Huang, X.; and Wu, Y. 2025. Towards visual discrimination and reasoning of real-world physical dynamics: Physics-grounded anomaly detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 30409–30419.

Liu, C.; Chu, Y.-M.; Hsieh, T.-I.; Chen, H.-T.; and Liu, T.-L. 2025. Learning Diffusion Models for Multi-view Anomaly Detection. In Leonardis, A.; Ricci, E.; Roth, S.; Russakovsky, O.; Sattler, T.; and Varol, G., eds., *Computer Vision – ECCV 2024*, 328–345. Springer Nature Switzerland.

Liu, X.; Wang, J.; Leng, B.; and Zhang, S. 2024. Dual-modeling decouple distillation for unsupervised anomaly detection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 5035–5044.

Ni, J.; Zhao, W.; Wang, D.; Zeng, Z.; You, C.; Wong, A.; and Huang, K. 2025. HOMER: Homography-Based Efficient Multi-view 3D Object Removal. *arXiv preprint arXiv:2501.17636*.

Roth, K.; Pemula, L.; Zepeda, J.; Schölkopf, B.; Brox, T.; and Gehler, P. 2022. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14318–14328.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. In *International Conference on Learning Representations*.

Song, J.; Park, D.; Baek, K.; Lee, S.; Choi, J.; Kim, E.; and Yoon, S. 2025. DefectFill: Realistic Defect Generation with Inpainting Diffusion Model for Visual Inspection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 18718–18727.

Sun, H.; Cao, Y.; Dong, H.; and Fink, O. 2025. Unseen Visual Anomaly Generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 25508–25517.

Sun, J.; Shen, Z.; Wang, Y.; Bao, H.; and Zhou, X. 2021. LoFTR: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8922–8931.

Tien, T. D.; Nguyen, A. T.; Tran, N. H.; Huy, T. D.; Duong, S.; Nguyen, C. D. T.; and Truong, S. Q. 2023. Revisiting reverse distillation for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 24511–24520.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, volume 30.

von Platen, P.; Patil, S.; Lozhkov, A.; Cuenca, P.; Lambert, N.; Rasul, K.; Davaadorj, M.; Nair, D.; Paul, S.; Berman, W.; Xu, Y.; Liu, S.; and Wolf, T. 2022. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers.

Wang, C.; Zhu, W.; Gao, B.-B.; Gan, Z.; Zhang, J.; Gu, Z.; Qian, S.; Chen, M.; and Ma, L. 2024. Real-iad: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 22883–22892.

Wang, J.; Cheng, J.; Gao, C.; Zhou, J.; and Shen, L. 2025a. Enhanced Fabric Defect Detection With Feature Contrast Interference Suppression. *IEEE Transactions on Instrumentation and Measurement*, 74: 1–12.

Wang, S.; Ding, X.; Mao, Y.; and Dai, Y. 2025b. ETV-MVS: Robust Visibility-Aware Multi-View Stereo with Epipolar Line-Based Transformer. *Big Data Mining and Analytics*, 8(3): 520–533.

Wu, C.-Y.; Johnson, J.; Malik, J.; Feichtenhofer, C.; and Gkioxari, G. 2023. Multiview compressive coding for 3D reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9065–9075.

Wyatt, J.; Leach, A.; Schmon, S. M.; and Zisserman, A. 2022. AnoDDPM: Anomaly Detection With Denoising Diffusion Probabilistic Models. In *International Conference on Machine Learning*. PMLR.

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 Leonardis, A.; Ricci, E.; Roth, S.; Russakovsky, O.; Sattler, T.; and Varol, G., eds., *Computer Vision – ECCV 2024*, 1–17. Cham: Springer Nature Switzerland.

Yao, X.; Li, R.; Qian, Z.; Wang, L.; and Zhang, C. 2024. Hierarchical gaussian mixture normalizing flow modeling for unified anomaly detection. In *European Conference on Computer Vision*, 92–108. Springer.

Zavrtanik, V.; Kristan, M.; and Skočaj, D. 2021. DRAEM–A discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 833–842.

Zavrtanik, V.; Kristan, M.; and Skočaj, D. 2023. D-RAK: A Denoising-based Reconstruction for Unsupervised Anomaly Kingdom Detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 6315–6324.

Zhang, W.; Yang, Y.; Huang, H.; Han, L.; Shi, K.; Liu, Y.-S.; and Han, Z. 2025. MonoInstance: Enhancing monocular priors via multi-view instance alignment for neural rendering and reconstruction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 21642–21653.

Zhang, X.; Li, S.; Li, X.; Huang, P.; Shan, J.; and Chen, T. 2023a. Destseg: Segmentation guided denoising student-teacher for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3914–3923.

Zhang, X.; Xu, M.; and Zhou, X. 2024. Realnet: A feature selection network with realistic synthetic anomaly for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16699–16708.

Zhang, Z.; Song, X.; Shen, Z.; and You, Y. 2023b. DiffusionAD: A Generative Approach for Anomaly Detection. *arXiv preprint arXiv:2311.16373*.

Zheng, B.; Gan, J.; Xu, X.; Chen, X.; Li, W.; Huang, X.; Ni, N.; and Wu, Y. 2025. Bridging 3D Anomaly Localization and Repair via High-Quality Continuous Geometric Representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 27063–27072.