

AnomalyPainter: Vision-Language-Diffusion Synergy for Zero-Shot Realistic & Diverse Industrial Anomaly Synthesis

Zhangyu Lai^{1†} Yilin Lu^{1†} Xinyang Li¹ Jianghang Lin¹ Yansong Qu¹
 Liujuan Cao^{1*} Ming Li² Rongrong Ji¹

¹Key Laboratory of Multimedia Trusted Perception and Efficient Computing,
 Ministry of Education of China, Xiamen University ²INSPUR DIGI ENT.

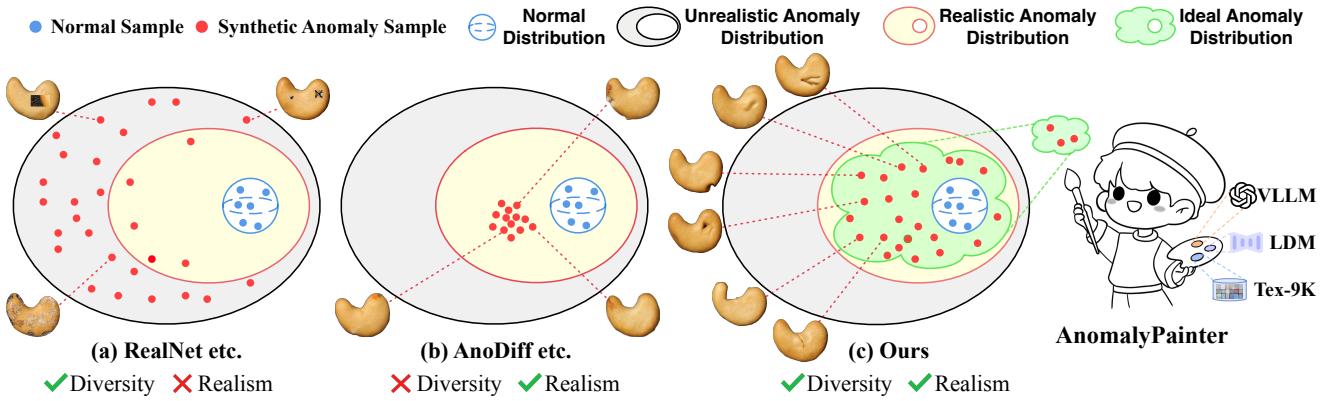


Figure 1. The blue hypersphere [24] represents the normal sample distribution, realistic anomaly distribution should be close to it, while unrealistic anomaly distribution should be farther. Anomaly samples synthesized by different methods exhibit different distributions.

Abstract

While existing anomaly synthesis methods have made remarkable progress, achieving both realism and diversity in synthesis remains a major obstacle. To address this, we propose AnomalyPainter, a zero-shot framework that breaks the diversity-realism trade-off dilemma through synergizing Vision Language Large Model (VLLM), Latent Diffusion Model (LDM), and our newly introduced texture library Tex-9K. Tex-9K is a professional texture library containing 75 categories and 8,792 texture assets crafted for diverse anomaly synthesis. Leveraging VLLM’s general knowledge, reasonable anomaly text descriptions are generated for each industrial object and matched with relevant diverse textures from Tex-9K. These textures then guide the LDM via ControlNet to paint on normal images. Furthermore, we introduce Texture-Aware Latent Init to stabilize the natural-image-trained ControlNet for industrial images. Extensive experiments show that AnomalyPainter outperforms existing methods in realism, diversity, and generalization, achieving superior downstream performance.

1. Introduction

Anomaly detection plays a crucial role in practical applications such as industrial quality control [24] and medical anomaly detection [17]. However, real-world anomaly samples are often exceedingly rare, presenting significant challenges for anomaly detection tasks, including image-level classification and pixel-level segmentation. Consequently, anomaly synthesis tasks [6, 9, 15, 16, 21, 28, 29, 46, 47, 50] gradually emerge as an advanced technique to expand the dataset of available anomaly samples. Depending on whether they employ a limited number of real anomaly samples, existing anomaly synthesis methods can be primarily categorized into zero and few-shot approaches.

Zero-shot methods like CutPaste [21], DRAEM [46], and Realnet [50], synthesize diverse anomalies by cropping and pasting patches from existing anomalies or anomaly texture datasets onto normal samples. However, this random operation often results in unreasonable anomaly content and abrupt transitions at anomaly boundaries, significantly reducing the realism. The distribution of zero-shot synthesized anomaly samples, abstractly represented as red scatter points, is similar to that shown in Figure 1 (a). These red scatter points tend to fall within the unrealistic anomaly distribution, distant from the feature space of normal sam-

*Corresponding Author.

†Equal Contribution.

ples, which are abstractly represented as blue scatter points.

Few-shot methods like DFMGAN [9], AnoGen [12], and AnoDiff [16] use generative models to learn anomaly patterns for generation. However, the samples synthesized by these methods often overfit the few anomaly samples provided during training, failing to cover the diverse types of anomalies that may appear in real-world objects. Similar to Figure 1 (b), the red scatter points representing the few-shot synthesized anomaly samples with similar features, tend to cluster in certain regions in realistic anomaly distribution.

In other words, existing anomaly synthesis methods face the diversity-realism trade-off dilemma. However, we argue that the two are not mutually exclusive. To achieve both diversity and realism, we propose AnomalyPainter, a zero-shot framework. It synergizes the Vision Language Large Model (VLLM) and Latent Diffusion Model (LDM) with our proposed texture library Tex-9K to simulate the formation of real anomalies via texture variations.

Specifically, we construct Tex-9K, a texture library with appropriate texture density, designed for diverse anomaly synthesis. It contains 75 categories and 8,792 professional texture image assets. Leveraging the extensive general knowledge of VLLM, for each industrial object, it generates reasonable and diverse anomaly text descriptions. These descriptions retrieve the most relevant textures from Tex-9K, which serve as anomaly pattern conditions to guide the LDM via ControlNet’s [49] edge-mask control for inpainting normal images. Since ControlNet is trained on natural images, it sometimes performs unstably on industrial images. Texture-Aware Latent Init is then introduced to stabilize ControlNet’s handling, ensuring that normal and texture images blend clearly in the latent space as the initial denoising point, enabling the LDM to achieve precise denoising performance. In short, our zero-shot AnomalyPainter paints reasonable content (via VLLM) with diverse anomaly patterns (via Tex-9K) on normal images while ensuring soft transitions (via LDM) at anomaly boundaries, ultimately synthesizing diverse and realistic anomaly samples that exhibit an ideal distribution, as shown in Figure 1(c).

Our contributions are threefold:

- We propose AnomalyPainter, a zero-shot framework that synergies the general knowledge of VLLM, the high-realism of LDM, and professional assets of Tex-9K, to synthesize realistic and diverse anomaly samples.
- We propose Tex-9K, a professional texture library containing 75 categories and 8,792 texture assets, designed for broadened texture diversity. Texture-Aware Latent Init is proposed to stabilize ControlNet’s edge-mask control, effectively translating the diverse texture assets provided by Tex-9K into realistic and diverse anomaly samples.
- Extensive experiments demonstrate that our synthesized anomaly samples surpass current state-of-the-art (SOTA) synthesis methods and effectively enhance the perfor-

mance of downstream anomaly detection tasks.

2. Related Work

2.1. Anomaly Synthesis

Anomaly synthesis has become an essential technique to support anomaly detection, especially when real anomaly samples are scarce. Zero-shot methods [6, 21, 46] crop and paste patches from existing anomalies or anomaly texture datasets onto normal samples. RealNet [50] further employs a generative model to apply noise to normal images and then denoise them to obtain an anomaly dataset, but ultimately pastes part of the anomaly back using a random mask. While these approaches can produce diverse anomaly samples, the synthesized samples often exhibit significant discrepancies when compared to real-world anomalies. Few-shot methods [9, 12, 15, 16, 28, 47] use generative models [2, 36] to learn anomaly patterns and generate additional anomaly samples. However, they impose strict constraints on data features, causing the synthesized anomalies to overfit the limited training samples.

2.2. Vision Language Large Models

Since the rise of vision-language models (VLMs) [48], CLIP [34] has gained significant attention. WinCLIP [18] first introduces CLIP for assisting anomaly detection, sparking a wave of subsequent advancements [5, 11, 17, 52]. However, relatively little exploration has been done in the area of anomaly synthesis. AnomalyControl [13] leverages VLM as a cross-modal intermediary, incorporating textual or visual information into the image denoising process to enable controllable anomaly synthesis, similar to our approach. However, its approach requires separate training for each type of control, limiting its generalizability. More recently, VLMs have been evolving into vision-language large models (VLLMs) [8, 30, 43]. Leveraging prior world knowledge, VLLMs can perform complex visual question answering with professional responses.

2.3. Latent Diffusion Models

Recent advances in latent diffusion models (LDMs) [31, 32, 35, 36, 39], such as Stable Diffusion [36], have significantly improved image generation. However, pure text-based control struggles to capture complex scene requirements, leading to the development of various diffusion model plugins [10, 14, 22, 38, 45, 49, 51] for more precise control. Among these, ControlNet [49] excels in structural control by integrating additional signals like edge maps, allowing users to specify structures during generation. While many pre-trained ControlNet models are available, they are typically fine-tuned on natural image datasets, leading to instability when applied to industrial objects. Training-free image composition methods [19, 23, 25–27] improve de-

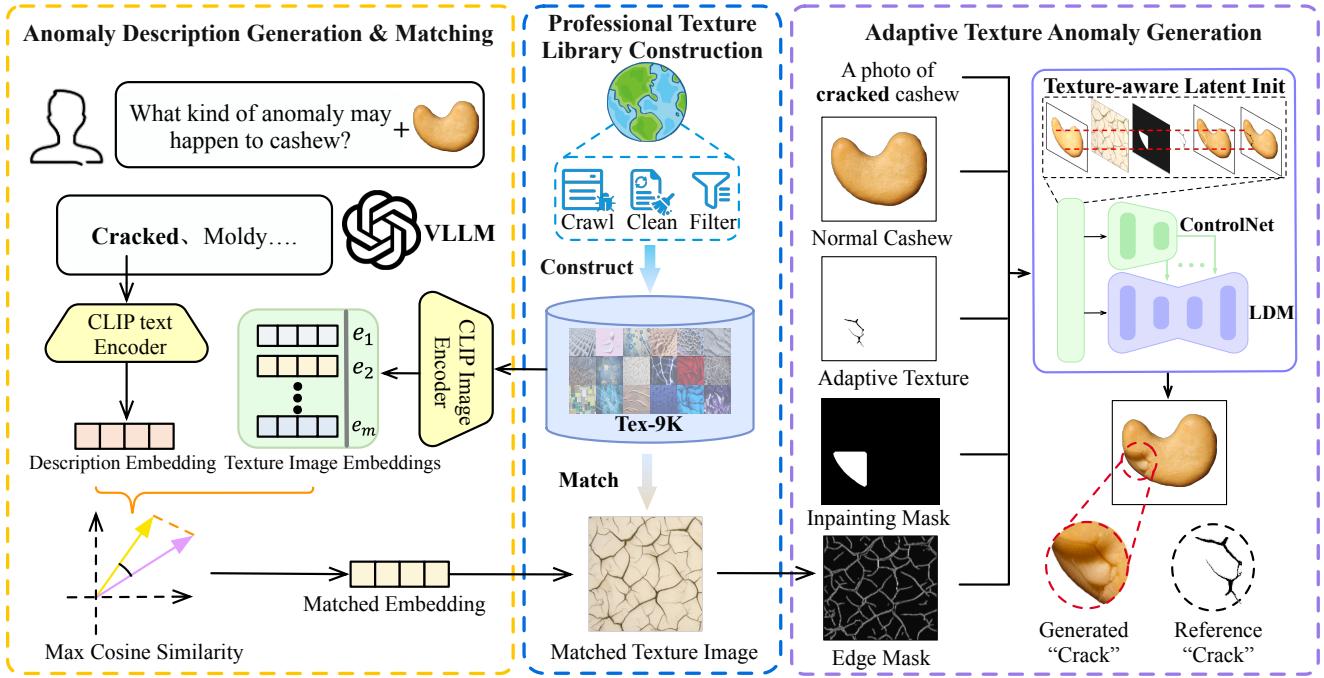


Figure 2. **Overview of AnomalyPainter.** Our framework synthesizes diverse and realistic anomaly samples through three main steps: **Middle: Professional Texture Library Construction** constructs Tex-9K, a texture library with 8,792 texture assets, designed to provide diverse textures crafted for anomaly synthesis. **Left: Anomaly Description Generation and Matching** utilizes VLLM to generate reasonable anomaly descriptions for each industrial object and matches them with relevant textures from Tex-9K using cosine similarity. **Right: Adaptive Texture Anomaly Generation** utilizes Texture-Aware Latent Init to stabilize ControlNet’s edge-mask control for LDM’s high-realism inpainting, ensuring the seamless integration of relevant textures into normal industrial object images.

noising by guiding attention or blending latents for cross-domain synthesis. However, these methods primarily insert whole objects into images and struggle to blend textures seamlessly with industrial objects.

3. Method

Empirically, we consider that the formation of real industrial anomaly samples is usually constrained by the physical properties of objects, which can be understood using the general knowledge of VLLMs. Under various potential random circumstances, texture variations related to the object may manifest in the image. To effectively simulate this and achieve diverse and realistic anomaly synthesis, we propose AnomalyPainter, which is implemented in three key steps: Professional Texture Library Construction, Anomaly Description Generation and Matching, and Adaptive Texture Anomaly Generation. The overview is shown in Figure 2.

3.1. Preliminaries

Latent Diffusion Models consist of two key components: an auto-encoder [20] and a latent denoising network. The autoencoder establishes a bi-directional mapping from the space of the original data to a low-resolution latent space: $z = \mathcal{E}(x), x = \mathcal{D}(z)$, where \mathcal{E} and \mathcal{D} are the encoder

and decoder respectively. The latent denoising network ϵ_θ is trained to denoise noisy latent given a specific timestep t and textual prompt embedding p . The diffusion process adopts the standard formulation DDIM [40], which comprises a forward add-noise diffusion and a backward denoising process. During noise addition, the noisy latent representation at a specified timestep t is obtained as: $z_t = \sqrt{\bar{\alpha}_t} \mathcal{E}(x) + \sqrt{1 - \bar{\alpha}_t} \epsilon$, $\bar{\alpha}_t$ is a monotonically decreasing noise schedule and $\epsilon \sim \mathcal{N}(0, 1)$ is random noise. By continuously denoising the random noise z_T with textual prompt embedding p through predicting noise $\epsilon_\theta(z_t, t, p)$, we can derive a fully denoised latent z_0 . Then, the final clean latent z_0 is passed through the latent decoder \mathcal{D} to generate the high-resolution image $x_0 = \mathcal{D}(z_0)$.

3.2. Professional Texture Library Construction

The motivation for constructing Tex-9K stems from the potential mismatch between our visual intuition for text descriptions and the understanding embedded in VLMs, which are pre-trained on large-scale image-text data from the web. For instance, when searching for images of the description ‘cracked’ online, not all results correspond to the clear crack textures required for fine-grained anomaly synthesis. Similarly, existing texture libraries, such as DTD [7],

often contain overly complex textures that are unsuitable for this purpose. To smoothly align text descriptions with suitable visual concepts for anomaly synthesis and provide more diverse textures, we expanded existing texture datasets by collecting additional texture images on the Internet and defining 75 commonly used texture categories.

The construction of our texture library can be divided into three parts: (a) Crawling: We use a web crawler to collect corresponding images from the Internet legally for each texture category. Data from existing texture datasets are also incorporated into the corresponding categories in our library. (b) Cleaning: The Canny operator [4] is applied to each image to extract the edge mask, and images with excessively dense or sparse textures edge are automatically discarded. (c) Filtering: The remaining data is manually screened to retain images with clear textures and remove potentially harmful or inappropriate content.

Ultimately, Tex-9K which retains 8,792 images, serves as the texture library to provide texture assets crafted for anomaly synthesis. See *Appendix.A* for more details.

3.3. Anomaly Description Generation & Matching

Previous zero-shot anomaly synthesis methods randomly paste patches onto normal images, which fail to capture reasonable anomaly types. To address this, we pioneer the application of VLLM’s general knowledge to enable zero-shot anomaly description simulation, detailed in Alg 1.

Specifically, let $O = \{o_1, o_2, \dots\}$ denote the set of industrial object categories. We first preprocess the entire Tex-9K by encoding all texture images $\mathcal{X} = \{x_{\text{tex},1}, x_{\text{tex},2}, \dots, x_{\text{tex},m}\}$ through the CLIP image encoder. This generates static Texture Images Embeddings $e_{\mathcal{X}} = \{e_1, e_2, \dots, e_m\}$, which are cached for persistent reuse. For each possible industrial object $o_i \in O$, we select a normal image $x_N^{o_i}$ corresponding to the object o_i , then pose a carefully designed question Q_{o_i} for o_i with the prompt template. We will detail the template in *Appendix.B*. By querying VLLMs with Q_{o_i} and $x_N^{o_i}$, we obtain the anomaly description answer $A_{o_i} = \{d_1, d_2, \dots, d_k\}$, where each $d \in A_{o_i}$ is a description. The CLIP text encoder then encodes d into description embedding e_d . After computing cosine similarity between e_d and each $e_i \in e_{\mathcal{X}}$, the max one is taken as matched embedding e_{match} , $e_{\text{match}} \in e_{\mathcal{X}}$. The corresponding $x_{\text{match}} \in \mathcal{X}$ is then used as the matched texture image.

Taking the VisA dataset as an example, let $O = \{\text{candle, cashew, ...}\}$ denote a collection of 12 object types. If $o_i = \text{cashew}$, we select a normal cashew image as $x_N^{o_i}$. A designed question Q_{o_i} can be regarded as “What kind of anomaly may happen to cashew?”. By leveraging the powerful general knowledge, VLLMs (e.g. GPT-4V) may respond $A_{o_i} = \{\text{cracked, moldy...}\}$. Take $d = \text{cracked}$ as an example, the embedding of text “cracked” will be used to match the most similar texture image in Tex-9K.

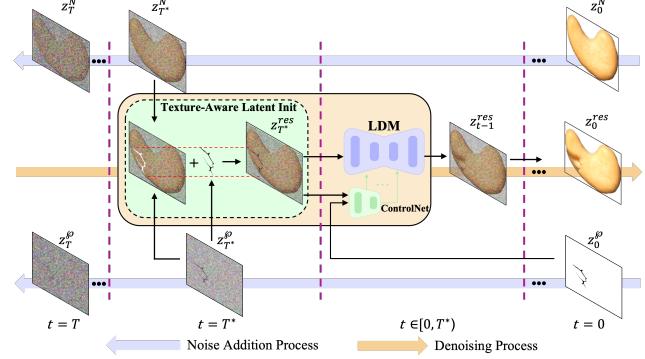


Figure 3. Texture-Aware Latent Initialization (TALI) blends normal image latent z^N and adaptive texture latent z^φ at a later timestep T^* to get $z_{T^*}^{\text{res}}$ as the starting point for better denoising result. For better clarity, the images are shown in the pixel space instead of the latent space.

3.4. Adaptive Texture Anomaly Generation

After obtaining the matched texture image, we propose Adaptive Texture Anomaly Generation, which seamlessly integrates the matched texture into a normal industrial image, creating a realistic anomaly sample, detailed in Alg 2.

Specifically, we first generate an adaptive texture image x_φ based on the matched texture image and the normal industrial object image x_N (dropping the superscript for simplicity), along with an inpainting mask M_{in} that indicates the region where the anomaly content is to be generated. We discuss how to get M_{in} and x_φ later in **Mask Generation**.

To realistically express texture variations in the normal image x_N in region M_{in} , we pioneer the application of ControlNet, which offers powerful edge-mask control ability. Since ControlNet is trained on natural images, it often becomes unstable on industrial data, leading to subtle or inconsistent defect generation. To address this, we introduce Texture-Aware Latent Initialization (TALI), enhancing its reliability in industrial anomaly synthesis (Figure 3).

Texture-aware Latent Init. The core idea of TALI is to blend the normal industrial image x_N with the adaptive texture image x_φ in the latent space as the starting point for denoising, in combination with ControlNet to achieve more stable anomaly generation. In fact, this part can be regarded as an image composition task, which aims to harmonize x_N and x_φ to generate the composite anomaly image x_{res} . Unlike traditional image composition methods, such as TF-ICON [26], which typically invert x_N and x_φ into their corresponding noisy latent representations z_T^N and z_T^φ at a pre-defined timestep T , we choose to begin at a later timestep T^* . This choice helps avoid excessive stylistic disconnection between the generated anomalous part of x_{res} and the normal industrial image x_N , while ensuring anomalous part of x_{res} remains geometrically stable and consistent with x_φ .

Mathematically, we choose $0 < T^* < T$ and use $z_{T^*}^{\text{res}}$ as the starting point for denoising:

$$\mathbf{z}_{T^*}^{\text{res}} = \mathbf{z}_{T^*}^N \odot (\mathbf{1} - \mathbf{M}_\wp^z) + \mathbf{z}_{T^*}^\wp \odot \mathbf{M}_\wp^z, \quad (1)$$

where \mathbf{M}_\wp^z is the segmentation mask of x_\wp in latent space.

After initialization, the pre-trained denoising network ϵ_θ with ControlNet will preserve the layout structure of $z_{T^*}^{\text{res}}$ during $t \in [0, T^*]$, while gradually harmonizing the anomalous texture with the normal image in the inpainting mask M_{in} region. The denoising process is achieved through the removal of the estimated noise $\epsilon_\theta(z_t^{\text{res}}, t, M_{\text{in}}, x_\wp, P)$, where M_{in} defines the region of the original image that can be inpainted, x_\wp serves as the ControlNet condition, P is an object-specific text prompt embedding. The text prompt is formulated as: “A photo of [d] [oi]”, where, for example, d could be “cracked” generated by VLLM, and o_i refers to an industrial category such as “cashew”. When the denoising reaches $t = 0$, the decoder \mathcal{D} is used to decode and obtain $x^{\text{res}} = \mathcal{D}(z_0^{\text{res}})$. We detail T^* selection in Sec 4.4.

Mask Generation. The previous anomaly synthesis methods rely on arbitrary inpainting masks, such as Perlin noise [50] or random masks [16], which often result in anomaly being placed on the background of industrial objects, reducing realism. To address this, we propose the following strategy for generating a more effective inpainting mask: (a) Randomly generate a rectangular mask M_r . (b) Compute the intersection $M_{\text{ov}} = M_r \odot M_u$ between M_r and foreground M_u (object segmentation [33] in normal image). Proceed only if the overlap $\text{Area}(M_r \odot M_u) > \text{thresh}_1$. (c) Compute the intersection between M_{ov} and M_{ca} (Canny edges [4] of matched texture). If the overlap $\text{Area}(M_{\text{ov}} \odot M_{\text{ca}}) > \text{thresh}_2$, we accept M_{ov} as the inpainting mask M_{in} , and use the corresponding texture region as the anomaly texture x_\wp . The final inpainting mask is obtained as $M_{\text{in}} = M_r \odot M_u$, and adaptive texture x_\wp is generated by applying morphological operations to $M_{\text{in}} \odot M_{\text{ca}}$ to ensure connectivity, detailed in Appendix C. This strategy generates a random and diverse mask while ensuring valid texture. We present an example of a generated mask and its corresponding adaptive texture image in Figure 4 (left).

Mask Refine. Since the inpainting mask M_{in} covers a large area, the LDMs’ generation process is relatively unconstrained, often resulting in anomalies appearing only in certain regions. A coarse refinement is then applied by computing the SSIM [44] difference map between the original image x_N and the generated anomaly image x_{res} within the inpainting region M_{in} , yielding a per-pixel similarity score, i.e. $\text{ScoreMap}_{\text{ssim}} = \text{SSIM}(x_N \odot M_{\text{in}}, x_{\text{res}} \odot M_{\text{in}})$. We then compute the mean similarity $\text{thresh}_{\text{mean}} = \text{Average}(\text{ScoreMap}_{\text{ssim}})$ within the region and retain pixels with similarity scores exceeding this threshold, i.e. $M_{\text{res}} = \text{ScoreMap}_{\text{ssim}} > \text{thresh}_{\text{mean}}$. We show an example of generated anomaly image with refined mask in Figure 4 (right).

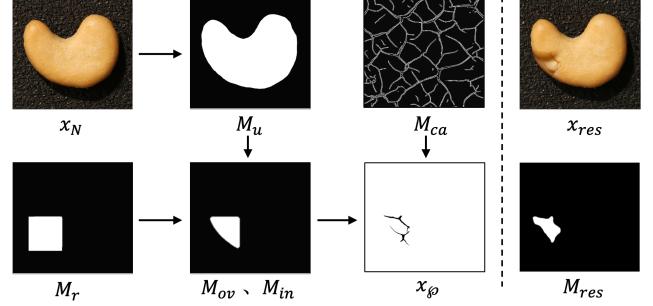


Figure 4. **Left:** An example of a successfully generated inpainting mask M_{in} and its corresponding adaptive texture image x_\wp . **Right:** An example of the generated anomaly result and the refined mask.

Algorithm 1 Anomaly Description Generation & Matching

Data: $O = \{o_1, o_2, \dots\}$, $\mathcal{X} = \{x_{\text{tex}_1}, x_{\text{tex}_2}, \dots, x_{\text{tex}_m}\}$
Cache: $e_{\mathcal{X}} = \{e_1, e_2, \dots, e_m\} \leftarrow \text{CLIP}_{\text{image}}(\mathcal{X})$
Input: $o_i \in O$, a normal image $x_N^{o_i}$
Output: Matched descriptions and texture images for o_i

- 1: Pose question $Q_{o_i} = \text{Template}(o_i)$
- 2: Obtain anomaly descriptions $A_{o_i} = \text{VLLM}(Q_{o_i}, x_N^{o_i})$
- 3: **for** description d in $A_{o_i} = \{d_1, d_2, \dots, d_k\}$ **do**
- 4: Obtain description embedding $e_d \leftarrow \text{CLIP}_{\text{text}}(d)$
- 5: Obtain matched embedding

$$e_{\text{match}} \leftarrow \arg \max_{e_i \in e_{\mathcal{X}}} \cos\langle e_d, e_i \rangle$$
- 6: Select the corresponding texture image $x_{\text{match}} \in \mathcal{X}$
- 7: **end for**
- 8: **return** All descriptions and matched texture images

Algorithm 2 Adaptive Texture Anomaly Generation

Input: x_N , x_{match} , T^* , P .
Cache: $M_u \leftarrow \text{Seg}(x_N)$, $M_{\text{ca}} \leftarrow \text{Canny}(x_{\text{match}})$
Output: Generated anomaly image x_{res} and mask M_{res} .

- 1: $M_{\text{in}}, x_\wp = \text{Mask Generation}(M_u, M_{\text{ca}})$
- 2: **TALI:**
- 3: $z_N, z_\wp = \mathcal{E}(x_N), \mathcal{E}(x_\wp)$
- 4: Sample Noise $\epsilon \sim \mathcal{N}(0, 1)$
- 5: Add Noise $z_{T^*}^N \leftarrow \sqrt{\bar{\alpha}_{T^*}} z_N + \sqrt{1 - \bar{\alpha}_{T^*}} \epsilon$
- 6: Add Noise $z_{T^*}^\wp \leftarrow \sqrt{\bar{\alpha}_{T^*}} z_\wp + \sqrt{1 - \bar{\alpha}_{T^*}} \epsilon$
- 7: Blend latents $z_{T^*}^{\text{res}} \leftarrow z_{T^*}^N \odot (1 - M_\wp^z) + z_{T^*}^\wp \odot M_\wp^z$
- 8: **Denoise with ControlNet:**
- 9: **for** $t \leftarrow T^*$ **downto** 1 **do**
- 10: $z_{t-1}^{\text{res}} \leftarrow \text{DDIM}(z_t^{\text{res}}, \epsilon_\theta(z_t^{\text{res}}, t, M_{\text{in}}, x_\wp, P))$.
- 11: **end for**
- 12: $x_{\text{res}} \leftarrow \mathcal{D}(z_0^{\text{res}})$.
- 13: $M_{\text{res}} = \text{Mask Refine}(x_N, x_{\text{res}}, M_{\text{in}})$
- 14: **return** $x_{\text{res}}, M_{\text{res}}$.

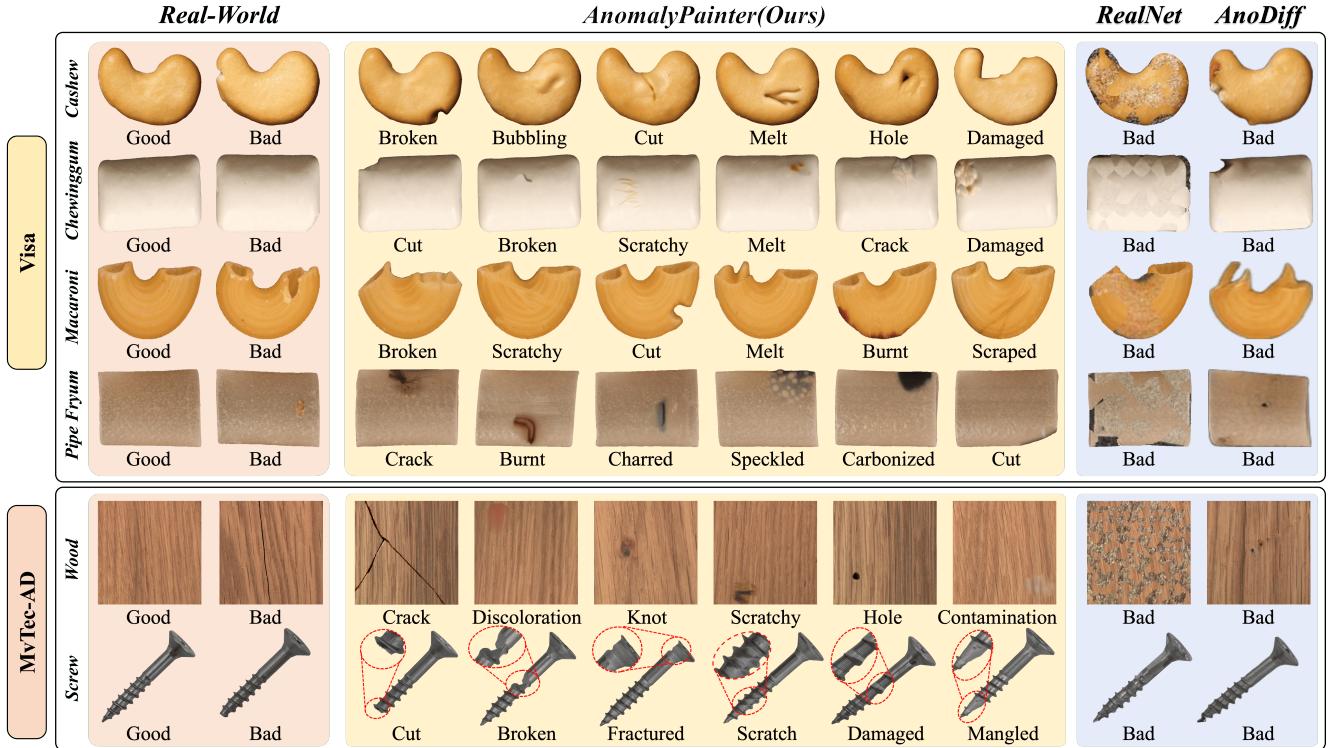


Figure 5. Qualitative Comparison. It is clear that our method can generate diverse and realistic anomaly data across various industrial objects in multiple datasets. It outperforms both the best few-shot method, AnoDiff, and the best zero-shot method, RealNet.

4. Experiments

4.1. Experimental Settings

Dataset. We conduct experiments on the MVTec AD [3] and VisA [53] datasets. MVTec AD consists of 15 categories with 5,354 images, suitable for single-object/textured anomaly inspection. VisA contains 12 categories with 10,821 images, featuring complex objects and multi-object categories. Unlike the fine-grained classification of MVTec AD, VisA employs a weak-category prior evaluation paradigm, which is more aligned with the needs of detecting unknown anomalies in industrial settings and better reflects anomaly synthesis generalizability and robustness. Experiments are primarily evaluated on VisA, with additional results on other datasets provided in Appendix.D.

Evaluation Metrics. For anomaly synthesis, we use Inception Score (IS) and intra-cluster pairwise LPIPS distance (IL) to assess synthesis quality and diversity. For anomaly inspection, we employ pixel-level and image-level AUROC as evaluation metrics.

Implementation Details. AnomalyPainter is implemented using the HuggingFace Diffusers library [42], built on the Stable Diffusion XL 1.0 (SDXL) [32] model and ControlNet-Canny [1], with a CLIP model utilizing a ViT-B/32 backbone. We adopt GPT-4V as our VLLM. Follow-

ing [26], we use a 20-step DDIM sampler, while starting denoising at $T^* = 16$. Similar to [16, 46], we generate 500 images per anomaly category and train a U-Net [37] model for downstream anomaly inspection tasks. As our method is training-free, each anomaly image synthesis takes only 6 seconds on a single Nvidia GeForce RTX 3090 GPU.

Baselines. We select a variety of high-performing open-sourced methods as baselines for downstream tasks, including zero-shot methods that include Cut-Paste [21], DRAEM [46], and RealNet [50], and few-shot methods that include DFMGAN [9], AnoGen [12], and AnoDiff [16]. For anomaly synthesis comparison, we choose the best-performing zero-shot and few-shot methods, RealNet and AnoDiff, for visualization and quality comparison.

4.2. Comparison in Anomaly Generation

Qualitative Comparison. We present anomaly images synthesized by different methods on the VisA and MVTec datasets, as shown in Fig 5. Although RealNet can generate anomaly images with noticeable defects, the results often appear visually unrealistic and confusing. AnoDiff struggles with effective anomaly generation, particularly on challenging datasets like VisA, where it maps different anomaly features into the same embedding, leading to feature confusion, lack of diversity, and uniformity. Further-

Category	Few-Shot			Zero-Shot			
	DFMGAN	AnoGen	AnoDiff	CutPaste	DRAEM	RealNet	Ours
<i>candle</i>	63.9/81.2	83.7/88.6	87.9/91.2	81.9/89.9	83.1/91.6	88.9/93.8	98.7/98.4
<i>capsules</i>	69.1/69.4	76.7/75.6	79.6/80.1	91.8/90.8	92.5/92.5	95.2/92.3	93.3/ 94.6
<i>cashew</i>	93.2/94.7	95.1/95.3	95.2/95.9	97.2/94.7	98.2/96.3	98.6/96.3	98.9/98.4
<i>chewinggum</i>	95.3/99.1	96.4/98.8	97.3/98.9	95.5/96.9	96.6/98.0	97.8/98.9	99.1/99.5
<i>fryum</i>	91.9/96.1	93.5/96.0	93.1/95.6	79.9/89.3	80.2/89.2	86.1/92.5	90.7/95.0
<i>macaroni1</i>	90.2/95.5	93.7/95.6	94.4/95.2	84.7/90.3	82.3/90.0	89.8/93.1	93.4/ 98.5
<i>macaroni2</i>	49.8/91.8	73.7/89.7	68.5/91.4	86.3/82.8	84.0/81.9	91.7/93.8	87.8/ 96.6
<i>pcb1</i>	93.8/96.3	94.3/94.7	95.8/96.3	92.9/92.3	94.5/93.5	93.9/94.2	93.3/95.4
<i>pcb2</i>	83.9/94.8	85.8/89.8	84.5/90.8	81.7/88.1	81.3/88.5	85.5/89.1	88.9/93.9
<i>pcb3</i>	80.6/83.5	81.6/83.1	84.3/85.1	57.6/78.3	69.4/78.7	71.3/83.2	89.2/92.5
<i>pcb4</i>	94.3/91.5	94.4/90.6	96.2/92.2	84.3/83.1	86.7/87.0	89.7/89.0	95.3/ 93.5
<i>pipe fryum</i>	82.3/95.9	91.2/96.7	93.9/97.4	82.3/97.5	84.4/98.2	94.0/97.3	98.7/98.6
<i>Average</i>	82.4/90.8	88.3/91.2	89.2/92.5	84.7/89.5	86.1/90.4	90.2/92.8	93.9/96.2

Table 1. Comparison of AUC-I/AUC-P across image-level and pixel-level anomaly localization on VisA dataset by training a U-Net on the synthesized data from DFMGAN, AnoGen, AnoDiff, CutPaste, DRAEM, RealNet, and our AnomalyPainter.

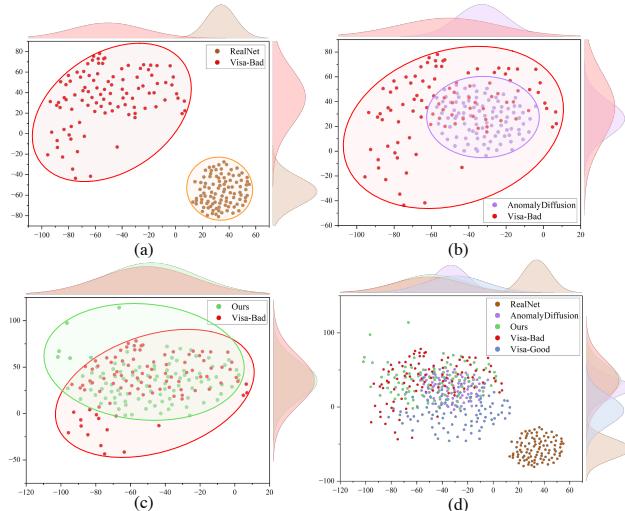


Figure 6. Marginal Group Plot of *cashew* (object in VisA) t-SNE results for anomaly samples synthesized by three methods, real anomaly samples (VisA-Bad) from test set, and normal samples (VisA-Good). See Appendix E for more examples and details.

more, due to architectural constraints, its generated images are blurry (256×256 resolution), with unnatural transitions between anomalies and objects. In contrast, our AnomalyPainter produces anomaly images with a resolution of up to 1024×1024, preserving structural details with natural transitions, higher realism, and greater diversity.

Quantitative Comparison. On the VisA dataset, we synthesize 500 anomaly images per object category and compute IS and IL metrics (Table 2). The results demonstrate that AnomalyPainter outperforms other methods in both diversity and realism.

Data Distribution Comparison. We use t-SNE [41] to visualize anomaly samples synthesized by three methods (RealNet, AnoDiff, and ours), along with real anomalies

Category	RealNet		AnoDiff		Ours	
	IS	IL	IS	IL	IS	IL
<i>candle</i>	1.33	0.27	1.33	0.18	1.65	0.29
<i>capsules</i>	1.65	0.44	1.32	0.33	1.67	0.39
<i>cashew</i>	1.65	0.31	1.29	0.27	1.83	0.36
<i>chewinggum</i>	1.69	0.37	1.27	0.36	1.77	0.43
<i>fryum</i>	1.35	0.22	1.13	0.18	1.69	0.28
<i>macaroni1</i>	1.75	0.22	1.50	0.21	1.73	0.21
<i>macaroni2</i>	1.78	0.32	1.61	0.24	1.92	0.41
<i>pcb1</i>	1.47	0.33	1.22	0.35	1.49	0.34
<i>pcb2</i>	1.37	0.33	1.56	0.30	1.53	0.31
<i>pcb3</i>	1.26	0.19	1.21	0.23	1.51	0.26
<i>pcb4</i>	1.35	0.30	1.27	0.28	1.50	0.27
<i>pipe fryum</i>	1.53	0.22	1.34	0.22	1.69	0.36
<i>Average</i>	1.52	0.29	1.34	0.26	1.67	0.33

Table 2. Quantitative comparison on IS and IL on VisA.

(VisA-Bad) and normal samples (VisA-Good) from the test set (Fig. 6). The overall results in Fig. 6 (d) show that the distribution of anomalies synthesized by AnomalyPainter closely aligns with VisA-Bad, while AnoDiff’s anomalies are restricted to a limited region within VisA-Bad, and RealNet’s anomalies deviate further from VisA-Bad. Additionally, VisA-Bad samples are located near VisA-Good. To provide a clearer comparison of the distribution of synthesized anomalies relative to VisA-Bad, we present the distribution of each method separately in Fig. 6 (a–c). The marginal distributions along the axes and the overlap of distribution ellipses in Fig. 6 further validate the theoretical assumptions proposed in Fig 1.

4.3. Comparison in Downstream Tasks

Anomaly Synthesis for Detection and Localization. We compare our method with existing anomaly generation approaches to evaluate its effectiveness in downstream anomaly detection and localization tasks. For each ob-

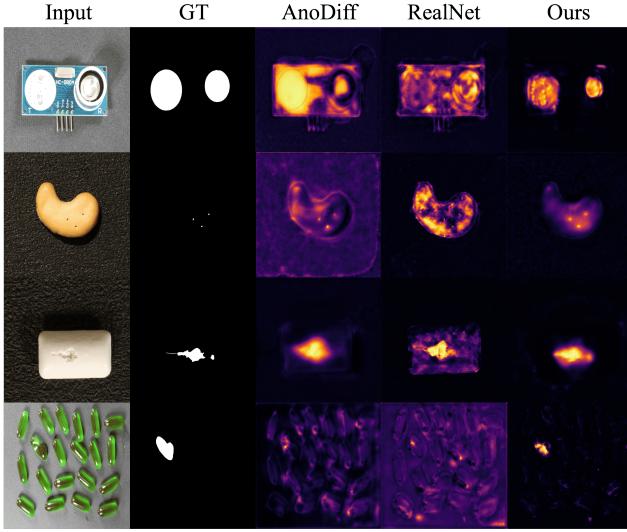


Figure 7. Quantitative anomaly localization comparison with a U-Net trained on the data synthesized by three methods. It shows that ours achieves the best anomaly localization results.

ADGM	TALI	ControlNet	IS	IL	AUC-I	AUC-P
✓	-	-	1.58	0.25	90.5	92.7
✓	✓	-	1.62	0.29	92.7	94.2
✓	-	✓	1.63	0.28	92.3	94.7
-	✓	✓	1.53	0.26	92.1	93.5
✓	✓	✓	1.67	0.33	93.9	96.2

Table 3. Ablation in VisA on our Anomaly Description Generation and Matching (ADGM), Texture-Aware Latent Initialization (TALI) and ControlNet in Adaptive Anomaly Texture Generation. The results indicate that omitting any of the proposed components leads to a noticeable decline in performance.

ject category, we compute image-level and pixel-level AUROC, AP, and F1-Max scores (due to table width limitations, only AUROC is presented, while AP and F1-Max results are provided in *Appendix D*). As shown in Table 1, the U-Net [37] trained on data generated by our method achieves the highest AUC-I (**93.9%**) and AUC-P (**96.2%**) on the VisA dataset, outperforming the SOTA zero-shot method RealNet by **3.7%** and **3.4%**, and the SOTA few-shot method AnoDiff by **4.7%** and **3.7%**, respectively. Additionally, Figure 7 presents qualitative comparisons of anomaly localization, further highlighting our approach.

4.4. Ablation Study

We evaluate the effectiveness of our components: Anomaly Description Generation and Matching (ADGM), Texture-Aware Latent Initialization (TALI) and ControlNet in Adaptive Anomaly Texture Generation. We design 5 different combinations to demonstrate the effectiveness of each component in Table. 3. First, with only ADGM module, our method degrades to a crop-and-paste method, resulting in a

Choice	$T^*=20$	$T^*=18$	$T^*=16$	$T^*=14$	$T^*=12$
IS/IL	1.64 / 0.30	1.69 / 0.32	1.67 / 0.33	1.65 / 0.31	1.59 / 0.32
AUC-I/AUC-P	92.8 / 94.9	93.4 / 95.4	93.9 / 96.2	93.1 / 95.1	92.9 / 95.1

Table 4. Quantitative Ablation with different T^* .

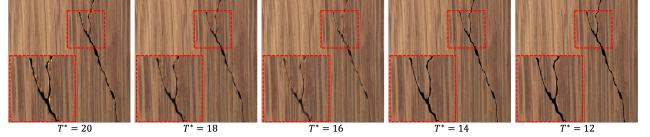


Figure 8. "A photo of cracked wood" results with differet T^* .

significant decrease in realism with $IL = 0.25$. Next, we introduce TALI and ControlNet separately, both of which improve the realism, with IL increasing to 0.29 and 0.28. When TALI and ControlNet are combined, the realism improves further, with IL reaching 0.33, which also performs best in downstream detection tasks. We also run the full TALI and ControlNet experiment without the ADGM module, using random textures as guidance. As expected, random textures often lead to mismatched anomalies, reducing realism with IL decreasing to 0.26.

T^* Selection. Following TF-ICON [26], we adopt the common practice of setting $T = 20$. To find best T^* later than T^* , we experiment with the influence of different choices of T^* and give quantitative results in Table. 4. A typical qualitative visualization on wood is also shown in Figure 8. It is clear that too large step (e.g. $T^* = 20$) may cause the inpainting intensity to be too strong, while too small step (e.g. $T^* = 12$) can result in an overly strong initialization binding, making the anomaly content appear excessively unnatural, thus decreasing the realism.

5. Conclusion

We propose AnomalyPainter, a novel zero-shot anomaly synthesis method that synthesizes diverse and realistic anomaly data. AnomalyPainter combines the ability of VLLMs and LDMs with our proposed Tex-9K to synthesize in a more real-world manner. Tex-9K consists of 75 categories and 8,792 texture images, designed to better support professional assets for the entire anomaly synthesis community. Extensive experiments show that AnomalyPainter outperforms existing anomaly synthesis methods and our synthesized anomaly data effectively improves the performance of the downstream inspection tasks.

Limitations. Similar to previous methods, our method still struggles to synthesize global layout anomalies (e.g., the same object appearing multiple times in an image or objects swapping positions).

Broader Impact. This study pioneers a universal open-world anomaly synthesis approach, enabling unknown anomalies synthesis without prior industrial anomaly samples. This breakthrough lays a crucial foundation for both industry and academia, paving the way for more adaptive and scalable anomaly detection systems.

References

- [1] Controlnet canny sdxl 1.0. <https://huggingface.co/diffusers/controlnet-canny-sdXL-1.0.6>
- [2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4432–4441, 2019. 2
- [3] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtac ad—a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9592–9600, 2019. 6, 1
- [4] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986. 4, 5, 1
- [5] 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 *European Conference on Computer Vision*, pages 55–72. Springer, 2024. 2
- [6] Ruitao Chen, Guoyang Xie, Jiaqi Liu, Jinbao Wang, Ziqi Luo, Jinfan Wang, and Feng Zheng. Easynet: An easy network for 3d industrial anomaly detection. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7038–7046, 2023. 1, 2
- [7] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014. 3, 1
- [8] Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Songyang Zhang, Haodong Duan, Wenwei Zhang, Yining Li, et al. Internlm-xcomposer2-4khd: A pioneering large vision-language model handling resolutions from 336 pixels to 4k hd. *Advances in Neural Information Processing Systems*, 37:42566–42592, 2025. 2, 1
- [9] Yuxuan Duan, Yan Hong, Li Niu, and Liqing Zhang. Few-shot defect image generation via defect-aware feature manipulation. In *Proceedings of the AAAI conference on artificial intelligence*, pages 571–578, 2023. 1, 2, 6
- [10] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 2
- [11] Zhaopeng Gu, Bingke Zhu, Guibo Zhu, Yingying Chen, Ming Tang, and Jinqiao Wang. Anomalygt: Detecting industrial anomalies using large vision-language models. In *Proceedings of the AAAI conference on artificial intelligence*, pages 1932–1940, 2024. 2
- [12] Guan Gui, Bin-Bin Gao, Jun Liu, Chengjie Wang, and Yunsheng Wu. Few-shot anomaly-driven generation for anomaly classification and segmentation. In *European Conference on Computer Vision*, pages 210–226. Springer, 2024. 2, 6
- [13] Shidan He, Lei Liu, and Shen Zhao. Anomalycontrol: Learning cross-modal semantic features for controllable anomaly synthesis. *arXiv preprint arXiv:2412.06510*, 2024. 2
- [14] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022. 2
- [15] Jie Hu, Yawen Huang, Yilin Lu, Guoyang Xie, Guannan Jiang, Yefeng Zheng, and Zhichao Lu. Anomalyxfusion: Multi-modal anomaly synthesis with diffusion. *arXiv preprint arXiv:2404.19444*, 2024. 1, 2
- [16] Teng Hu, Jiangning Zhang, Ran Yi, Yuzhen Du, Xu Chen, Liang Liu, Yabiao Wang, and Chengjie Wang. Anomalydiffusion: Few-shot anomaly image generation with diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8526–8534, 2024. 1, 2, 5, 6
- [17] Chaoqin Huang, Aofan Jiang, Jinghao Feng, Ya Zhang, Xinchao Wang, and Yanfeng Wang. Adapting visual-language models for generalizable anomaly detection in medical images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11375–11385, 2024. 1, 2
- [18] Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. Winclip: Zero-/few-shot anomaly classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19606–19616, 2023. 2
- [19] Liyao Jiang, Negar Hassanpour, Mohammad Salameh, Mohammadreza Samadi, Jiao He, Fengyu Sun, and Di Niu. Pixelman: Consistent object editing with diffusion models via pixel manipulation and generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025. 2
- [20] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations (ICLR) 2014, Banff, AB, Canada*, 2014. 3
- [21] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9664–9674, 2021. 1, 2, 6
- [22] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22511–22521, 2023. 2
- [23] Jiaqi Liu, Tao Huang, and Chang Xu. Training-free composite scene generation for layout-to-image synthesis. In *European Conference on Computer Vision*, pages 37–53. Springer, 2024. 2
- [24] Jiaqi Liu, Guoyang Xie, Jinbao Wang, Shangnian Li, Chengjie Wang, Feng Zheng, and Yaochu Jin. Deep industrial image anomaly detection: A survey. *Machine Intelligence Research*, 21(1):104–135, 2024. 1
- [25] Shengyuan Liu, Bo Wang, Ye Ma, Te Yang, Xipeng Cao, Quan Chen, Han Li, Di Dong, and Peng Jiang. Training-free subject-enhanced attention guidance for compositional text-to-image generation. *arXiv preprint arXiv:2405.06948*, 2024. 2
- [26] Shilin Lu, Yanzhu Liu, and Adams Wai-Kin Kong. Tf-icon: Diffusion-based training-free cross-domain image composi-

- tion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2294–2305, 2023. 4, 6, 8
- [27] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. 2
- [28] Shuanlong Niu, Bin Li, Xinggang Wang, and Hui Lin. Defect image sample generation with gan for improving defect recognition. *IEEE Transactions on Automation Science and Engineering*, 17(3):1611–1622, 2020. 1, 2
- [29] Utkarsh Ojha, Yijun Li, Jingwan Lu, Alexei A Efros, Yong Jae Lee, Eli Shechtman, and Richard Zhang. Few-shot image generation via cross-domain correspondence. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10743–10752, 2021. 1
- [30] OpenAI. Gpt-4v(ision) technical work and authors, <https://openai.com/contributions/gpt-4v/>. Technical report, 2, 1
- [31] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 2
- [32] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024. 2, 6
- [33] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar Zaiane, and Martin Jagersand. U2-net: Going deeper with nested u-structure for salient object detection. page 107404, 2020. 5
- [34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2
- [35] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022. 2
- [36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 2
- [37] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5–9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 6, 8
- [38] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 2
- [39] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2
- [40] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3
- [41] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 7, 2
- [42] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. <https://github.com/huggingface/diffusers>, 2022. 6
- [43] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 2
- [44] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5
- [45] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. I_ADAPTER: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 2
- [46] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Draem: a discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8330–8339, 2021. 1, 2, 6
- [47] Gongjie Zhang, Kaiwen Cui, Tzu-Yi Hung, and Shijian Lu. Defect-gan: High-fidelity defect synthesis for automated defect inspection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2524–2534, 2021. 1, 2
- [48] Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2
- [49] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 2
- [50] Ximiao Zhang, Min Xu, and Xiuzhuang Zhou. Realnet: A feature selection network with realistic synthetic anomaly for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16699–16708, 2024. 1, 2, 5, 6

- [51] Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K. Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 2023. [2](#)
- [52] Jiaqi Zhu, Shaofeng Cai, Fang Deng, Beng Chin Ooi, and Junran Wu. Do llms understand visual anomalies? uncovering llm's capabilities in zero-shot anomaly detection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 48–57, 2024. [2](#)
- [53] Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *European Conference on Computer Vision*, pages 392–408. Springer, 2022. [6](#), [1](#)

A. Tex-9K Construction

We construct Tex-9K to provide a diverse and high-quality selection of texture patterns for fine-grained anomaly synthesis. Specifically, for 75 selected anomaly categories, we use web crawlers to collect 200 relevant images per category from Google. Additionally, if suitable images are available in existing datasets (e.g., DTD [7]), we integrate them to enhance coverage. To ensure texture clarity, we apply an automated density filtering process. Using the Canny edge detector [4], we generate edge masks for all images and discard those where edge coverage exceeds 70% or falls below 2% of the total image area. The remaining images undergo careful manual selection, resulting in a curated dataset of 8,792 high-quality texture images. Figure 9 provides examples from Tex-9K.

Furthermore, for each image, we generate descriptive text prompts using the multi-modal large language model InternLM-XComposer [8], enhancing its usability for conditioned anomaly synthesis. See Figure 10 for some examples.

B. VLLM question template

For each industrial object category o_i , we use the template to query the GPT-4V [30] API to obtain the answer:
“You are a professional industrial anomaly engineer. Now I am doing some tasks on industrial defect photo generation. Please carefully analyze the material of the object [o_i] in the image and provide possible defects that could occur on it. I don’t need that much explanation, just give me the brief answer like “cracked, faded”. Please make a specific analysis according to the specific situation.” In fact, you can create a custom prompt based on your own needs to get the description you want.

C. Mask Strategy

Mask Generation

(1) Random Rectangle Mask Generation: We randomly generate a rectangular mask on an $H \times W$ image, where the mask’s length is chosen from $[l_{\text{rate}} \times H, h_{\text{rate}} \times H]$ and the width from $[l_{\text{rate}} \times W, h_{\text{rate}} \times W]$. l_{rate} and h_{rate} are adjustable parameters. This results in a random mask M_r .

(2) Foreground Segmentation and Overlap Filtering: A segmentation model extracts the foreground mask M_u from normal object image x_N . We compute the intersection M_{ov} between M_r and M_u . If the overlap exceeds a threshold thresh_1 , we proceed. Otherwise, we regenerate M_r in (1).

(3) Texture Validity Check: We compute the intersection of M_{ov} and the Canny edge map M_{ca} extracted from the matched texture image. If the valid overlap exceeds a threshold thresh_2 , we accept M_{ov} as the inpainting mask M_{in} and adaptive texture x_ϕ is generated by applying mor-

phological operations to $M_{in} \odot M_{ca}$ to ensure connectivity. If not, we regenerate M_r in (1).

In the specific implementation, l_{rate} and h_{rate} are set to 0.1 and 0.3, respectively. thresh_1 is set to 0.3 and thresh_2 is set to 0.05. The morphological operations first invert the pixel values of $M_{in} \odot M_{ca}$, followed by a single 5×5 kernel dilation and subsequent erosion to get the complete and connected texture.

D. More Comparisons and Downstream Experiments on Visa and Mvtec

More Quantitative Comparison and Analysis. We evaluated various zero-shot and few-shot anomaly synthesis methods, including DFMGAN [9], AnoGen [16], AnoDiff [16], CutPaste [21], DRAEM [46], RealNet [50], and our proposed AnomalyPainter. For each object category in the MVTEC [3] dataset, we synthesized 500 anomaly images and computed the Inception Score (IS) and Inception Loss (IL) for all methods, with detailed results shown in Table 5. The experiments demonstrate that AnomalyPainter significantly outperforms the compared methods in terms of both diversity and realism of generated images. Additionally, supplementary experimental results on the Visa [53] dataset, as presented in Table 6, further validate the superior performance of AnomalyPainter.

Metric	DFMGAN	AnoGen	AnoDiff	Cutpaste	DRAEM	RealNet	Ours
IS	1.72	1.68	1.80	1.52	1.73	1.70	1.91
IL	0.20	0.23	0.32	0.15	0.24	0.22	0.35

Table 5. Quantitative comparison of IS and IL on MVTEC AD.

Metric	DFMGAN	AnoGen	AnoDiff	Cutpaste	DRAEM	RealNet	Ours
IS	1.35	1.28	1.34	1.37	1.44	1.52	1.67
IL	0.19	0.21	0.26	0.17	0.23	0.29	0.33

Table 6. Quantitative comparison of IS and IL on Visa.

Qualitative Comparison on More Complex Categories. It is evident that our method can generate diverse and realistic anomaly data even for more complex categories, including multi-instance capsule, candle, and the more intricate PCB category. In comparison, it outperforms the best few-shot method, AnoDiff, as well as the best zero-shot method, RealNet.

Downstream Task Evaluation: Image-Level Detection and Pixel-Level Localization. To validate the generalization capability of anomaly synthesis methods, we conduct comprehensive evaluations on both the MVTEC AD and Visa datasets. For the Visa dataset, we synthesize 500 anomaly image-mask pairs per object category and train a U-Net model for downstream task evaluation. For the

Method	AUC-I	AP-I	F1-I	AUC-P	AP-P	F1-P
DFMGAN	82.4	81.7	78.8	90.8	29.5	34.0
AnoGEN	88.3	86.8	81.6	91.2	32.9	37.8
AnoDiff	89.2	87.9	82.8	92.5	34.3	38.9
CutPaste	84.7	83.5	79.5	89.5	28.0	33.7
DRAEM	86.1	84.7	81.2	90.4	29.7	34.9
RealNet	90.2	89.3	84.4	92.8	34.9	39.4
Ours	93.9	92.8	88.3	96.2	40.7	45.3

Table 7. Performance comparison of anomaly data synthesized by different methods on VISA downstream detection tasks.

Method	AUC-I	AP-I	F1-I	AUC-P	AP-P	F1-P
DFMGAN	87.2	94.8	94.7	90.0	62.7	62.1
AnoGEN	97.7	98.8	97.4	97.9	75.3	69.0
AnoDiff	99.2	99.7	98.7	99.1	81.4	76.3
CutPaste	75.8	89.9	88.4	91.7	52.9	51.4
DRAEM	94.6	97.0	94.4	92.2	62.7	53.1
RealNet	95.3	97.3	94.7	95.5	67.4	64.0
Ours	97.6	98.3	97.3	98.3	74.9	67.9

Table 8. Performance comparison of anomaly data synthesized by different methods on MVTec AD downstream detection tasks.

MVTec AD dataset, which provides fine-grained category annotations, we further synthesize 500 image-mask pairs for each anomaly type within every object category to verify robustness in complex scenarios. Experimental results (as shown in Table 7 and Figure 8) demonstrate that: (1)On the Visa dataset, our method significantly outperforms existing approaches without requiring explicit category annotations, exhibiting strong adaptability to unseen classes. (2)On the MVTec AD dataset, our approach surpasses all zero-shot baselines and even exceeds several 2023-2024 few-shot SOTA methods (e.g., DFMGAN [9] and AnoGen [12]) in specific metrics. This comparative analysis confirms the stability and generalizability of our method across datasets and tasks, particularly aligning with practical industrial inspection requirements where prior knowledge is typically limited.

E. More t-SNE result in Visa dataset

To comprehensively analyze the distribution of synthesized anomaly data from different methods, we apply t-SNE [41] to 500 anomaly images generated by RealNet, Anomaly-Diffusion, and our method on the Visa dataset. Additionally, we select 500 normal industrial images and 500 real anomaly images (by replicating the 100 available real images for each object to reach 500). In total, we have five

categories of data, with 500 images per category, which are reduced to 5x500 points for visualization. In the main text, the 500 points for each method are further clustered into 100 points using k-means for visual clarity. Here we showcase six other examples in Visa, each fully displaying the situation of 500 points of five categories in Figure 11.

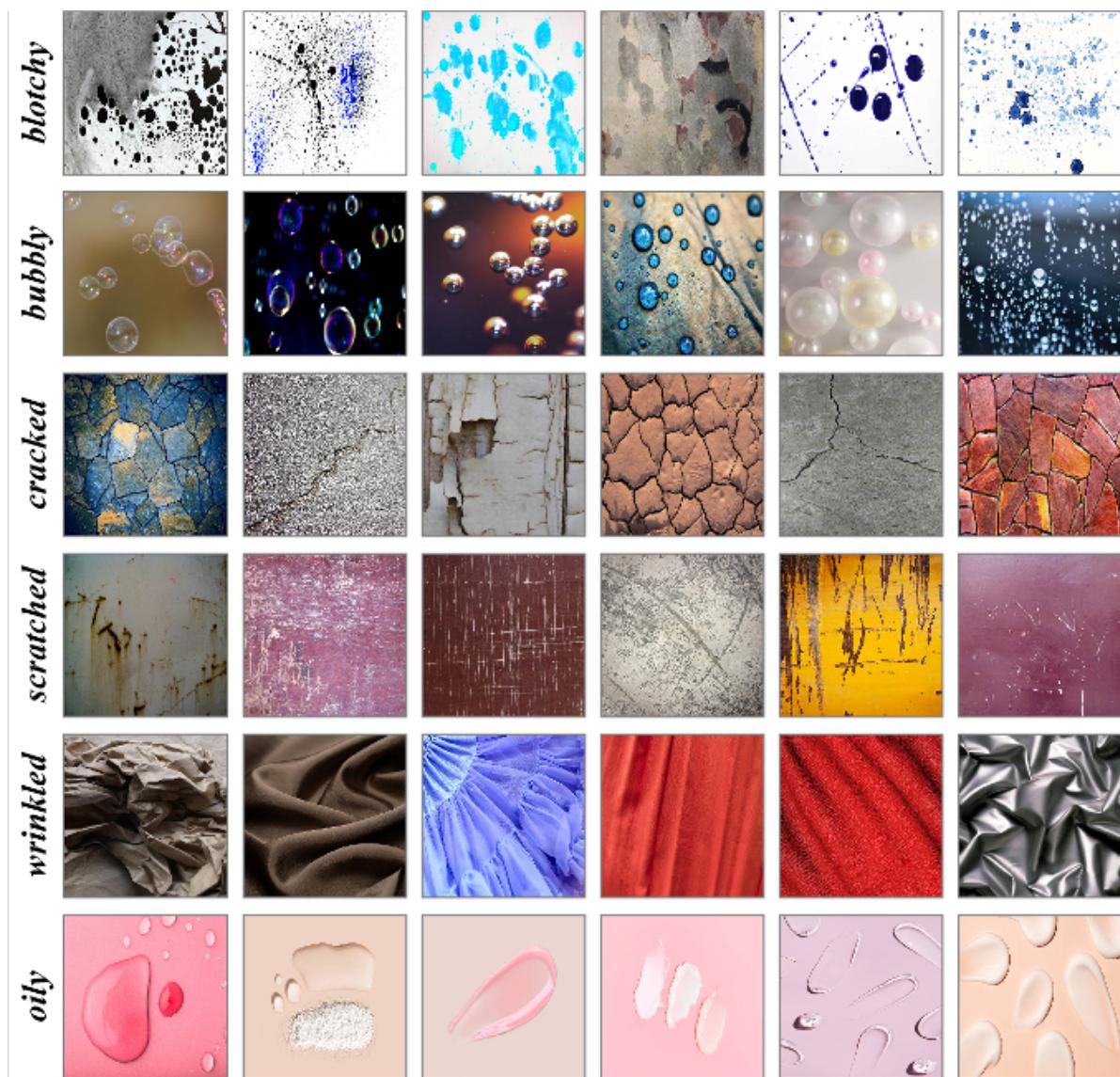


Figure 9. Examples of our proposed Tex-9K.

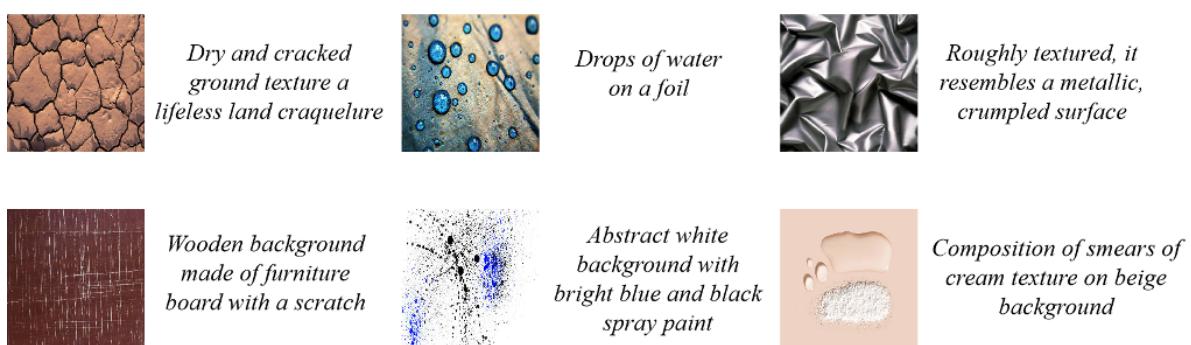


Figure 10. Examples from Tex-9K with corresponding descriptive text prompts

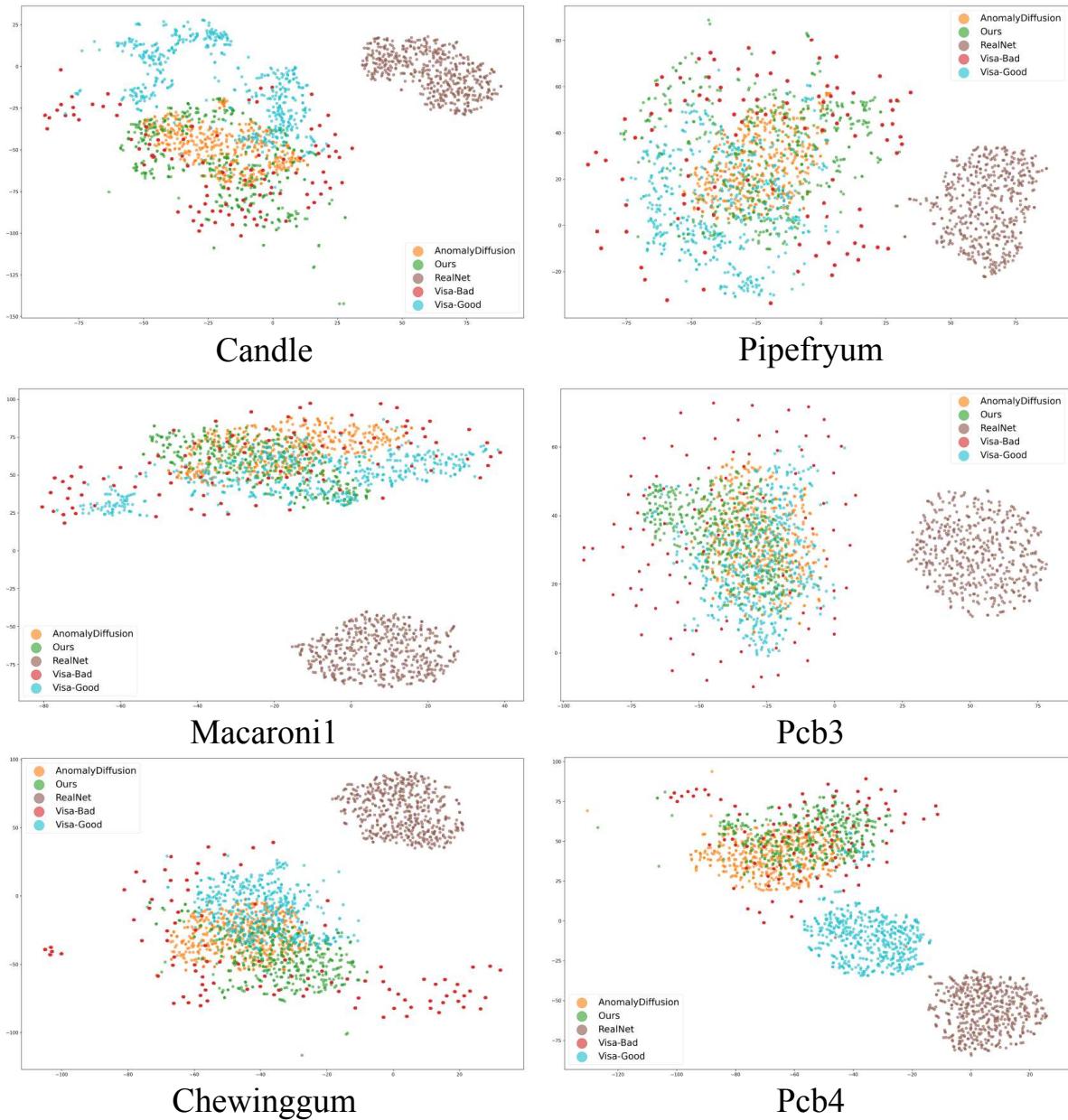


Figure 11. More t-SNE visualization examples of industrial objects in Visa.

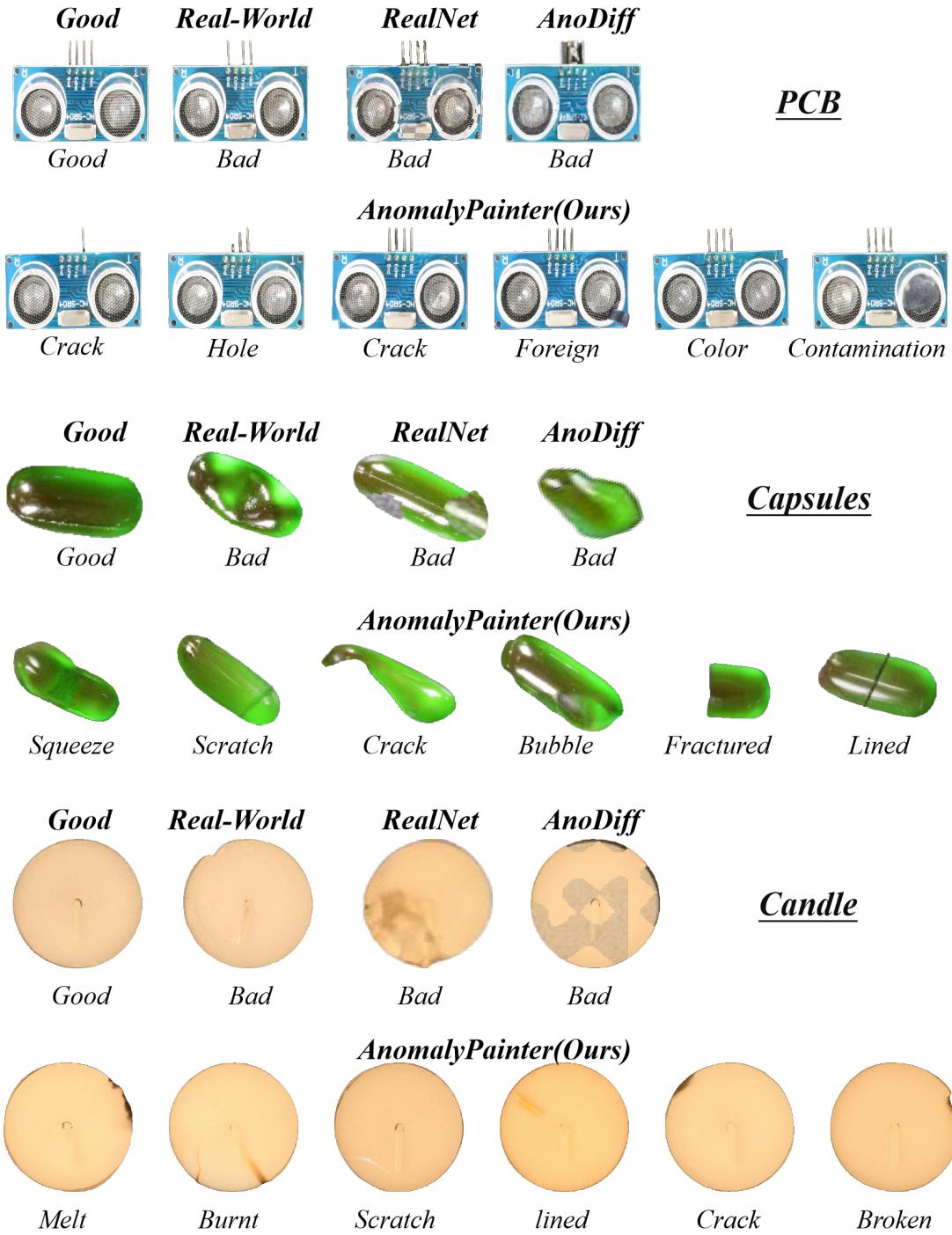


Figure 12. Qualitative comparison on complex categories. Our method generates diverse and realistic anomalies for multi-instance capsule, candle, and complex PCB, outperforming AnoDiff and RealNet.