VLMDiff: Leveraging Vision-Language Models for Multi-Class Anomaly Detection with Diffusion

Samet Hicsonmez, Abd El Rahman Shabayek, Djamila Aouada
University of Luxembourg
Luxembourg, Luxembourg
{samet.hicsonmez,abdelrahman.shabayek,djamila.aouada}@uni.lu

Abstract

Detecting visual anomalies in diverse, multi-class real-world images is a significant challenge. We introduce VLMDiff, a novel unsupervised multi-class visual anomaly detection framework. It integrates a Latent Diffusion Model (LDM) with a Vision-Language Model (VLM) for enhanced anomaly localization and detection. Specifically, a pre-trained VLM with a simple prompt extracts detailed image descriptions, serving as additional conditioning for LDM training. Current diffusion-based methods rely on synthetic noise generation, limiting their generalization and requiring per-class model training, which hinders scalability. VLMDiff, however, leverages VLMs to obtain normal captions without manual annotations or additional training. These descriptions condition the diffusion model, learning a robust normal image feature representation for multi-class anomaly detection. Our method achieves competitive performance, improving the pixel-level Per-Region-Overlap (PRO) metric by up to 25 points on the Real-IAD dataset and 8 points on the COCO-AD dataset, outperforming state-of-the-art diffusion-based approaches. Code is available at https://github.com/giddyyupp/VLMDiff.

1. Introduction

Visual Anomaly Detection (AD) involves identifying and localizing abnormal regions in images, with applications across various computer vision tasks. These include quality control in manufacturing [2, 46, 67] where detecting defects ensures product reliability, and autonomous driving [5] where identifying unseen obstacles is crucial for safety. The key challenge in visual AD lies in the scarcity of labeled abnormal samples—while normal cases are often well-documented, collecting and annotating anomalous instances is typically impractical due to their rarity and diversity [35]. Consequently, unsupervised approaches that rely solely on normal examples are essential for real-world deployment, enabling robust AD without the need for exhaustive manual annotations [11].

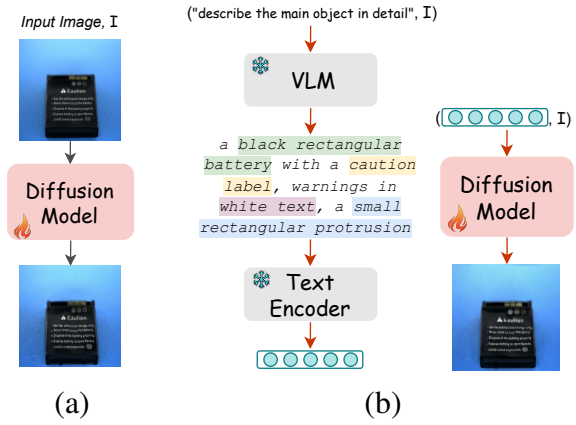

Figure 1. Comparison of (a) current diffusion-based approaches and (b) our approach. Our method extracts textual descriptions from VLMs to guide the training of the underlying diffusion model.

The primary approaches for visual AD can be categorized into augmentation-based [29, 43, 55], embedding-based [18, 24, 25], reconstruction-based [19, 45, 60], and hybrid approaches [53, 63] that combine multiple techniques. Among these, reconstruction-based methods emerge as a natural solution which train models to learn normal sample reconstructions. During inference, when an anomalous sample is encountered, the model will attempt to reconstruct it as a normal sample, resulting in a higher reconstruction error compared to normal inputs. This reconstruction error is then used to identify anomalies. Various architectures have been explored for this purpose, including autoencoders [1], masked vision transformers (ViTs) [60], and Mamba models [19]. Recently, diffusion-based [22, 39, 44] approaches have been proposed for AD [20, 30, 48, 49, 57, 62] as another line of reconstruction-based methods. However, first, their perfor-

mance still lags behind the current state-of-the-art methods, especially on datasets with subtle defect types [46], primarily due to weak guidance signals during training. Second, most of these models [15, 48, 49, 57, 62], except for [20], are trained on a per-class basis, making them impractical for large-scale datasets with multiple object categories. Last but not least, their pipeline relies on synthetic anomaly generation in the image space, which limits them to specific defect types and domains.

Another line of research [6, 17, 54, 65] explores leveraging the strong zero-shot capabilities of vision-language models (VLMs) for visual AD. These methods rely on learning effective text prompts to generate accurate descriptions of both normal and anomalous cases. However, to learn prompts for anomalies, they typically require labeled auxiliary anomalous data from external datasets, which is often difficult to obtain in practice. Additionally, since object defects occur at the pixel level, prompt learners often struggle to accurately determine whether an anomaly is present, limiting their effectiveness in fine-grained AD.

In this work, we propose a multi-class AD framework that puts together the strong descriptive power of VLMs and diffusion models, dubbed as VLMDiff. It leverages textual descriptions extracted from off-the-shelf VLMs without requiring prompt learning. We devise one simple prompt to get the detailed description of the main object for the training. These descriptions are embedded using a text encoder. The encoded vector is then incorporated as an additional conditioning signal for the diffusion model to improve the reconstruction performance. Our approach exploits Latent Diffusion Models (LDMs) [39] for an efficient reconstruction process. Figure 1 presents a high-level comparison between existing diffusion-based methods and VLMDiff.

VLMDiff is not limited to a specific domain, it shows superior performance on industrial (Real-IAD [46]) and more general purpose (COCO-AD [59]) large scale AD datasets. It significantly outperforms the state-of-the-art diffusion-based methods on pixel-level metrics. Notably, our method achieves up to a 25-point improvement on the Real-IAD [46] dataset in the pixel-level PRO metric. Unlike the existing diffusion-based methods [15, 57, 62], VLMDiff supports multi-class AD where only a single model per dataset is trained rather than training a single model per class which would improve 1) **generalization** thanks to broader normality representation, 2) **robustness** thanks to reduced overfitting by avoiding per class training, 3) **scalability** thanks to reduced model complexity by maintaining and deploying a single model only, 4) **efficiency** thanks to faster training and inference, 5) **AD in complex scenarios** thanks to capturing contextual relationships between different object categories within a dataset, and 6) **detection of novel anomalies** thanks to the holistic representation of normality that is not constrained to any specific class-based definition. The proposed VLMDiff has the following key contributions:

- We leverage the strong descriptive capabilities of vision-language models (VLMs) to guide the learning of the diffusion-based multi-class AD model.
- Without requiring specially designed or learned prompts, VLMDiff substantially improves the baseline pixel-level performance.
- Learning a multi-class AD model rather than an individual class per category. This has a tangible impact on generalization, scalability and efficiency.
- We demonstrate the effectiveness of our approach on two diverse datasets, highlighting its robustness and generalizability.

2. Related Work

In this section, we first give a summary of prominent visual AD methods, then focus on the diffusion-based approaches, and finally elaborate on the prompt learning based methods which use textual descriptions as an additional modality.

Visual AD. Existing methods can be broadly categorized into three strategies: embedding-based, augmentation-based, and reconstruction-based.

Embedding based approaches [18, 24, 25, 33, 41] first extract high-dimensional image or patch features from pretrained networks, and then compare features coming from normal and anomalous data to locate anomalies. Very first approaches [33] use clustering on the extracted patch features from ResNet [21] pretrained on ImageNet [14]. Later works [10, 12, 24, 41] add memory banks to extend the available feature context. Normalizing flow based methods [18, 25] extract multi-level patch features and learn either the distribution of positive patches or minimize the distance between each positive patch. Knowledge distillation is also used within a teacher-student framework in [4, 47].

Another line of work [29, 43, 55, 64] employs augmentations to simulate anomalous data. DRAEM [55] trains an autoencoder to reconstruct both normal and generated noisy images using Perlin [37], and a discriminative network locates the anomalies.

Generative methods learn to reconstruct normal samples using VAE [2], GAN [1, 42], ViT [60], or more recently, diffusion models [20, 48, 49, 57]. During inference, a higher reconstruction error will result from anomalous inputs compared to the normal ones. Bergmann et al. [2] show a VAE trained with a structural similarity (SSIM) objective that is able to detect pixel-level defects. GANomaly [1] employs a GAN architecture to detect image-level anomalies. ViTAD [60] uses vision transformers in VAE architecture and obtains state-of-the-art performance.

Diffusion Models. These models [22, 39] show unprecedented image generation capabilities, which makes them a natural candidate for AD. The first works using diffu-

sion are proposed in the medical domain [48, 49]. Later works [15, 20, 30, 52, 57, 62] extend diffusion models to industrial defect localization. DiffAD [62] consists of two sub-networks: a Latent Diffusion Model (LDM) to reconstruct samples and a discriminative network to segment defect locations. DiffusionAD [57] follows a similar approach as DiffAD with the addition of a synthetic anomaly generation pipeline. During training, synthetic anomaly data is generated either using an external dataset [9] for overlapping classes or Perlin noise [37] similar to DRAEM [55]. In order to reduce the inference time, a norm-guided one-step denoising approach is employed. The main drawbacks of these approaches are that, first, they train separate models for each class, which limits them from generalizing to large datasets with many objects, and second, the generation and segmentation models are separately trained. Transfusion [15] merges the generation and anomaly localization parts into a single diffusion model which works in image space [22]. The diffusion model takes input as Perlin noise added to normal images along with noise masks and outputs both the noise-removed images along with the noise masks.

Recently, DiAD [20] proposed using a conditional diffusion model which relaxed the need to train separate models for each class. Moreover, the anomaly segmentation is done by comparing the multi-level features extracted from a fixed pretrained network for input and reconstructed images, instead of training a separate segmentation network. These two properties make DiAD versatile to adapt to different datasets which could have multiple objects in the images as anomalies. However, the performance of DiAD and previous diffusion-based methods is falling behind the current state-of-the-art on large and complex datasets [46, 59], most likely due to the weak guidance they are using. Our method strengthens the guidance by using detailed text descriptions extracted from powerful VLMs.

Multi Modal Methods. Alternatively, so far, text data is used in the context of prompt learning for extracting short anomaly descriptions. These methods utilize CLIP [38] which shows exceptional zero-shot detection capability in various unsupervised vision tasks. CLIP-AD [31] is one of the very first approaches using plain CLIP image and text features for image-level AD. Later methods [6, 17, 23, 65] focus on either crafting or learning useful text prompts that capture the possible states of anomaly classes. However, in order to learn useful prompts, these methods utilize external anomaly data. Different from these, LAVAD [54] proposes a training-free video AD framework supported by VLMs. They generate detailed captions using Blip-2 [26] for each frame in a video sequence and measure the alignment between CLIP image and text encoders. We also follow a prompt learning-free methodology in our work. A recent study by Xu et al. [50] introduces a novel human-annotated dataset for anomaly detection enriched with detailed anomaly descriptions, and shows that fine-tuning VLMs on this dataset significantly improves both detection performance and reasoning ability. However, this method lacks anomaly localization. For a more comprehensive review of VLMs in AD, we refer to the following surveys [32, 51].

3. VLMDiff

We present the detailed structure of our method in Figure 2. The training pipeline starts by feeding the input image I to the pretrained image encoder $\mathcal{E}_I$ to extract the latent representation z. In parallel, we forward the image to an off-the-shelf large VLM along with a generic prompt $\mathcal{P}$ to get the image description $\text{VLM}(I, \mathcal{P})$. The image description is further encoded into a condition vector c using a pretrained text encoder $\mathcal{E}_T$. The diffusion process starts by adding noise T times to the latent vector z and getting the noised latent vector z_t. Then, denoising network recovers input z from noisy latent z_t with the guidance coming from the conditional vector c. Note that, unlike the current state-of-the-art diffusion-based methods, which train separate models per category, we train a single model per dataset.

At inference time, the denoised latent $\tilde{z}$ is decoded back to the image space using the pretrained image decoder $\mathcal{D}_I$ to get the reconstructed image $\tilde{I}$. In order to segment the anomaly locations, the input image I and the reconstructed image $\tilde{I}$ are fed to an off-the-shelf self-supervised model to extract the features F and $\tilde{F}$, respectively. Finally, the pixels with high dissimilarity between these two feature vectors denote the Anomaly Map A.

Vision Language Model (VLM) The mainstream approach to guide the unsupervised diffusion-based AD methods is to use some form of noisy input generation by either using external defect datasets or generating synthetic defects using noise generators. We argue that this form of guidance is both unrealistic for specific anomaly types like scratches and does not generalize to natural image datasets such as COCO-AD [59]. Instead, we propose to guide the diffusion model with detailed image descriptions using large VLMs. Considering the main goal of the training pipeline is to learn to reconstruct the input latent, we extract detailed descriptions of the input images that could help this objective.

For industrial datasets (e.g. Real-IAD [46]), the normal training images are examined by the VLM model by a prompt to describe the main object. The description query is $\mathcal{P}_D$: "*Describe the main object in detail.*". Whereas for the inference part, that has defective objects as well, we observe that prompting the VLM to describe the object results in noisy descriptions. We decide to use no text description for the inference part and let the diffusion model generate unconditionally. Hence the prompt used for training becomes $\mathcal{P}_T = \mathcal{P}_D$, and during inference we do not use any text conditioning.

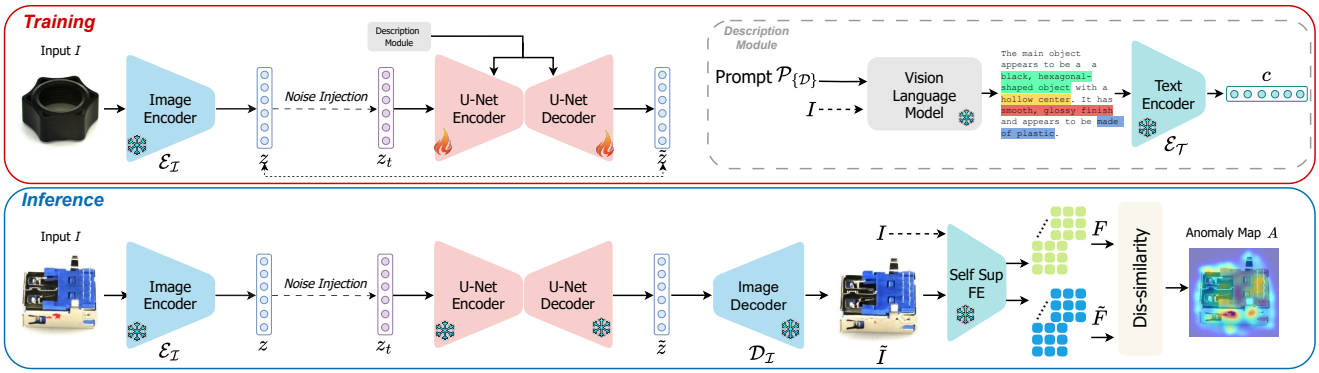

Figure 2. The processing pipeline of VLMDiff. During training (top), a normal (i.e. anomaly-free) image is fed to both 1) an off-the-shelf VLM (on the right) to extract the detailed description of the object using the query ($\mathcal{P}_D$), which is further encoded into condition vector c using text encoder, and to 2) a pretrained image encoder (note that we finetune the image autoencoder using only normal images beforehand) to get the latent vector. Then, a diffusion process adds noise to the latent vector and learns to denoise it with the guidance coming from the condition vector. During inference (bottom), the same process as training is followed, except there is no text description is used to condition the diffusion model. The denoised latent code is fed to the pretrained image decoder to get the reconstructed image. Anomaly segmentation is done by finding the dissimilar locations on the feature maps of input and reconstructed normal images.

For natural image datasets (e.g. COCO-AD [59]), the anomalies are object-level (i.e. unseen classes during training) not pixel-level, and VLMs show exceptional performance to describe the objects, even segment them [56, 66], in an image as they are trained in this manner. Thus, we use the description query with a slight modification for both training and inference (i.e. $\mathcal{P}_T = \mathcal{P}_I = \mathcal{P}_D$). Our description query $\mathcal{P}_D$ on COCO-AD is: "*Describe the visual features of image in detail.*". In this way, we expect the VLM to mention the objects and their relations, which will guide the diffusion model. These descriptions are extracted without using any label information. Finally, the description is encoded using a text encoder (e.g. CLIP [38]) into the text condition vector c. For a detailed analysis on the effect of using queries, refer to Section Ablation Experiments.

Diffusion Model We use the same diffusion architecture presented in [39] which comprises an image autoencoder and a denoising attention-based U-Net [40] model. In order to extract the latent vectors from images, we first train the image autoencoder using only the training set which contains normal images.

For a given input image $I \in \mathbb{R}^{H\times W\times 3}$ in pixel space, we first encode it into latent space $z = \mathcal{E}_\mathcal{I}(I)$ where $z \in \mathbb{R}^{h\times w\times d}$ using the trained image encoder $\mathcal{E}_\mathcal{I}$. The forward process can be characterized by:

$$z_t = \sqrt{\bar{\alpha}_t} z + \sqrt{1-\bar{\alpha}_t}\epsilon_t, \epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \quad (1)$$

where $\alpha_t = 1 - \beta_t$, and the cumulative product $\bar{\alpha}_t$ is given by $\bar{\alpha}_t = \prod_{i=1}^{T} \alpha_i = \prod_{i=1}^{T}(1-\beta_i)$. Here, β_i defines the noise schedule, which controls the amount of noise introduced at each timestep.

The noised representation z_t, and the text condition vector c are fed into both the denoising U-Net encoder and decoder networks. After T steps of the reverse denoising process, the denoised latent vector $\tilde{z}$ is obtained. In the end, the reconstructed image $\tilde{I}$ is generated by forwarding the $\tilde{z}$ to the pretrained image decoder $\mathcal{D}_\mathcal{I}(\tilde{z})$. The training objective of VLMDiff is:

$$\mathcal{L}_{VLMDiff} = \mathbb{E}_{z,t,c,\epsilon\sim\mathcal{N}(0,1)}\left[\|\epsilon - \epsilon_\theta(z_t, t, c)\|_2^2\right]. \quad (2)$$

where the objective function aims to minimize the difference between the noise added to the data at a certain time step ϵ and the noise ϵ_θ predicted by the model. Note that our diffusion model is trained in a multi-class setting by using all the normal images from training sets without utilizing any class labels.

Anomaly Segmentation Model The anomaly segmentation part is only used during inference to localize the defective pixels. For a given test image I, after following the same process as training, the image decoder $\mathcal{D}_\mathcal{I}$ reconstructs the denoised latent vector $\tilde{z}$ to image space $\tilde{I}$. The difference between the input and reconstructed image will denote the defective pixels.

The input image I and the reconstructed image $\tilde{I}$ are fed to a pretrained model to extract the features F and $\tilde{F}$, respectively. Then we resize these patch features to the input image size and calculate the pixel-wise cosine similarity. Finally, we calculate $1 - cossim(\tilde{F}, F)$ to localize the anomaly regions. In order to maintain the unsupervised nature of our model, we utilize a self-supervised method DINO [7] for the anomaly segmentation, and use the patch features from the last layer.

4. Experiments

4.1. Datasets

We experimented with two very large datasets, industrial defect detection Real-IAD [46], and real-world anomaly

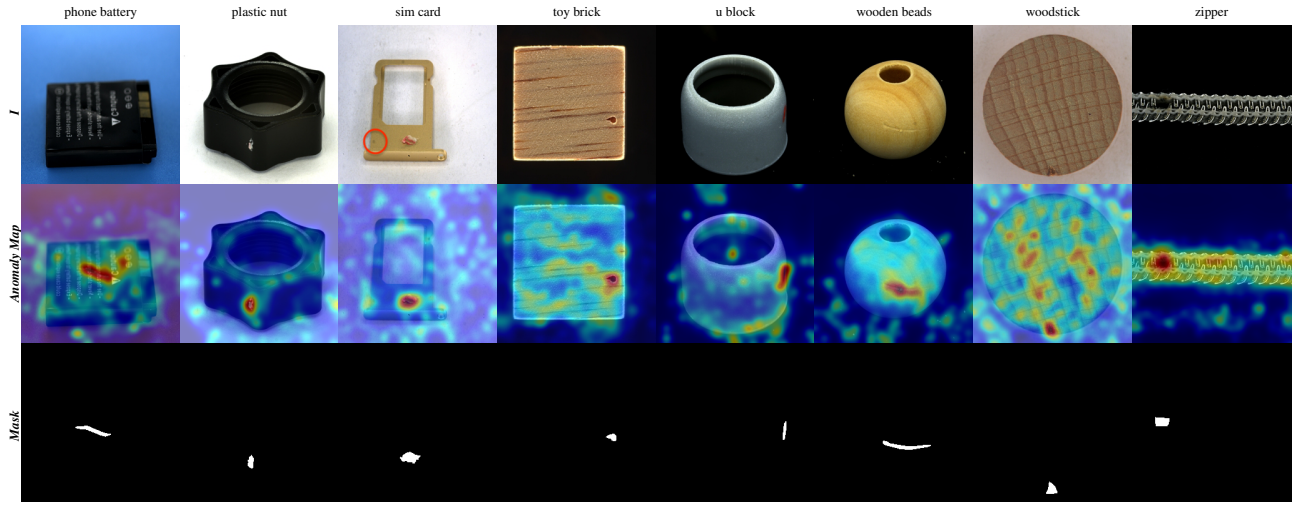

Figure 3. Example anomalous images from Real-IAD dataset, and predicted anomaly segmentation maps using VLMDiff.

Dataset	Categories		Images		
	Train	Test	Train	Test	
			Normal	Anomaly	Normal
Real-IAD [46]	30	30	36,465	51,329	63,256
COCO-AD [59]	61	81	30,438	1,291	3,661
			65,133	2,785	2,167
			79,083	3,328	1,624
			77,580	3,253	1,699

Table 1. Details of the datasets used in experiments.

detection COCO-AD [59]. Real-IAD contains 30 objects with multiple images from different views and has orders of magnitude more training and test images compared to previous defect detection datasets like MVTec-AD [3] and VISA [67]. Moreover, the dataset includes a diverse range of defect types, including very small and challenging anomalies.

COCO-AD [59] fills the need for having a real-world anomaly localization dataset. It is derived from the original COCO [27] dataset by choosing consecutive 20 classes as anomalies out of present 80 classes in each split. The images that contain any one of the selected 20 classes are removed from the training sets. Similarly, validation set is constructed by selecting the images which contain any anomaly classes labeled as anomalies, and the rest are labeled as normal. Details of both datasets are presented in Table 1.

4.2. Experimental Setup

We implemented our method using PyTorch [36]. For all datasets, images are resized to 256×256 pixels before training. All models are trained on Nvidia A100 GPUs. Below, we give implementation and training details for all subnetworks.

Image Autoencoder is trained using only normal training (anomaly-free) images with KL divergence loss on the latent space, and GAN [16] objective on the image space. We trained the autoencoder for 50 epochs on Real-IAD and COCO-AD datasets using a batch size of 32 and a learning rate of $4.5e-5$. Our autoencoder implementation is the same as the one used in LDM [39].

Denoising Model Denoising U-Net weights are initialized using the pretrained Stable Diffusion v1.5 weights [1]. We trained our models with a batch size of 12 and a learning rate of $1e-5$ using the training approach presented in ControlNet [61]. LDM is conditioned on CLIP-encoded [38] text descriptions extracted from VLM.

VLM. Text descriptions for train images are extracted using InternVL-2 [8] $8B$ version which achieves state-of-the-art performance on various benchmarks. We did not conduct any prompt learning or searching to devise $\mathcal{P}_D$.

State-of-the-art comparison. VLMDiff is compared with methods from all AD categories and the results are reported in Tables 2 and 3. The most recent diffusion-based multi-class (DiAD [20]) and single-class (TransFusion [15]) methods are included. All methods are trained for 100 epochs using their official codes. To compare with other methods, we used the results reported in a recent survey [58] from the authors of the Real-IAD [46] and COCO-AD [59] datasets.

In the supplementary material, we present per-class results on Real-IAD, per-split results on COCO-AD, additional quantitative results on the MVTec-AD [3] and VISA [67] datasets, and more visual comparisons on Real-IAD and COCO-AD datasets.

[1] huggingface.co/stable-diffusion-v1-5/stable-diffusion-v1-5

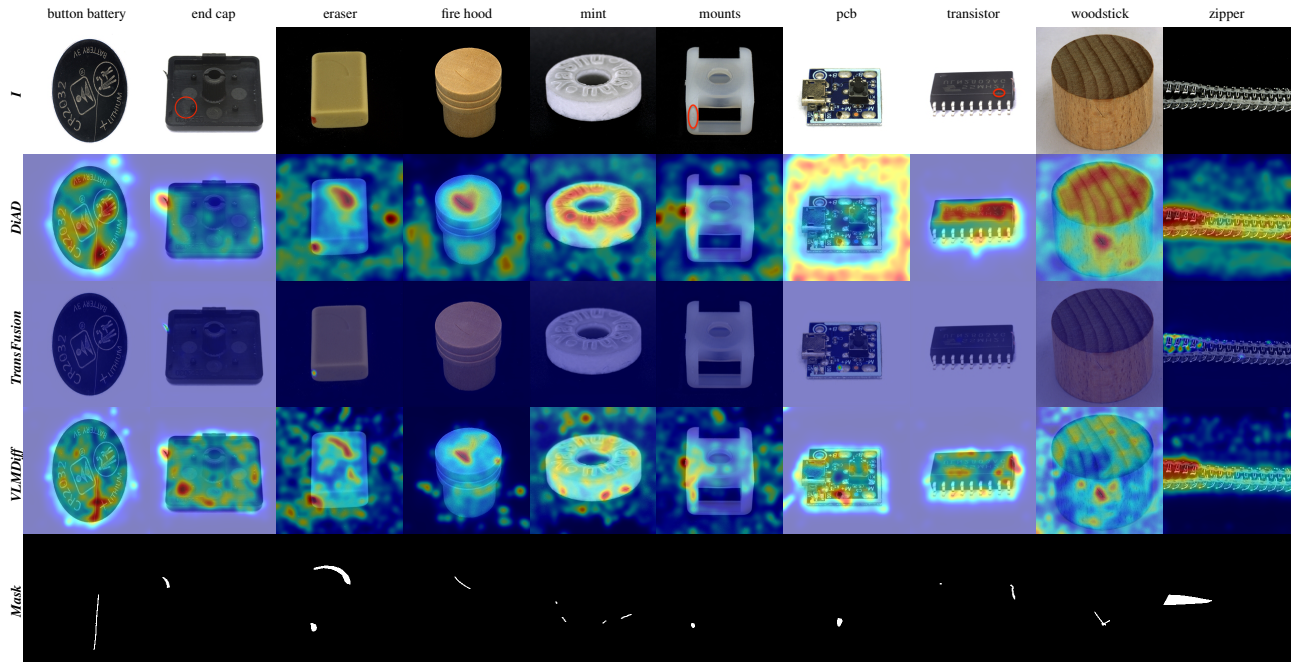

Figure 4. Visual comparison of diffusion-based methods on Real-IAD dataset. DiAD detects the anomaly locations with the expense of having many false positives.

4.3. Qualitative Results

Anomaly maps of VLMDiff are shown in Figure 3, and a visual comparison of diffusion-based methods on a subset of Real-IAD classes is presented in Figure 4. DiAD detects many normal locations as anomalies with high confidence. Whereas Transfusion misses many anomalous pixels, and the detected ones have low confidence values. Especially when the defects are not very severe, such as scratches, TransFusion could not localize, most likely due to the employed noise generation scheme. VLMDiff detects various types of defects correctly, and even sometimes locates missed annotations (denoted as red circles) as in the cases of *end cap*, *mounts* and *transistor* classes. This focus on fine-grained localization, however, is leading to a lower ROC_I score compared to baselines.

We show example results from four splits of the COCO-AD dataset in Figure 5. TransFusion failed to detect many anomaly locations as expected, since the method is bound to a specific domain. On the other hand, the domain-independent methods, DiAD and VLMDiff, successfully locate the anomalous objects even if the scene is highly complex. DiAD performs well when there is a single anomaly class present in a given image such as the first and second columns where *person* and *skateboard* are the only anomaly objects, respectively. However, in the last column where three anomaly classes are present, DiAD only detects parts of the *keyboard* object correctly. On all splits, VLMDiff achieves the best localization performance, even detecting both of the small *mouse* objects in the last column.

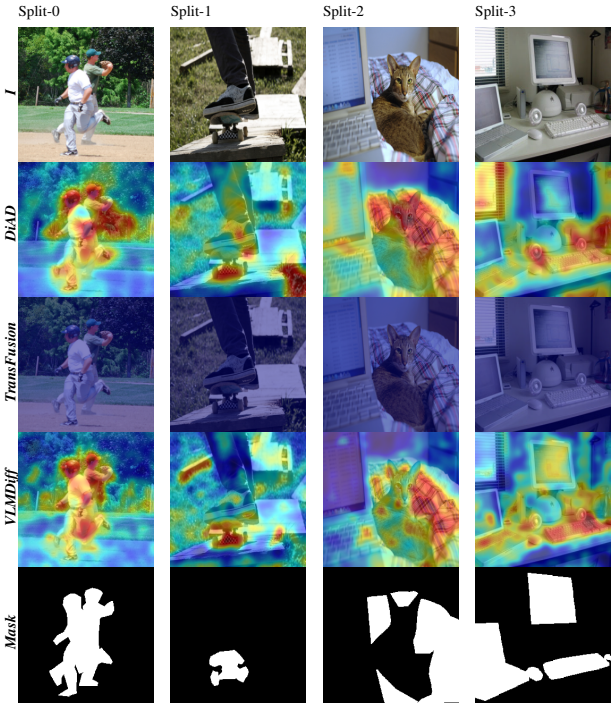

Figure 5. Visual comparison of diffusion-based methods on COCO-AD dataset.

4.4. Quantitative Results

Metrics. Following the common practice, we used three metrics: Mean Area under the ROC curve mAU –

	Method	ROC_I	ROC_P	PRO
Aug.	DRAEM [55]	50.9	44.0	13.6
	SimpleNet [29]	54.9	76.1	42.4
Emb.	CFA [24]	55.7	81.3	48.8
	CFLOW-AD [18]	77.0	94.8	80.4
	PyramidalFlow [25]	54.4	71.1	34.9
Hyb.	UniAD † [53]	83.1	97.4	87.1
	RD++ [45]	83.6	97.7	<u>90.7</u>
	DesTSeg [63]	79.3	80.3	56.1
Rec.	RD [13]	82.7	97.2	90.0
	ViTAD † [60]	82.7	97.2	84.8
	MambaAD [19]	<u>86.3</u>	<u>98.5</u>	90.5
Dif.	TransFusion [15]	**78.6**	84.2	61.6
	DiAD † [20]	75.6	88.0	58.1
	VLMDiff †	78.0	**97.1**	**87.7**

Table 2. Results on the large scale industrial AD REAL-IAD [46] dataset. †: multi-class setting

	Method	ROC_I	ROC_P	PRO
Aug.	DRAEM [55]	53.5	49.9	15.3
	SimpleNet [29]	55.6	60.2	26.1
Emb.	CFA [24]	56.7	56.2	17.9
	CFLOW-AD [18]	<u>67.7</u>	76.0	<u>47.7</u>
	PyramidalFlow [25]	51.6	50.0	15.0
Hyb.	UniAD † [53]	55.2	64.6	34.3
	RD++ [45]	57.5	68.2	42.2
	DesTSeg [63]	54.4	54.5	24.4
Rec.	RD [13]	57.6	66.5	39.8
	ViTAD † [60]	66.9	<u>76.2</u>	39.1
	MambaAD [19]	62.8	68.9	41.6
Dif.	TransFusion [15]	58.4	57.8	6.8
	DiAD † [20]	59.0	68.1	30.8
	VLMDiff †	**59.1**	**69.0**	**38.8**

Table 3. Results on the real world COCO-AD [59] dataset.

Variants	VLM Desc.		Metrics		
	Train	Inference	ROC_I	ROC_P	PRO
DiAD [20]	✗	✗	75.6	88.0	58.1
VLMDiff	✓	✓	72.6	95.9	84.0
VLMDiff	✓	✗	**78.0**	**97.1**	**87.7**

Table 4. Ablation experiments on REAL-IAD [46] using VLM descriptions for training, inference or both.

ROC [55] on image and pixel levels, and pixel-level Per-Region-Overlap PRO [4] score. Note that for a fair comparison, we use the same evaluation script for all the diffusion models, which consistently gave better results compared to the original ones used in each method.

Real-IAD. Quantitative results are presented in Table 2. Our method significantly outperforms diffusion-based approaches on the pixel-level ROC_P and PRO metrics, and achieves on-par performance compared to state-of-the-art methods. DiAD [20] suffers from the detection of normal regions as anomalies, and TransFusion [15] misses many anomalous locations. We observe that there are some irregularities in the background of the Real-IAD images, which are sometimes detected as anomalies with DiAD and our method. Hence, both of these methods have relatively low image-level scores compared to TransFusion. On the other hand, although TransFusion performs well when the defect is obvious, such as a dent or crack, it fails to detect most of the defect types, such as scratches and discolorations.

COCO-AD. In order to show the generalization capability of our method to other domains, we experimented with a real-world AD dataset, see Table 3. In comparison to diffusion methods, VLMDiff improved the PRO metric by 8 points, which shows that our method is not bound to any specific domain and could be used in real-world AD as well. VLMDiff achieves on par results on all three metrics compared to other state-of-the-art approaches.

4.5. Ablation Experiments

Effect of using text descriptions in inference. Recall that we train our model with only normal images and corresponding textual descriptions extracted using a pretrained VLM. During inference, we experimented with two variants; a) using textual descriptions with the P_D and b) using no text description at all. The results are presented in Table 4 for Real-IAD dataset, along with the no VLM baseline DiAD [20]. The test without any textual descriptions achieves the best overall performance. We observed that for some of the test images with a visible defect, the VLM generates descriptions with additional anomaly info such as *horizontal crack present*, *there is a dent on the battery*. These descriptions guide the diffusion model to generate an image with the mentioned defects which harms the performance of the VLMDiff.

However, on the real image dataset COCO-AD where the anomalies are unseen objects, using textual descriptions boosts the performance by 3 points on every metric.

Exploring different Vision Language Models. Second, we explore different VLMs to guide the training of the diffusion model, see Table 5. In addition to InternVL-2, we also extract image descriptions using Blip2 [26] and the recent DeepSeekVL-v3 [28] model. We observe that Blip2 generates very short descriptions and fails to describe the objects in detail, especially in the industrial domain datasets. This phenomenon is reflected in the results, and Blip2 achieved the lowest performance. On the other hand,

Variants	Real-IAD			COCO-AD		
	ROC_I	ROC_P	PRO	ROC_I	ROC_P	PRO
Blip2 [26]	76.8	96.6	86.3	59.9	68.5	38.1
DeepSeekv3-1.3B [28]	77.2	97.0	87.5	58.5	68.1	38.1
InternVL-2-8B [8]	78.0	97.1	87.7	59.1	69.0	38.8
GT caption	N/A	N/A	N/A	61.1	66.5	39.9

Table 5. Comparison of different VLMs for extracting image descriptions. We also use available human captions on COCO-AD. InternVL-2 achieved the best overall performance.

SD AE	Metrics		
Train	ROC_I	ROC_P	PRO
✗	77.0	95.5	83.7
✓	**78.0**	**97.1**	**87.7**

Table 6. The effect of finetuning the image auto-encoder.

DeepSeekVL and InternVL describe the main object or the image very clearly by focusing on details. On all datasets, these two VLMs achieved similar performance. On the COCO-AD dataset, we also train a model using the available ground truth image captions coming from five different annotators. This model achieved the highest image-level score because the captions provide information about all the normal classes present in the images.

Varying Feature Extractor. We experimented with variants of the DINO [7], DINO-v2 [34] and also an ImageNet [14] pretrained ResNet [21] model for the segmentation part in Table 7. Using an ImageNet pretrained model as the feature extractor boosts the image-level metric by almost 3 points. The main reason for this increase stems from the strong confidence scores for the anomaly locations. Using the DINO model with big patch sizes drops the performance on the pixel-level metrics as expected. Since the defects on the industrial datasets are very small, having a single feature for a large image patch suffers dramatically. Note that, for the DINO-v2, the test image is resized to 224×224 pixels.

Using pretrained Image Auto-encoder. As an ablation experiment (see Table 6), we trained VLMDiff on Real-IAD dataset using the baseline Stable Diffusion image auto-encoder weights without any finetuning. Note that in this setting we are training only the denoising UNet. The performance metrics drop significantly, e.g. ROC_I 1.0, ROC_P 1.6 and PRO 4.0 points, which shows the importance of finetuning the image auto-encoder in specialized datasets.

Using only VLM descriptions for AD. As an additional experiment, we use the VLM to determine only the presence or absence of anomalies and evaluate its image-level performance in terms of image-level ROC (ROC_I) for each class. On the Real-IAD dataset, the average ROC_I across all classes is 54.5, which is substantially lower than the performance achieved by our method (cf. Table 2). This highlights the limitations of relying solely on VLM responses for anomaly detection and underscores the advantage of our approach in using VLMs for AD. For a more detailed analysis of using VLMs for only anomaly detection, refer to [50].

Model	Metrics		
	ROC_I	ROC_P	PRO
DINO-v2 ViTS/14	61.4	58.3	16.1
DINO R50	65.5	88.6	56.8
DINO ViTB/16	77.2	94.6	77.5
DINO ViTS/16	75.6	94.7	76.8
DINO ViTB/8	76.6	96.4	85.7
DINO ViTS/8	78.0	**97.1**	**87.7**
ImageNet R50	**81.1**	96.3	83.6

Table 7. Ablation study on using variants of DINO and an ImageNet pretrained ResNet-50 on the Real-IAD.

4.6. Limitations

Our current approach utilizes a fixed, simple query to extract image descriptions from VLMs, a design choice made to focus on the core diffusion-based anomaly detection method. While this straightforward strategy already significantly improves upon previous diffusion-based anomaly detection methods, the full potential of VLMs could be further harnessed. Future work could explore optimizing these queries through prompt learning techniques. Additionally, designing queries tailored to the specific characteristics of each dataset could yield richer descriptions and, consequently, lead to enhanced performance.

5. Conclusion

Our proposed framework, VLMDiff, effectively bridges the gap between the descriptive power of VLMs and the reconstruction capabilities of diffusion models for multi-class anomaly detection. By leveraging simple, generic prompts and avoiding the limitations of per-class training and synthetic anomaly generation, VLMDiff significantly advances the state-of-the-art, achieving substantial performance gains, particularly in pixel-level metrics. Our approach not only enhances generalization, scalability, and efficiency but also opens new avenues for robust and practical anomaly detection in diverse and complex real-world scenarios.

Acknowledgments. This research was funded in whole or in part by the Luxembourg National Research Fund (FNR), grant reference DEFENCE22/17813724/AUREA.

References

[1] Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *ACCV*, pages 622–637, 2019. 1, 2

[2] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. *arXiv preprint arXiv:1807.02011*, 2018. 1, 2

[3] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*, pages 9592–9600, 2019. 5, 1

[4] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *CVPR*, 2020. 2, 7

[5] Daniel Bogdoll, Maximilian Nitsche, and J Marius Zöllner. Anomaly detection in autonomous driving: A survey. In *CVPRW*, pages 4488–4499, 2022. 1

[6] Yunkang Cao, Jiangning Zhang, Luca Frittoli, Yuqi Cheng, Weiming Shen, and Giacomo Boracchi. Adaclip: Adapting clip with hybrid learnable prompts for zero-shot anomaly detection. In *ECCV*, pages 55–72, 2024. 2, 3

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pages 9650–9660, 2021. 4, 8

[8] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *CVPR*, pages 24185–24198, 2024. 5, 8

[9] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, pages 3606–3613, 2014. 3

[10] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. *CoRR*, abs/2005.02357, 2020. 2

[11] Yajie Cui, Zhaoxiang Liu, and Shiguo Lian. A survey on unsupervised anomaly detection algorithms for industrial images. *IEEE Access*, 11:55297–55315, 2023. 1

[12] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: A patch distribution modeling framework for anomaly detection and localization. In *Pattern Recognition. ICPR International Workshops and Challenges*, 2021. 2

[13] Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *CVPR*, pages 9737–9746, 2022. 7, 1

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 2, 8

[15] Matic Fučka, Vitjan Zavrtanik, and Danijel Skočaj. TransFusion – A Transparency-Based Diffusion Model for Anomaly Detection. In *ECCV*, 2024. 2, 3, 5, 7, 1

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, pages 2672–2680, 2014. 5

[17] Zhaopeng Gu, Bingke Zhu, Guibo Zhu, Yingying Chen, Hao Li, Ming Tang, and Jinqiao Wang. Filo: Zero-shot anomaly detection by fine-grained description and high-quality localization. In *ACM MM*, pages 2041–2049, 2024. 2, 3

[18] Denis Gudovskiy, Shun Ishizaka, and Kazuki Kozuka. Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *WACV*, pages 98–107, 2022. 1, 2, 7

[19] Haoyang He, Yuhu Bai, Jiangning Zhang, Qingdong He, Hongxu Chen, Zhenye Gan, Chengjie Wang, Xiangtai Li, Guanzhong Tian, and Lei Xie. Mambaad: Exploring state space models for multi-class unsupervised anomaly detection. *arXiv preprint arXiv:2404.06564*, 2024. 1, 7

[20] Haoyang He, Jiangning Zhang, Hongxu Chen, Xuhai Chen, Zhishan Li, Xu Chen, Yabiao Wang, Chengjie Wang, and Lei Xie. A diffusion-based framework for multi-class anomaly detection. In *AAAI*, pages 8472–8480, 2024. 1, 2, 3, 5, 7

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 2, 8

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, pages 6840–6851, 2020. 1, 2, 3

[23] Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. Winclip: Zero-/few-shot anomaly classification and segmentation. In *CVPR*, 2023. 3

[24] Sungwook Lee, Seunghyun Lee, and Byung Cheol Song. Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. *IEEE Access*, 10: 78446–78454, 2022. 1, 2, 7

[25] Jiarui Lei, Xiaobo Hu, Yue Wang, and Dong Liu. Pyramidflow: High-resolution defect contrastive localization using pyramid normalizing flow. In *CVPR*, pages 14143–14152, 2023. 1, 2, 7

[26] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, pages 19730–19742, 2023. 3, 7, 8

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 5

[28] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024. 7, 8

[29] Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. Simplenet: A simple network for image anomaly detection and localization. In *CVPR*, pages 20402–20411, 2023. 1, 2, 7

[30] Victor Livernoche, Vineet Jain, Yashar Hezaveh, and Siamak Ravanbakhsh. On diffusion modeling for anomaly detection. In *ICLR*, 2024. 1, 3

[31] Philipp Liznerski, Lukas Ruff, Robert A. Vandermeulen, Billy Joe Franks, Klaus Robert Muller, and Marius Kloft. Exposing outlier exposure: What can be learned from few, one, and zero outlier images. *Transactions on Machine Learning Research*, 2022. 3

[32] Atsuyuki Miyai, Jingkang Yang, Jingyang Zhang, Yifei Ming, Yueqian Lin, Qing Yu, Go Irie, Shafiq Joty, Yixuan Li, Hai Li, et al. Generalized out-of-distribution detection and beyond in vision language model era: A survey. *arXiv preprint arXiv:2407.21794*, 2024. 3

[33] Paolo Napoletano, Flavio Piccoli, and Raimondo Schettini. Anomaly detection in nanofibrous materials by cnn-based self-similarity. *Sensors*, 18, 2018. 2

[34] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. 8

[35] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2):1–38, 2021. 1

[36] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 5

[37] Ken Perlin. An image synthesizer. 1985. 2, 3

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 3, 4, 5

[39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 1, 2, 4, 5

[40] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention*, pages 234–241, 2015. 4

[41] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *CVPR*, pages 14318–14328, 2022. 2, 1

[42] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. *Medical image analysis*, 54:30–44, 2019. 2

[43] Abd El Rahman Shabayek, Arunkumar Rathinam, Matthieu Ruthven, Djamila Aouada, and Tazdin Amietszajew. Ai-enabled thermal monitoring of commercial (phev) li-ion pouch cells with feature-adapted unsupervised anomaly detection. *Journal of Power Sources*, 2025. 1, 2

[44] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, pages 2256–2265, 2015. 1

[45] Tran Dinh Tien, Anh Tuan Nguyen, Nguyen Hoang Tran, Ta Duc Huy, Soan Duong, Chanh D Tr Nguyen, and Steven QH Truong. Revisiting reverse distillation for anomaly detection. In *CVPR*, pages 24511–24520, 2023. 1, 7

[46] Chengjie Wang, Wenbing Zhu, Bin-Bin Gao, Zhenye Gan, Jiangning Zhang, Zhihao Gu, Shuguang Qian, Mingang Chen, and Lizhuang Ma. Real-iad: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. In *CVPR*, pages 22883–22892, 2024. 1, 2, 3, 4, 5, 7

[47] Shenzhi Wang, Liwei Wu, Lei Cui, and Yujun Shen. Glancing at the patch: Anomaly localization with global and local feature comparison. In *CVPR*, 2021. 2

[48] Julia Wolleb, Florentin Bieder, Robin Sandkühler, and Philippe C Cattin. Diffusion models for medical anomaly detection. In *International Conference on Medical image computing and computer-assisted intervention*, pages 35–45, 2022. 1, 2, 3

[49] Julian Wyatt, Adam Leach, Sebastian M Schmon, and Chris G Willcocks. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *CVPRW*, pages 650–656, 2022. 1, 2, 3

[50] Jiacong Xu, Shao-Yuan Lo, Bardia Safaei, Vishal M Patel, and Isht Dwivedi. Towards zero-shot anomaly detection and reasoning with multimodal large language models. In *CVPR*, pages 20370–20382, 2025. 3, 8

[51] Ruiyao Xu and Kaize Ding. Large language models for anomaly and out-of-distribution detection: A survey. *arXiv preprint arXiv:2409.01980*, 2024. 3

[52] Hang Yao, Ming Liu, Zhicun Yin, Zifei Yan, Xiaopeng Hong, and Wangmeng Zuo. Glad: towards better reconstruction with global and local adaptive diffusion models for unsupervised anomaly detection. In *ECCV*, pages 1–17, 2024. 3

[53] Zhiyuan You, Lei Cui, Yujun Shen, Kai Yang, Xin Lu, Yu Zheng, and Xinyi Le. A unified model for multi-class anomaly detection. *NeurIPS*, 35:4571–4584, 2022. 1, 7

[54] Luca Zanella, Willi Menapace, Massimiliano Mancini, Yiming Wang, and Elisa Ricci. Harnessing large language models for training-free video anomaly detection. In *CVPR*, pages 18527–18536, 2024. 2, 3

[55] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Draem-a discriminatively trained reconstruction embedding for sur-

face anomaly detection. In *ICCV*, pages 8330–8339, 2021. 1, 2, 3, 7

[56] Hao Zhang, Feng Li, Xueyan Zou, Shilong Liu, Chunyuan Li, Jianwei Yang, and Lei Zhang. A simple framework for open-vocabulary segmentation and detection. In *ICCV*, pages 1020–1031, 2023. 4

[57] Hui Zhang, Zheng Wang, Zuxuan Wu, and Yu-Gang Jiang. Diffusionad: Norm-guided one-step denoising diffusion for anomaly detection. *arXiv preprint arXiv:2303.08730*, 2023. 1, 2, 3

[58] Jiangning Zhang, Haoyang He, Zhenye Gan, Qingdong He, Yuxuan Cai, Zhucun Xue, Yabiao Wang, Chengjie Wang, Lei Xie, and Yong Liu. Ader: A comprehensive benchmark for multi-class visual anomaly detection. *arXiv preprint arXiv:2406.03262*, 2024. 5, 2

[59] Jiangning Zhang, Chengjie Wang, Xiangtai Li, Guanzhong Tian, Zhucun Xue, Yong Liu, Guansong Pang, and Dacheng Tao. Learning feature inversion for multi-class anomaly detection under general-purpose coco-ad benchmark. *arXiv preprint arXiv:2404.10760*, 2024. 2, 3, 4, 5, 7, 1

[60] Jiangning Zhang, Xuhai Chen, Yabiao Wang, Chengjie Wang, Yong Liu, Xiangtai Li, Ming-Hsuan Yang, and Dacheng Tao. Exploring plain vit reconstruction for multi-class unsupervised anomaly detection. *CVIU*, 2025. 1, 2, 7

[61] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *CVPR*, pages 3836–3847, 2023. 5

[62] Xinyi Zhang, Naiqi Li, Jiawei Li, Tao Dai, Yong Jiang, and Shu-Tao Xia. Unsupervised surface anomaly detection with diffusion probabilistic model. In *ICCV*, pages 6782–6791, 2023. 1, 2, 3

[63] Xuan Zhang, Shiyu Li, Xi Li, Ping Huang, Jiulong Shan, and Ting Chen. Destseg: Segmentation guided denoising student-teacher for anomaly detection. In *CVPR*, pages 3914–3923, 2023. 1, 7

[64] Ximiao Zhang, Min Xu, and Xiuzhuang Zhou. Realnet: A feature selection network with realistic synthetic anomaly for anomaly detection. In *CVPR*, pages 16699–16708, 2024. 2, 1

[65] Qihang Zhou, Guansong Pang, Yu Tian, Shibo He, and Jiming Chen. Anomalyclip: Object-agnostic prompt learning for zero-shot anomaly detection. In *ICLR*, 2024. 2, 3

[66] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *NeurIPS*, 36:19769–19782, 2023. 4

[67] Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *ECCV*, pages 392–408, 2022. 1, 5

VLMDiff: Leveraging Vision-Language Models for Multi-Class Anomaly Detection with Diffusion

Supplementary Material

6. Overview

In this supplementary material, we first share the quantitative results on MVTec-AD [3] and VISA [67] datasets and comparisons with the same methods presented in the main text. Then, we present per-class results on the Real-IAD [46], and per-split results on the COCO-AD [59] dataset, only for the diffusion-based methods to compare. Finally, we show more visual results of our method and diffusion methods on each class of the Real-IAD dataset, and on each split of the COCO-AD dataset.

Table 8. Details of MVTec-AD and VISA datasets.

Dataset	Categories		Images		
	Train	Test	Train	Test	
			Normal	Anomaly	Normal
MVTec AD [3]	15	15	3,629	1,258	467
VisA [67]	12	12	8,659	962	1,200

6.1. MVTec and VISA results

Dataset statistics for MVTec-AD [3] and VISA [67] are presented in Table 8. We trained the best-performing diffusion methods for 100 and 300 epochs on the MVTec-AD [3] and VISA [67] datasets to compare their performance on the same epoch training regime. We present the results in Table 9 and Table 10 for MVTec-AD and VISA, respectively. On MVTec-AD, our method achieved the best ROC_P and PRO scores among the diffusion-based approaches, which show the exceptional localization performance of VLMDiff. VISA dataset results show similar patterns and our method achieved the best ROC_P score by improving more than 5 points.

	Method	ROC_I	ROC_P	PRO
Aug.	DRAEM [55]	54.5/55.2	47.6/48.7	14.3/15.8
	SimpleNet [29]	95.4/79.2	96.8/82.4	86.9/62.0
	RealNet [64]	84.8/82.9	72.6/69.8	56.8/51.2
Emb.	CFA [24]	57.6/55.8	54.8/43.9	25.3/19.3
	PatchCore [41]	98.8/ -	98.3/ -	94.2 / -
	CFLOW-AD [18]	91.6/92.7	95.7/95.8	88.3/89.0
	PyramidalFlow [25]	70.2/66.2	80.0/74.2	47.5/40.0
Hyb.	UniAD † [53]	92.5/96.8	95.8/96.8	89.3/91.0
	RD++ [45]	97.9/95.8	97.3/97.3	93.2/92.9
	DesTSeg [63]	96.4/96.3	92.0/92.6	83.4/82.6
Rec.	RD [13]	93.6/90.5	95.8/95.9	91.2/91.2
	ViTAD † [60]	98.3/98.4	97.6/97.5	92.0/91.7
	MambaAD [19]	97.8/98.5	97.4/97.6	93.4/93.6
Dif.	DiffAD [62]	80.7/91.8	79.7/88.4	65.1/78.4
	TransFusion [15]	90.4/**95.3**	80.9/90.6	72.4/83.5
	DiAD † [20]	88.9/92.0	89.3/89.3	63.9/64.4
	VLMDiff †	86.9/90.6	94.9/**95.9**	86.7/**89.4**

Table 9. Results on the MVTec AD dataset [3] for 100/300 epochs training. †: multi-class setting.

	Method	ROC_I	ROC_P	PRO
Aug.	DRAEM [55]	55.1/56.2	37.5/45.0	10.0/16.0
	SimpleNet [29]	86.4/80.7	96.6/94.4	79.2/74.2
	RealNet [64]	71.4/79.2	61.0/65.4	27.4/33.9
Emb.	CFA [24]	66.3/67.1	81.3/83.0	50.8/48.7
	CFLOW-AD [18]	86.5/87.2	97.7/97.8	86.8/87.3
	PyramidalFlow [25]	58.2/69.0	77.0/79.1	42.8/52.6
Hyb.	UniAD † [53]	89.0/91.4	98.3/98.5	86.5/89.0
	RD++ [45]	93.9/93.1	98.4/98.4	91.9/91.4
	DesTSeg [63]	89.9/89.0	86.7/84.8	61.1/57.5
Rec.	RD [13]	90.6/93.9	98.0/98.1	91.9/91.9
	ViTAD [60]	90.4/90.3	98.2/98.2	85.7/85.8
	MambaAD [19]	94.5/93.6	98.4/98.2	92.1/90.5
Dif.	DiffAD [62]	78.6/89.2	82.9/85.5	65.7/76.7
	TransFusion [15]	87.4/**92.5**	82.1/90.3	55.4/64.7
	DiAD † [20]	84.8/90.5	82.5/83.4	44.5/44.3
	VLMDiff †	79.0/80.9	96.0/**97.0**	77.0/**81.0**

Table 10. Results on the VISA AD dataset [67] for 100/300 epochs training. †: multi-class setting.

We conducted extended ablation studies to thoroughly evaluate our method. These experiments focused on three key aspects: 1) the choice of VLMs for extracting image descriptions during training, 2) the impact of including a specific prompt during the inference stage, and 3) the selection of the feature extractor for inference.

Our comparison of VLMs for image description extraction (Table 11) revealed that InternVL-2 consistently achieved the best overall performance across both datasets. Further investigation into inference-time prompting (Table 12) with InternVL-2 showed that employing prompt $\mathcal{P}_\mathcal{D}$ led to a noticeable performance drop on both datasets. Lastly, our analysis of different feature extractors during inference (Table 13) indicated that DINO ViT-S with a patch

Variants	MVTec-AD			VISA		
	ROC_I	ROC_P	PRO	ROC_I	ROC_P	PRO
Blip2	89.8	95.7	88.6	82.3	97.0	80.7
DeepSeekv3-1.3B	90.7	95.5	89.0	81.8	96.8	81.0
InternVL-2-8B	90.6	95.9	89.4	80.9	97.0	81.0

Table 11. Ablation experiments using different VLMs to extract anomaly descriptions on MVTec-AD and VISA datasets.

Dataset	ROC_I	ROC_P	PRO
MVTec-AD	-2.2	-2.4	-6.5
VISA	-8.1	-2.9	-9.9

Table 12. Relative change when we use $\mathcal{P}_{\mathcal{D}}$ query during inference. Using text description from InternVL-2 for inference has a negative impact on all metrics.

size of 8 delivered the strongest overall results.

6.2. COCO-AD per split results

Per-split results on COCO-AD [59] are shown in Table 14. VLMDiff shows a noticeable improvement compared to the baselines, especially in the first split where there are significantly fewer normal images.

6.3. Real-IAD per class results

Tables 15 and 16 present per-class results on the Real-IAD [46] dataset for diffusion-based methods. Except for a few cases, VLMDiff achieves the best ROC_P and PRO on all classes. A detailed overview of the performance of other methods can be found in [58].

6.4. More visuals for Real-IAD

We present more visual comparisons in Figures 6-11 on Real-IAD. Specifically, we show two results per object category in the dataset. VLMDiff shows superior localization capability compared to strong baselines. Moreover, in some cases, we observe that it even finds unmarked potential defective pixels. For instance, in Figure 6 first *bottle cap* image has a small blue dot which is marked as an anomaly by our method. Similarly, both *sim card* objects have small defective pixels which are again detected by VLMDiff.

6.5. More visuals for COCO-AD

Figures 12 and 13 show three example results per split, and in each split, we pick different anomaly classes to show the performance across various objects. As a real-world dataset, COCO-AD is more complex and challenging compared to previous industrial domain datasets. Nevertheless, VLMDiffachieves significantly better anomaly localization across multiple classes.

Model	Metrics		
	ROC_I	ROC_P	PRO
ImageNet R50	88.8	92.6	81.0
DINO-v2 ViTS/14	64.6	61.3	16.8
DINO R50	86.8	90.3	71.0
DINO ViTB/16	89.8	92.8	79.3
DINO ViTS/16	85.1	90.2	74.3
DINO ViTB/8	87.5	93.6	82.6
DINO ViTS/8	**90.6**	**95.9**	**89.4**

Table 13. Ablation experiments using variants of DINO and an ImageNet pretrained Resnet-50 for anomaly segmentation on MVTec-AD dataset with InternVL-2 descriptions.

	Method	ROC_I	ROC_P	PRO
Split-0	DiAD †	57.5	67.0	28.8
	TransFusion	56.1	54.8	12.8
	VLMDiff †	**62.6**	**74.3**	**43.8**
Split-1	DiAD †	54.4	71.3	28.8
	TransFusion	**57.1**	62.2	6.6
	VLMDiff †	52.7	**71.5**	**37.5**
Split-2	DiAD †	**63.8**	68.0	33.2
	TransFusion	61.4	58.4	2.7
	VLMDiff †	62.9	**69.3**	**40.7**
Split-3	DiAD †	**60.1**	**65.9**	32.3
	TransFusion	59.0	56.1	5.0
	VLMDiff †	58.2	60.8	**33.2**
Avg	DiAD †	59.0	68.1	30.8
	TransFusion	58.4	57.8	6.8
	VLMDiff †	**59.1**	**69.0**	**38.8**

Table 14. Per split results on the COCO-AD dataset [59] for 100 epochs training. †: multi-class setting.

	Method	ROC_I	ROC_P	PRO
audiojack	DiAD	76.5	91.6	63.3
	Transfusion	**80.3**	85.9	51.2
	VLMDiff	77.5	**97.8**	**87.1**
bottle cap	DiAD	**91.6**	94.6	73.0
	Transfusion	65.4	70.9	43.3
	VLMDiff	77.3	**98.4**	**92.3**
button battery	DiAD	80.5	84.1	66.9
	Transfusion	**88.1**	94.5	**76.8**
	VLMDiff	72.8	**96.9**	74.5
end cap	DiAD	**85.1**	81.3	38.2
	Transfusion	64.3	56.6	32.8
	VLMDiff	71.6	**96.4**	**85.9**
eraser	DiAD	**80.0**	91.1	67.5
	Transfusion	74.3	75.8	55.1
	VLMDiff	78.9	**98.2**	**89.6**
fire hood	DiAD	**83.3**	91.8	66.7
	Transfusion	72.0	84.9	57.7
	VLMDiff	73.2	**98.3**	**89.4**
mint	DiAD	**76.7**	91.1	64.2
	Transfusion	60.8	68.8	30.8
	VLMDiff	64.3	**93.8**	**73.0**
mounts	DiAD	75.3	84.3	48.8
	Transfusion	**81.5**	86.1	73.2
	VLMDiff	78.7	**98.8**	**93.2**
pcb	DiAD	**86.0**	92.0	66.5
	Transfusion	77.7	94.9	64.6
	VLMDiff	82.5	**98.3**	**88.9**
phone battery	DiAD	**82.3**	**96.8**	85.4
	Transfusion	77.0	88.0	66.6
	VLMDiff	80.8	91.1	**90.2**
plastic nut	DiAD	71.9	81.1	38.6
	Transfusion	75.4	90.7	59.5
	VLMDiff	**80.6**	**98.7**	**93.4**
plastic plug	DiAD	**88.7**	92.9	66.1
	Transfusion	82.2	91.9	76.6
	VLMDiff	70.5	**97.1**	**86.0**
porcelain doll	DiAD	**72.6**	93.1	70.4
	Transfusion	70.2	85.8	64.2
	VLMDiff	**72.6**	**97.7**	**88.7**
regulator	DiAD	72.1	84.2	44.4
	Transfusion	**74.3**	81.1	49.0
	VLMDiff	61.7	**97.7**	**88.0**
rolled strip	DiAD	68.4	87.7	63.4
	Transfusion	**98.0**	87.1	81.0
	VLMDiff	86.6	**99.6**	**98.3**

Table 15. Per class results on Real-IAD dataset for diffusion models, part 1.

	Method	ROC_I	ROC_P	PRO
sim card	DiAD	72.6	89.9	60.4
	Transfusion	91.8	96.6	82.5
	VLMDiff	**92.9**	**98.3**	**90.0**
switch	DiAD	73.4	90.5	64.2
	Transfusion	82.0	86.6	59.4
	VLMDiff	**83.9**	**96.8**	**90.7**
tape	DiAD	73.9	81.7	47.3
	Transfusion	**91.9**	94.6	83.7
	VLMDiff	89.5	**99.3**	**96.9**
terminalblock	DiAD	62.1	75.5	38.5
	Transfusion	70.6	85.6	70.3
	VLMDiff	**82.1**	**99.4**	**96.2**
toothbrush	DiAD	**91.2**	82.0	54.5
	Transfusion	88.5	87.5	66.1
	VLMDiff	80.3	**95.1**	**84.8**
toy	DiAD	66.2	82.1	50.3
	Transfusion	**81.0**	74.8	56.0
	VLMDiff	68.4	**90.8**	**78.8**
toy brick	DiAD	68.4	93.5	66.4
	Transfusion	65.1	76.2	47.0
	VLMDiff	**72.8**	**96.1**	**85.6**
transistor1	DiAD	73.1	88.6	58.1
	Transfusion	**86.9**	85.0	56.9
	VLMDiff	82.4	**96.7**	**85.5**
u block	DiAD	75.2	88.8	54.2
	Transfusion	78.9	91.0	65.6
	VLMDiff	**79.8**	**98.5**	**90.3**
usb	DiAD	58.9	78.0	28.0
	Transfusion	80.8	87.1	68.3
	VLMDiff	**88.7**	**99.4**	**96.3**
usb adaptor	DiAD	**76.9**	94.0	75.5
	Transfusion	69.9	87.2	57.8
	VLMDiff	71.6	**95.6**	**78.5**
vcpill	DiAD	64.1	90.2	60.8
	Transfusion	72.8	76.1	45.1
	VLMDiff	**83.5**	**97.3**	**85.2**
wooden beads	DiAD	62.1	85.0	45.6
	Transfusion	**79.3**	76.0	53.8
	VLMDiff	73.5	**97.2**	**85.8**
woodstick	DiAD	74.1	90.9	60.7
	Transfusion	**77.5**	91.2	67.5
	VLMDiff	69.2	**95.6**	**79.2**
zipper	DiAD	86.0	90.2	53.5
	Transfusion	**98.3**	87.4	85.4
	VLMDiff	91.8	**97.5**	**87.7**
Avg	DiAD	75.6	88.0	58.1
	Transfusion	**78.6**	84.2	61.6
	VLMDiff	78.0	**97.1**	**87.7**

Table 16. Per class results on Real-IAD dataset for diffusion models, part 2.

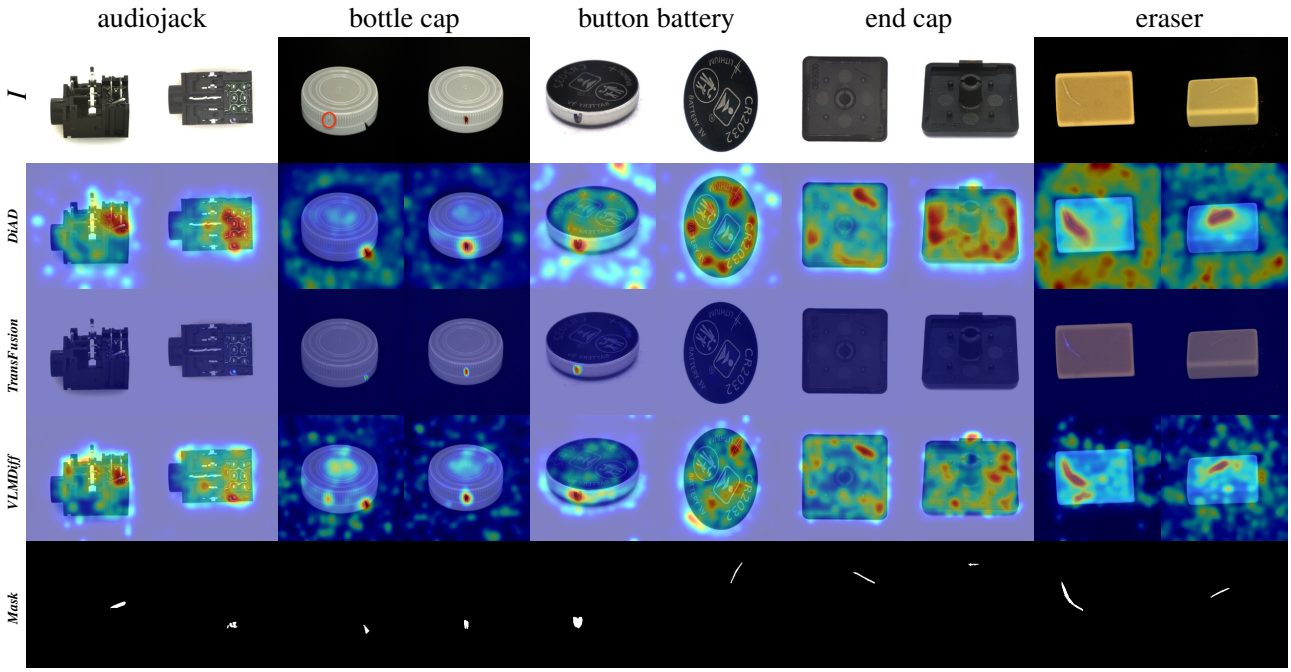

Figure 6. Visual comparison of diffusion-based methods on Real-IAD dataset.

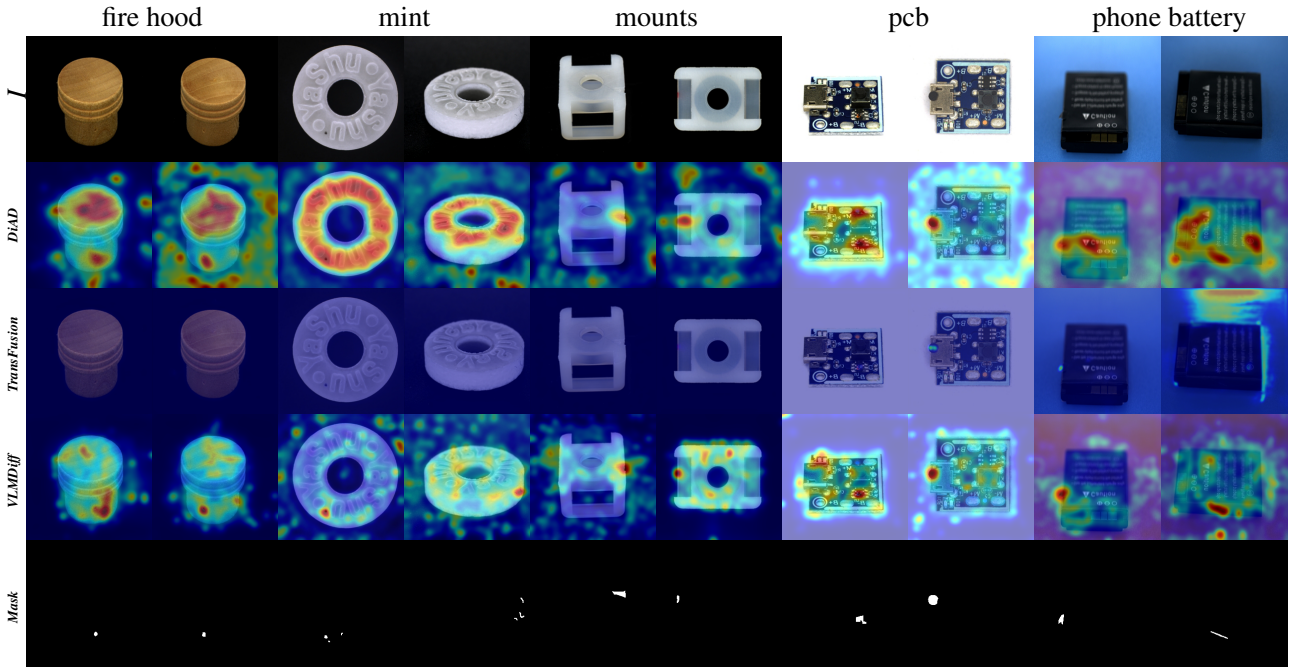

Figure 7. Visual comparison of diffusion-based methods on Real-IAD dataset.

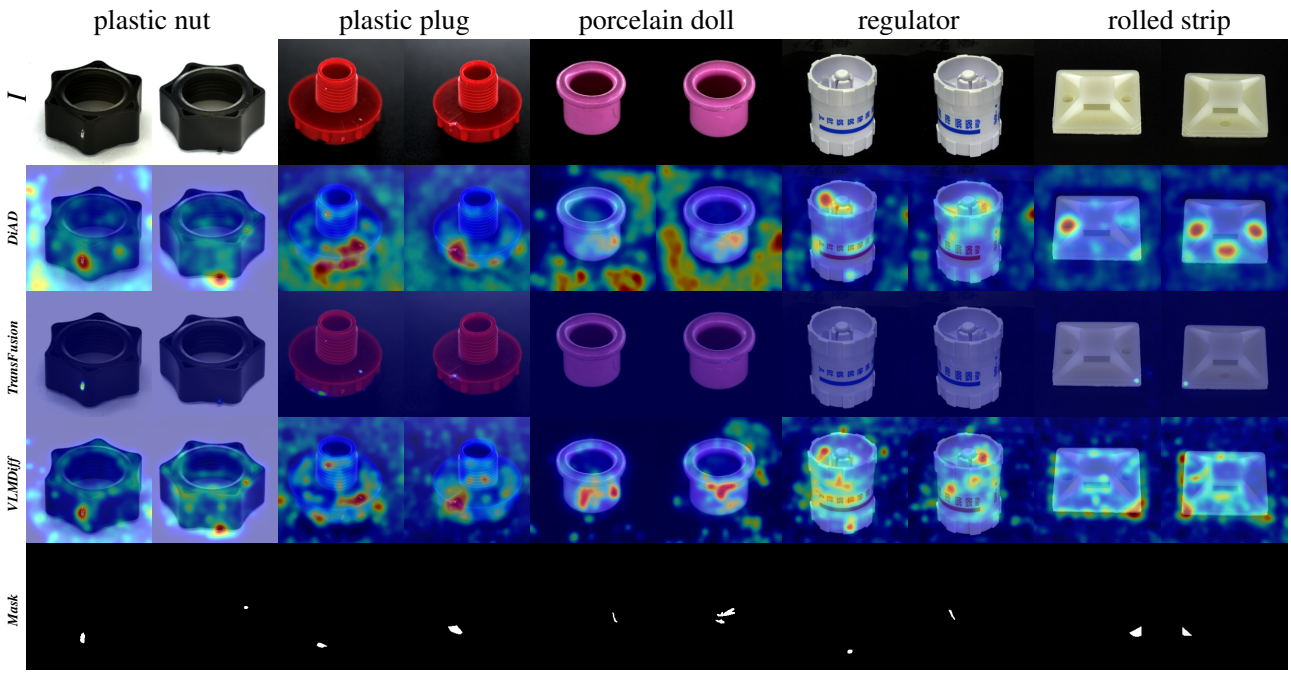

Figure 8. Visual comparison of diffusion-based methods on Real-IAD dataset.

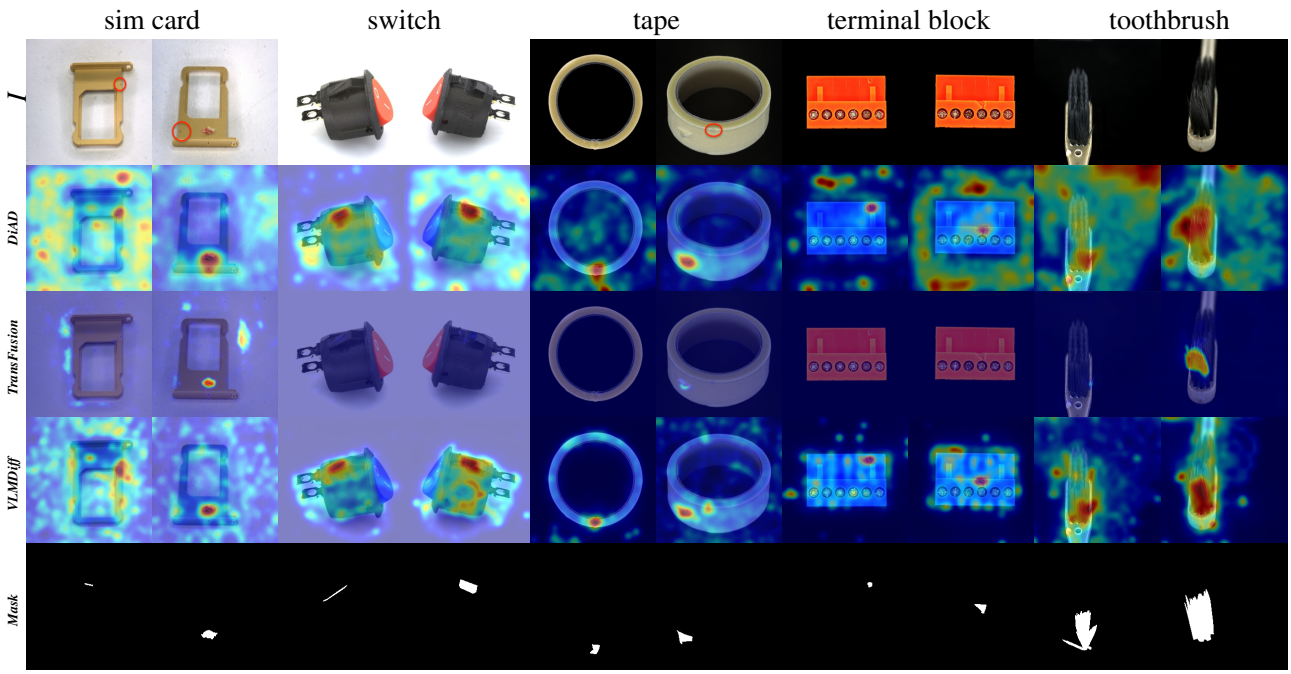

Figure 9. Visual comparison of diffusion-based methods on Real-IAD dataset.

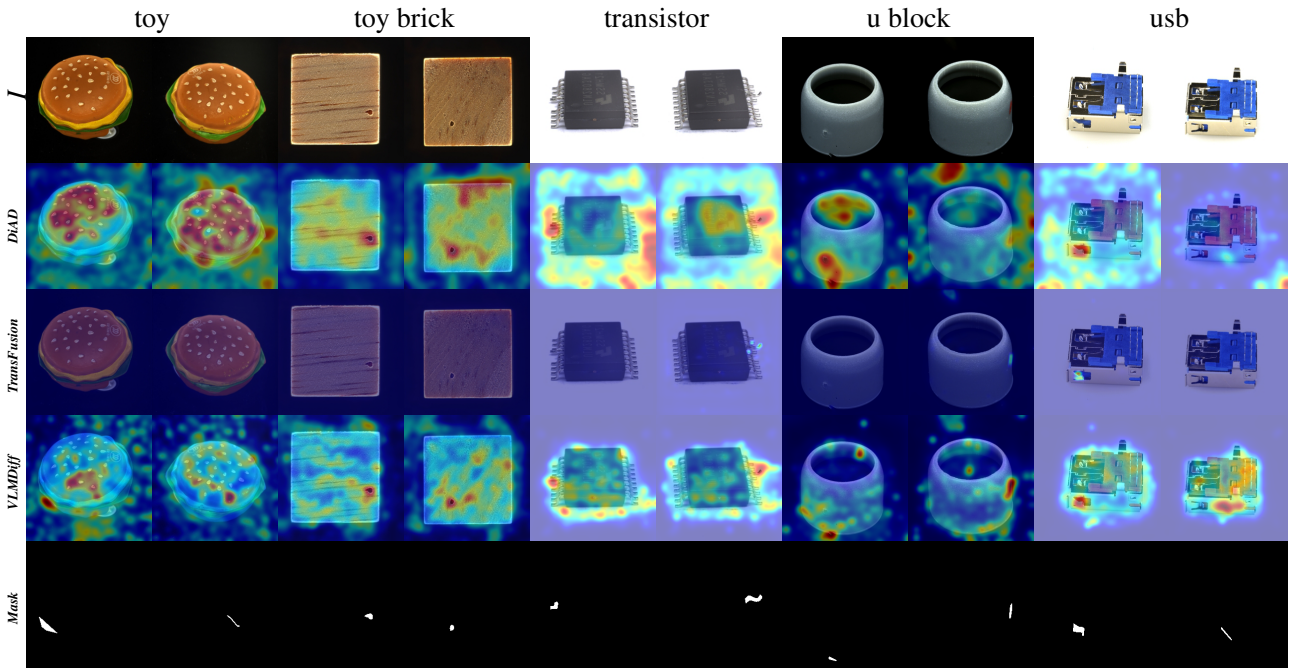

Figure 10. Visual comparison of diffusion-based methods on Real-IAD dataset.

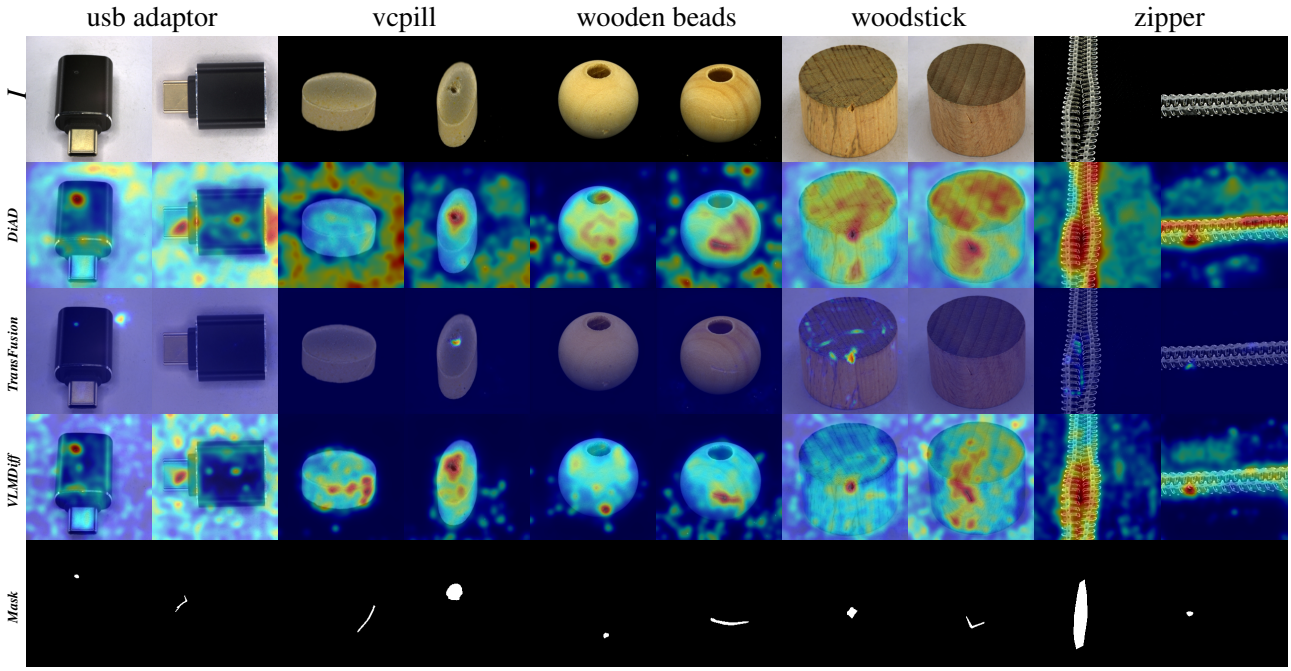

Figure 11. Visual comparison of diffusion-based methods on Real-IAD dataset.

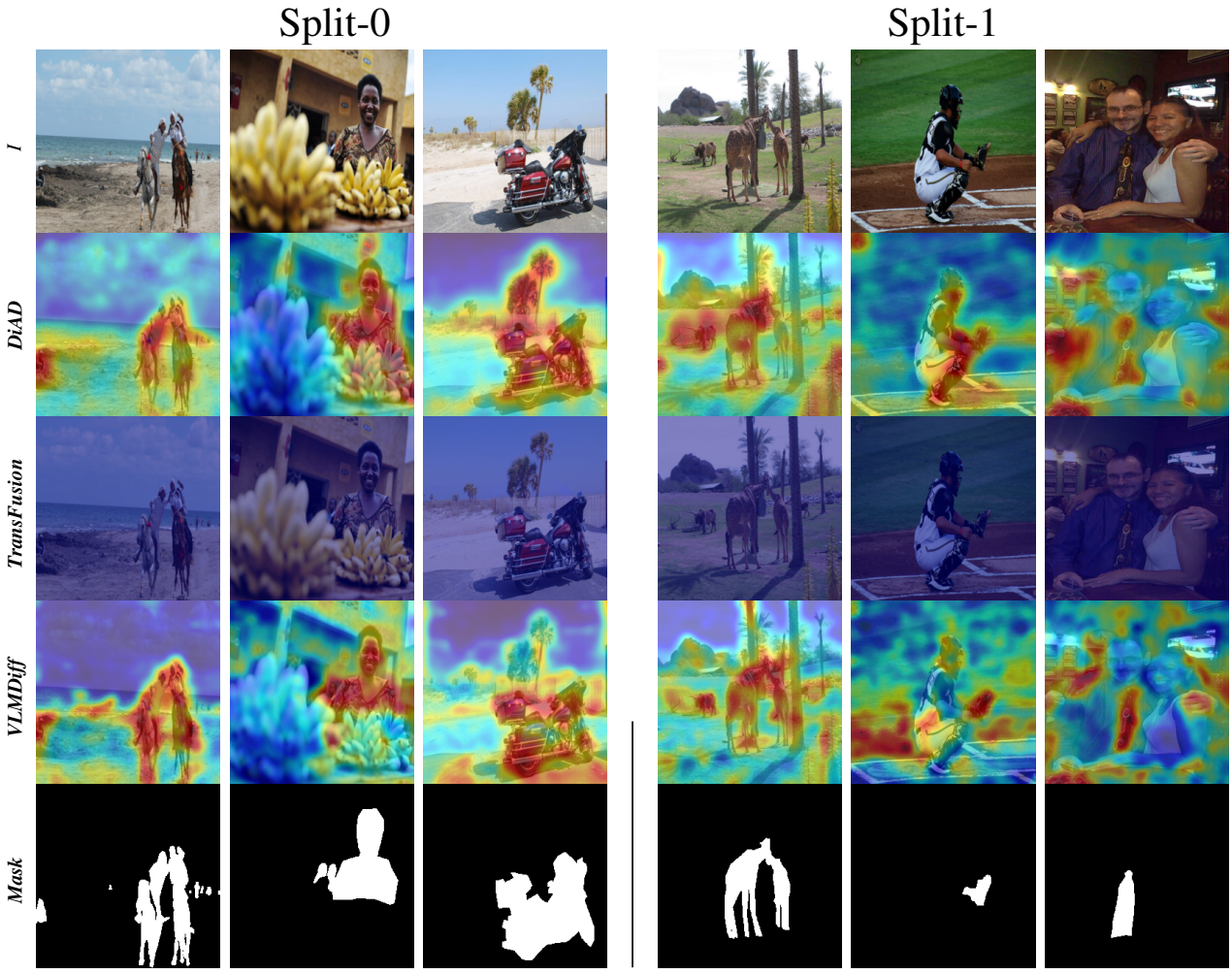

Figure 12. Visual comparison of diffusion-based methods on COCO-AD dataset on Split-0 and Split-1.

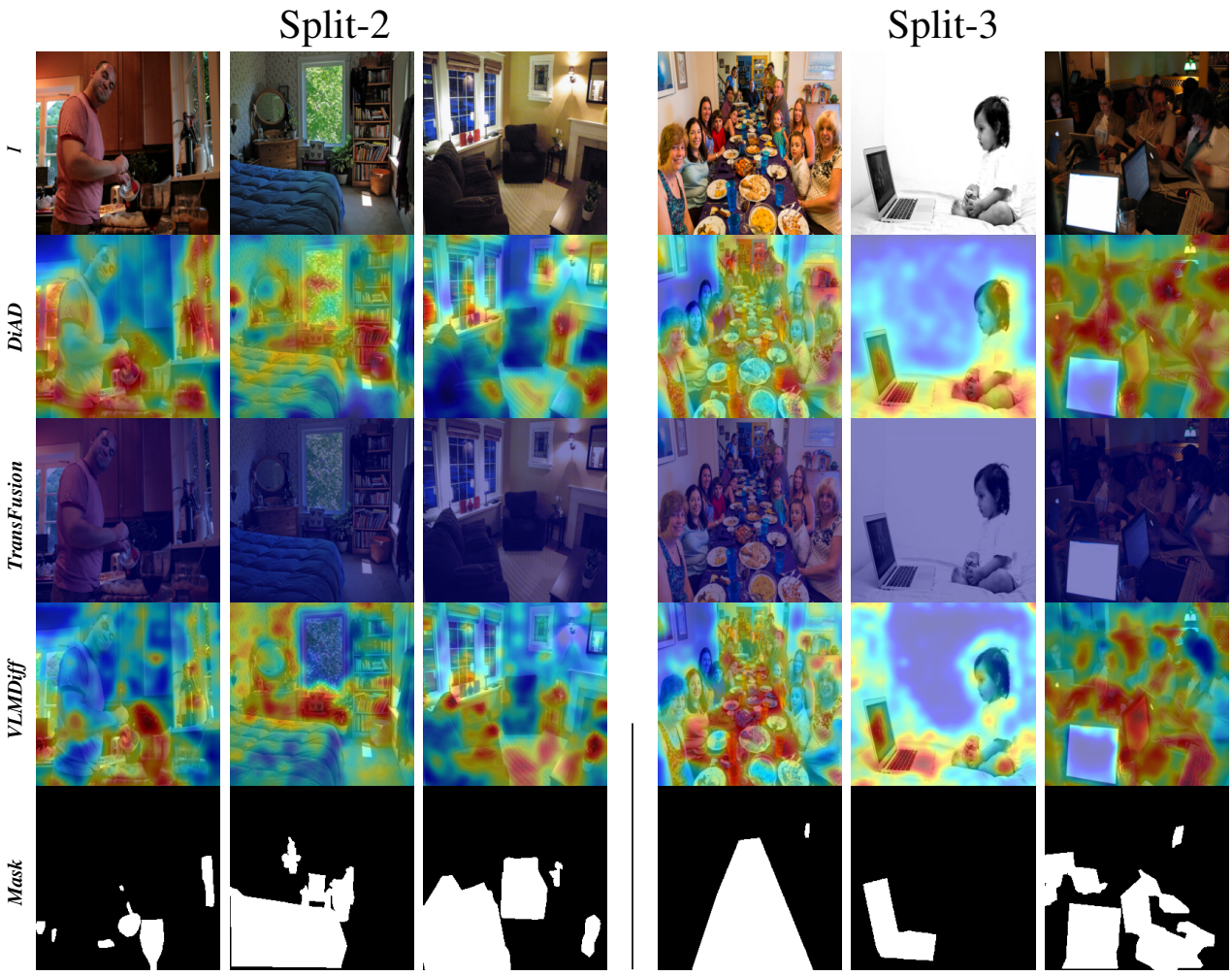

Figure 13. Visual comparison of diffusion-based methods on COCO-AD dataset on Split-2 and Split-3.