AnomalyDiffusion: Few-Shot Anomaly Image Generation with Diffusion Model

Teng Hu[1*], **Jiangning Zhang**[2*], **Ran Yi**[1†], **Yuzhen Du**[1], **Xu Chen**[2],
Liang Liu[2], **Yabiao Wang**[2], **Chengjie Wang**[1,2]
[1]**Shanghai Jiao Tong University** [2]**Youtu Lab, Tencent**
{hu-teng, ranyi, Haaaaaaaaaa}@sjtu.edu.cn;
{vtzhang, cxxuchen, leoneliu, caseywang, jasoncjwang}@tencent.com;

Abstract

Anomaly inspection plays an important role in industrial manufacture. Existing anomaly inspection methods are limited in their performance due to insufficient anomaly data. Although anomaly generation methods have been proposed to augment the anomaly data, they either suffer from poor generation authenticity or inaccurate alignment between the generated anomalies and masks. To address the above problems, we propose *AnomalyDiffusion*, a novel diffusion-based few-shot anomaly generation model, which utilizes the strong prior information of latent diffusion model learned from large-scale dataset to enhance the generation authenticity under few-shot training data. Firstly, we propose Spatial Anomaly Embedding, which consists of a learnable anomaly embedding and a spatial embedding encoded from an anomaly mask, disentangling the anomaly information into anomaly appearance and location information. Moreover, to improve the alignment between the generated anomalies and the anomaly masks, we introduce a novel Adaptive Attention Re-weighting Mechanism. Based on the disparities between the generated anomaly image and normal sample, it dynamically guides the model to focus more on the areas with less noticeable generated anomalies, enabling generation of accurately-matched anomalous image-mask pairs. Extensive experiments demonstrate that our model significantly outperforms the state-of-the-art methods in generation authenticity and diversity, and effectively improves the performance of downstream anomaly inspection tasks. The code and data are available in https://github.com/sjtuplayer/anomalydiffusion.

1 Introduction

In recent years, industrial anomaly inspection algorithms, *i.e.,* anomaly detection, localization, and classification, plays a crucial role in industrial manufacture (Duan et al. 2023). However, in real-world industrial production, the anomaly samples are very few, posing a significant challenge for anomaly inspection (Fig. 1-top). To mitigate the issue of few anomaly data, existing anomaly inspection mostly relies on unsupervised learning methods that only use normal samples (Zavrtanik, Kristan, and Skočaj 2021; Li et al. 2021), or few-shot supervised learning methods (Zhang et al. 2023a). Although these methods perform well in anomaly detection, they have limited performance in anomaly localization and cannot handle anomaly classification.

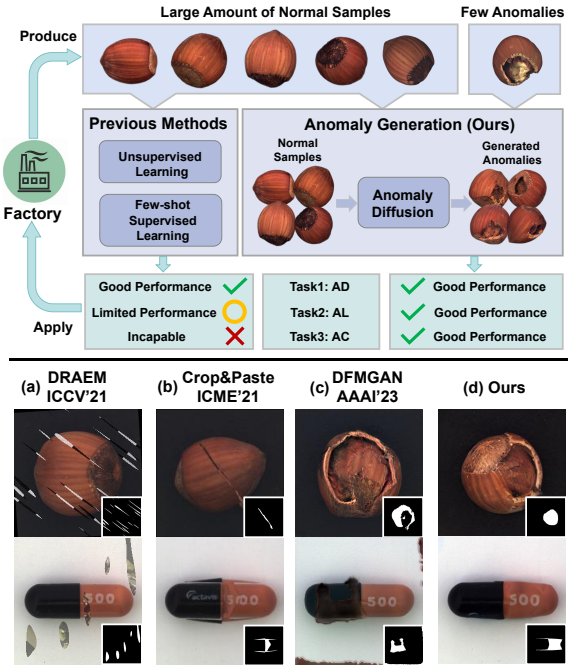

Figure 1: ***Top:*** Our model generates extensive anomaly data, which supports the downstream Anomaly Detection (AD), Localization (AL) and Classification (AC) tasks, while previous methods mainly rely on unsupervised learning or few-shot supervised learning due to the limited anomaly data; ***Bottom:*** Generated anomaly results on hazelnut-crack and capsule-squeeze of our model and existing anomaly generation methods, where our results are the most authentic.

To cope with the problem of scarce anomaly samples, researchers propose anomaly generation methods to supplement the anomaly data, which can be divided into two types: ***1)*** *The model-free methods* randomly crop and paste patches from existing anomalies or anomaly texture dataset onto normal samples (Li et al. 2021; Lin et al. 2021; Zavrtanik, Kristan, and Skočaj 2021). But such methods exhibit poor authenticity in the synthesized data (Fig. 1-bottom-a/b). ***2)*** *The GAN-based methods* (Zhang et al. 2021; Niu et al. 2020;

*Equal contributions.
†Corresponding author.

A异常扩散：基于扩散模型的少样本异常图像生成

滕虎[1*]、张江宁[2*]、易冉[1†]、杜玉珍[1]、陈旭[2]、刘亮[2]、王亚标[2]、王成杰[1,21]上海交通大学[2]腾讯优图实验室
{hu-teng, ranyi, Haaaaaaaaaa}@sjtu.edu.cn;
{vtzhang, cxxuchen, leoneliu, caseywang, jasoncjwang}@tencent.com;

摘要

异常检测在工业制造中扮演着重要角色。由于异常数据不足，现有异常检测方法的性能受到限制。尽管已有异常生成方法被提出以扩充异常数据，但这些方法要么存在生成真实性不足的问题，要么面临生成异常与掩码对齐不准确的挑战。为解决上述问题，我们提出*AnomalyDiffusion*——一种基于扩散模型的少样本异常生成新方法，该方法利用从大规模数据集中学习到的潜在扩散模型强先验信息，在少样本训练数据下提升生成真实性。首先，我们提出空间异常嵌入机制，该机制由可学习的异常嵌入和从异常掩码编码的空间嵌入构成，将异常信息解耦为异常外观与位置信息。此外，为提升生成异常与异常掩码的对齐精度，我们引入自适应注意力重加权机制。该机制基于生成异常图像与正常样本的差异，动态引导模型更关注生成异常不明显区域，从而生成精确匹配的异常图像-掩码对。大量实验表明，我们的模型在生成真实性与多样性方面显著优于现有先进方法，并能有效提升下游异常检测任务性能。代码与数据详见：https://github.com/sjtuplayer/anomalydiffusion。

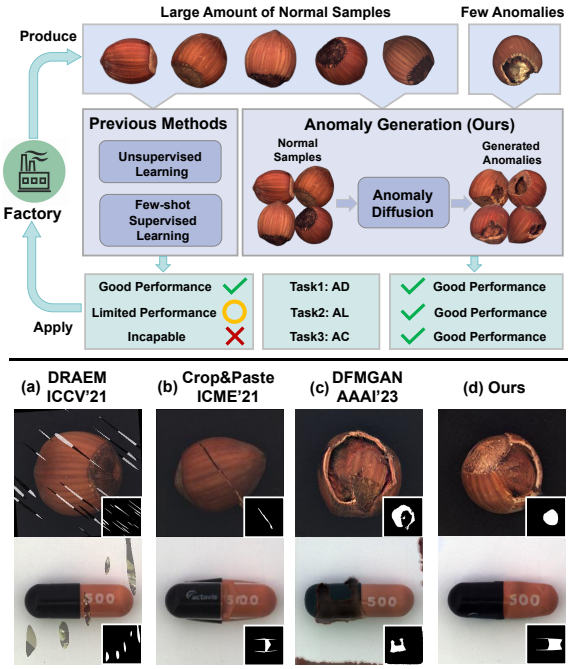

图1：***Top:*** 我们的模型生成大量异常数据，为下游异常检测（AD）、定位（AL）和分类（AC）任务提供支持，而先前方法因异常数据有限主要依赖无监督学习或少样本监督学习；***Bottom:*** 在榛子裂纹和胶囊挤压数据集上，我们模型与现有异常生成方法的效果对比，我们的生成结果具有最高真实度。

1 引言

近年来，工业异常检测算法在*i.e.,*异常检测、定位与分类方面，对工业生产起着至关重要的作用（Duan等人，2023年）。然而在实际工业生产中，异常样本数量极少，这给异常检测带来了巨大挑战（图1-顶部）。为缓解异常数据稀缺的问题，现有异常检测方法主要依赖仅使用正常样本的无监督学习方法（Zavrtanik、Kristan与Skočaj，2021年；Li等人，2021年），或小样本监督学习方法（Zhang等人，2023a年）。尽管这些方法在异常检测中表现良好，它们在异常定位方面性能有限，且无法处理异常分类问题。

为应对异常样本稀缺的问题，研究者提出通过异常生成方法来补充异常数据，主要分为两类：***1)*** *The model-free methods* 从现有异常样本或异常纹理数据集中随机裁剪并粘贴图像块至正常样本（Li et al. 2021; Lin et al. 2021; Zavrtanik, Kristan, and Skočaj 2021），但此类方法生成的数据真实性较差（图1-bottom-a/b）。***2)*** *The GAN-based methods* (Zhang et al. 2021; Niu et al. 2020;

*Equal contributions.
†Corresponding author.

Duan et al. 2023) utilize Generative Adversarial Networks (GANs) (Goodfellow et al. 2014) to generate anomalies, but most of them require a large amount of anomaly samples for training. The only few-shot generation model DFM-GAN (Duan et al. 2023) employs StyleGAN2 (Karras et al. 2020) pretrained on normal samples, and then performs domain adaption with a few anomaly samples. But the generated anomalies are not accurately aligned with the anomaly masks (Fig. 1-bottom-c). To sum up, the existing anomaly generation methods either fail to generate authentic anomalies or accurately-aligned anomalous image-mask pairs by learning from few-shot anomaly data, which limits their improvement in the downstream anomaly inspection tasks.

To address the above issues, we propose *AnomalyDiffusion*, a novel anomaly generation method based on the diffusion model, which generates anomalies onto the input normal samples with the anomaly masks. By leveraging the strong prior information of a pretrained LDM (Rombach et al. 2022) learned from large-scale dataset (Schuhmann et al. 2021), we can extract better anomaly representation using only a few anomaly images and boost the generation authenticity and diversity. To generate anomalies with specified type and locations, we propose *Spatial Anomaly Embedding*, which disentangles anomaly information into an anomaly embedding (a learned textual embedding to represent the appearance type of anomaly) and a spatial embedding (encoded from an anomaly mask to indicate the locations). By disentangling anomaly location from appearance, we can generate anomalies in any desired positions, which enables producing a large amount of anomalous image-mask pairs for the downstream tasks. Moreover, we propose an *Adaptive Attention Re-weighting Mechanism* to allocate more attention to the areas with less noticeable generated anomalies, which dynamically adjusts the cross-attention maps based on disparities between the generated images and input normal samples during the diffusion inference stage. This adaptive mechanism results in accurately aligned generated anomaly images and anomaly masks, which greatly facilitates downstream anomaly localization tasks.

Extensive qualitative and quantitative experiments and comparisons demonstrate that our *AnomalyDiffusion* outperforms state-of-the-art anomaly generation models in terms of generation authenticity and diversity. Moreover, our generated anomaly images can be effectively applied to downstream anomaly inspection tasks, yielding a pixel-level **99.1%** AUROC and **81.4%** AP score in anomaly localization on MVTec (Bergmann et al. 2019). The main contribution of this paper can be summarized as follows:

- We propose *AnomalyDiffusion*, a few-shot diffusion-based anomaly generation method, which disentangles anomalies into anomaly embedding (for anomaly appearance) and spatial embedding (for anomaly location), and generates authentic and diverse anomaly images.
- We design *Adaptive Attention Re-weighting Mechanism*, which adaptively allocates more attention to the areas with less noticeable generated anomalies, improving the alignment between the generated anomalies and masks.
- Extensive experiments demonstrate the superiority of our model over the state-of-the-art competitors, and our generated anomaly data effectively improves the performance of downstream anomaly inspection tasks, which will be released to facilitate future research.

2 Related Work

2.1 Generative Models

Generative models. VAEs (Kingma and Welling 2013) and GANs (Goodfellow et al. 2014) have achieved great progress in image generation. Recently, diffusion model (Nichol and Dhariwal 2021) demonstrates a more enhanced potential in generating images in a wide range of domains. Latent diffusion model (LDM) (Rombach et al. 2022) further improves the generation ability through compression of the diffusion space and obtains strong prior information by training on LAION dataset (Schuhmann et al. 2021).

Few-shot image generation. Few-shot image generation aims to generate diverse images with limited training data. Early methods propose modifying network weights (Mo, Cho, and Shin 2020), using various regularization techniques (Li et al. 2020) and data augmentation (Tran et al. 2021) to prevent overfitting. To deal with the extremely limited data (less than 10), recent works (Ojha et al. 2021; Wang et al. 2022; Hu et al. 2023) introduce cross-domain consistency losses to keep the generated distribution. Textual Inversion (Gal et al. 2022) and Dreambooth (Ruiz et al. 2023) encode a few images into the textual space of a pretrained LDM, but cannot control the generated locations accurately.

2.2 Anomaly Inspection

Anomaly inspection. The anomaly inspection task consists of anomaly detection, localization and classification. Some existing methods (Schlegl et al. 2017, 2019; Liang et al. 2023) rely on image reconstruction, comparing the differences between reconstructed images and anomaly images to achieve anomaly detection and localization. Moreover, deep feature modeling-based methods (Lee, Lee, and Song 2022; Cao et al. 2022; Roth et al. 2022; Gu et al. 2023; Wang et al. 2023) build a feature space for input images and then compare the differences between features to detect and localize anomalies. Additionally, some supervised learning-based methods (Zhang et al. 2023a) utilize a small number of anomaly samples to enhance the anomaly localization capabilities. Some studies conduct zero-/few-shot AD without using or with only a small number of anomaly samples (Jeong et al. 2023; Cao et al. 2023; Chen, Han, and Zhang 2023; Chen et al. 2023; Zhang et al. 2023b; Huang et al. 2022). Although these methods have shown promising results in anomaly detection, their performance in anomaly localization is still limited due to the lack of anomaly data.

Anomaly generation. The scarcity of anomaly data has sparked research interest in anomaly generation. DRAEM (Zavrtanik, Kristan, and Skočaj 2021), CutPaste (Li et al. 2021), Crop-Paste (Lin et al. 2021) and PRN (Zhang et al. 2023a) crop and paste unrelated textures or existing anomalies into normal sample. But they either generate less realistic anomalies or have limited generated

段等人（2023）采用生成对抗网络（GANs）（Goodfellow等人，2014）生成异常样本，但大多数方法需要大量异常样本进行训练。目前唯一的少样本生成模型DFM-GAN（段等人，2023）利用在正常样本上预训练的StyleGAN2（Karras等人，2020），随后通过少量异常样本进行领域自适应。但生成的异常样本与异常掩码未能精确对齐（图1底部-c）。综上所述，现有异常生成方法要么无法生成逼真异常样本，要么难以通过少样本异常数据学习生成精确对齐的图像-掩码对，这限制了下游异常检测任务的性能提升。

针对上述问题，我们提出*AnomalyDiffu- sion*——一种基于扩散模型的新型异常生成方法，该方法通过异常掩码将异常生成到输入的正常样本上。通过利用预训练LDM（Rombach等人2022）从大规模数据集（Schuhmann等人2021）中学到的强大先验信息，我们仅需少量异常图像即可提取更优质的异常表征，同时提升生成的真实性与多样性。为实现指定类型和位置的异常生成，我们提出*Spatial Anomaly Em- bedding*，该方法将异常信息解耦为异常嵌入（用于表征异常外观类型的可学习文本嵌入）和空间嵌入（由异常掩码编码以指示位置）。通过将异常位置与外观解耦，我们可在任意指定位置生成异常，从而为下游任务批量生成异常图像-掩码对。此外，我们提出
*Adaptive Attention Re-weighting Mechanism*机制，通过扩散推理阶段动态调整生成图像与输入正常样本差异区域的交叉注意力图，使模型更关注生成异常不明显的区域。这种自适应机制确保了生成的异常图像与异常掩码的精确对齐，极大促进了下游异常定位任务的性能。

广泛的定性和定量实验及比较表明，我们的*AnomalyDiffusion*在生成真实性和多样性方面优于最先进的异常生成模型。此外，我们生成的异常图像能够有效应用于下游异常检测任务，在MVTec数据集（Bergmann等人，2019）的异常定位中实现了像素级99.1%的AUROC和81.4%的AP评分。本文的主要贡献可归纳如下：

- 我们提出了*AnomalyDiffusion*，一种基于扩散模型的少样本异常生成方法，该方法将异常解耦为异常嵌入（控制异常外观）和空间嵌入（控制异常位置），从而生成逼真且多样化的异常图像。
- 我们设计了
Adaptive Attention Re-weighting Mechanism，它能自适应地将更多注意力分配给生成异常不太明显的区域，从而改善生成异常与掩码之间的对齐。
- 大量实验证明了
我们的模型相较于现有最优竞争对手表现出色，且生成的异常数据有效提升了下游异常检测任务的性能。为助力未来研究，相关资源将予以公开。

2 相关工作

2.1 生成模型

生成模型。变分自编码器（Kingma和Welling 2013）与生成对抗网络（Goodfellow等人2014）在图像生成领域取得了重大进展。近期，扩散模型（Nichol和Dhariwal 2021）展现出在跨领域图像生成方面更强大的潜力。潜在扩散模型（Rombach等人2022）通过压缩扩散空间进一步提升了生成能力，并借助在LAION数据集（Schuhmann等人2021）上的训练获得了强先验信息。

小样本图像生成。小样本图像生成旨在利用有限训练数据生成多样化图像。早期方法通过调整网络权重（Mo等人2020）、采用多种正则化技术（Li等人2020）和数据增强（Tran等人2021）来防止过拟合。针对极端有限数据（少于10张），近期研究（Ojha等人2021；Wang等人2022；Hu等人2023）引入跨域一致性损失以保持生成分布。文本反转（Gal等人2022）和Dreambooth（Ruiz等人2023）将少量图像编码至预训练LDM的文本空间，但无法精确控制生成位置。

2.2 异常检测

异常检测。异常检测任务包括异常识别、定位与分类。现有方法中，部分研究（Schlegl等人2017、2019；Liang等人2023）依赖图像重建技术，通过比对重建图像与异常图像的差异来实现异常识别与定位。此外，基于深度特征建模的方法（Lee、Lee与Song 2022；Cao等人2022；Roth等人2022；Gu等人2023；Wang等人2023）通过构建输入图像的特征空间，进而比较特征差异以检测和定位异常。另有基于监督学习的方法（Zhang等人2023a）利用少量异常样本来增强异常定位能力。部分研究在不使用或仅使用少量异常样本的情况下实现零样本/少样本异常检测（Jeong等人2023；Cao等人2023；Chen、Han与Zhang 2023；Chen等人2023；Zhang等人2023b；Huang等人2022）。尽管这些方法在异常检测方面成效显著，但由于异常数据匮乏，其在异常定位方面的性能仍受限。

异常生成。异常数据的稀缺性引发了人们对异常生成的研究兴趣。DRAEM（Zavrtanik、Kristan和Skočaj 2021）、Cut-Paste（Li等人2021）、Crop-Paste（Lin等人2021）和PRN（Zhang等人2023a）通过裁剪并粘贴无关纹理或现有异常到正常样本中。但它们要么生成的异常不够真实，要么生成能力有限。

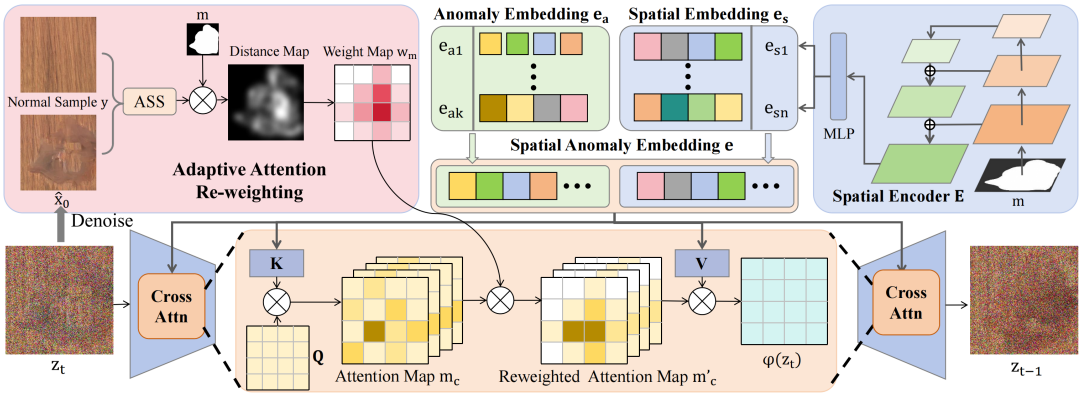

Figure 2: **Overall framework of our AnomalyDiffusion:** ***1)*** The *Spatial Anomaly Embedding* e, consisting of an anomaly embedding e_a (a learned textual embedding to represent anomaly appearance type) and a spatial embedding e_s (encoded from an input anomaly mask m to indicate anomaly locations), serves as the text condition to guide the anomaly generation process; ***2)*** The *Adaptive Attention Re-weighting Mechanism* computes the weight map w_m based on the difference between the denoised image $\hat{x}_0$ and the input normal sample y, and adaptively reweights the cross-attention map m_c by the weight map w_m to help the model focus more on the less noticeable anomaly areas during the denoising process.

diversity. The GAN-based model SDGAN (Niu et al. 2020) and Defect-GAN (Zhang et al. 2021), generate anomalies on normal samples by learning from anomaly data. But they require a large amount of anomaly data and cannot generate anomaly mask. DFMGAN (Duan et al. 2023) transfers a StyleGAN2 (Karras et al. 2020) pretained on normal samples to anomaly domain, but lacks generation authenticity and accurate alignment between generated anomalies and masks. In contrast, our model incorporates spatial anomaly embedding and adaptive attention re-weighting mechanism, which can generate anomalous image-mask pairs with great diversity and authenticity.

3 Method

Our *AnomalyDiffusion* aims to generate a large amount of anomaly data aligned with anomaly masks, by learning from a few anomaly samples. The inputs to our model include an anomaly-free sample y and an anomaly mask m, and the output is an image with anomalies generated in the mask area, while the remaining region is consistent with the input anomaly-free sample.

As shown in Fig. 2, our *AnomalyDiffusion* is developed based on Latent Diffusion Model (Rombach et al. 2022). To disentangle the anomaly location information from anomaly appearance, we propose Spatial Anomaly Embedding e, which consists of an anomaly embedding e_a (for anomaly appearance) and a spatial embedding e_s (for anomaly location). Moreover, to enhance the alignment between the generated anomalies and given masks, we introduce an Adaptive Attention Re-weighting Mechanism, which helps the model to allocate more attention to the areas with less noticeable generated anomalies (Fig. 3(c)).

Specifically, the anomaly embedding e_a provides the anomaly appearance type information, with one e_a corresponding to a certain type of anomaly (*e.g.,* hazelnut-crack, capsule-squeeze), which is learned by our masked textual inversion (Sec. 3.2). And the spatial embedding e_s provides the anomaly location information, which is encoded from the input anomaly mask m by a spatial encoder E (shared among all anomalies). By combining the anomaly embedding e_a with spatial embedding e_s, the spatial anomaly embedding e contains both the anomaly appearance and spatial information, which serves as the text condition in the diffusion model to guide the generation process. With the the spatial anomaly embedding as condition, given a normal sample, we generate an anomaly image with the blended diffusion process (Avrahami, Lischinski, and Fried 2022):

$$x_{t-1} = p_\theta(x_{t-1}|x_t, e) \odot m + q(y_{t-1}|y_0) \odot (1 - m), \quad (1)$$

where x_t is the generated anomaly image at timestep t, y_0 is the input normal sample, m is the anomaly mask, and $q(\cdot)$ and $p_\theta(\cdot)$ are the forward and backward process in diffusion as illustrated in Sec. 3.1.

3.1 Preliminaries

Denoising diffusion probabilistic models (DDPM) (Ho, Jain, and Abbeel 2020) has achieved significant success in image generation tasks. It employs a forward process to add noise into the data and then learns denoising during the backward process, thereby accomplishing the fitting of the training data distribution. With the training image x_0, the forward process $q(\cdot)$ in diffusion model is formulated as:

$$\begin{aligned} q\left(x_1, \ldots, x_T \mid x_0\right) &:= \prod_{t=1}^{T} q\left(x_t \mid x_{t-1}\right), \\ q\left(x_t \mid x_{t-1}\right) &:= \mathcal{N}\left(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t \mathbf{I}\right), \end{aligned} \quad (2)$$

where β_t is the variance at timestep t.

The backward process is approximated by predicting the mean $\mu_\theta(x_t, t)$ and variance $\Sigma_\theta\left(x_t, t\right)$ (set as a constant in DDPM) of a Gaussian distribution iteratively by:

$$p_\theta\left(x_{t-1} \mid x_t\right) := \mathcal{N}\left(x_{t-1}; \mu_\theta\left(x_t, t\right), \Sigma_\theta\left(x_t, t\right)\right). \quad (3)$$

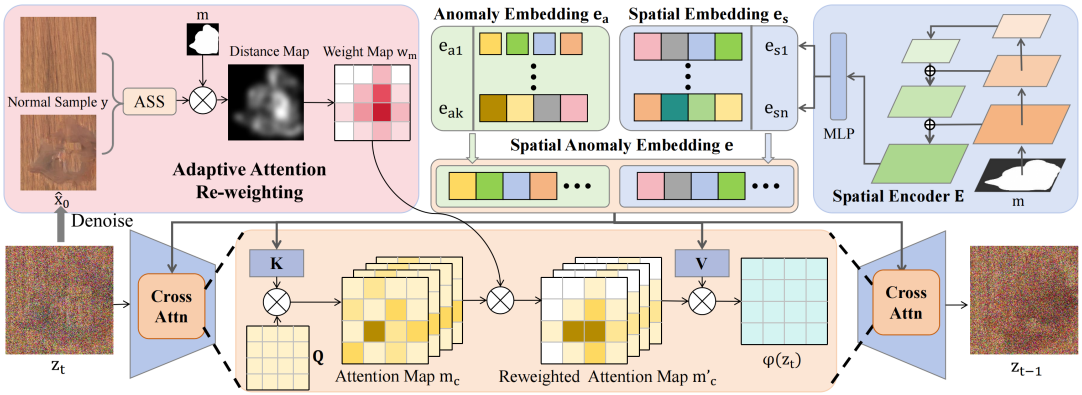

图2：我们的AnomalyDiffusion整体框架：***1)*** *Spatial Anomaly Embedding* e由表示异常外观类型的已学习文本嵌入e_a (a)和从输入异常掩码m编码以指示异常位置)的空间嵌入e_s (组成，作为文本条件指导异常生成过程；***2)*** *Adaptive Attention Re-weighting Mechanism*根据去噪图像$\hat{x}_0$与输入正常样本y之间的差异计算权重图w_m，并通过权重图w_m自适应重加权交叉注意力图m_c，以帮助模型在去噪过程中更关注不易察觉的异常区域。

多样性。基于GAN的模型SDGAN（Niu等人，2020年）和Defect-GAN（Zhang等人，2021年）通过从异常数据中学习，在正常样本上生成异常。但它们需要大量异常数据且无法生成异常掩码。DFMGAN（Duan等人，2023年）将在正常样本上预训练的StyleGAN2（Karras等人，2020年）迁移到异常领域，但存在生成真实性不足以及生成异常与掩码间对齐不精确的问题。相比之下，我们的模型融合了空间异常嵌入和自适应注意力重加权机制，能够生成具有高度多样性和真实性的异常图像-掩码对。

3 方法

我们的*AnomalyDiffusion*旨在通过从少量异常样本中学习，生成与异常掩码对齐的大量异常数据。模型的输入包括无异常样本y和异常掩码m，输出则是在掩码区域生成异常、其余区域与输入无异常样本保持一致的图像。

如图2所示，我们的*AnomalyDiffusion*基于隐扩散模型（Rombach等人，2022）开发。为了从异常外观中解耦异常位置信息，我们提出了空间异常嵌入e，该模块包含用于异常外观的e_a (异常嵌入)和用于异常位置的e_s (空间嵌入)。此外，为增强生成异常与给定掩码之间的对齐效果，我们引入了自适应注意力重加权机制，该机制能帮助模型将更多注意力分配到生成异常不够明显的区域（图3(c)）。

具体而言，异常嵌入e_a提供了异常外观类型信息，每个e_a对应特定类型的异常（*e.g.,*榛果裂纹、胶囊挤压），这是通过我们的掩码文本反演学习的（第3.2节）。而空间嵌入e_s则提供异常位置信息由输入异常掩码m通过一个所有异常共享的空间编码器E (编码而成。通过将异常嵌入e_a与空间嵌入e_s相结合，空间异常嵌入e同时包含异常外观和空间信息，作为扩散模型中的文本条件来指导生成过程。以空间异常嵌入为条件，给定正常样本时，我们通过混合扩散过程生成异常图像（Avrahami, Lischinski, and Fried 2022）：

$$x_{t-1} = p_\theta(x_{t-1}|x_t, e) \odot m + q(y_{t-1}|y_0) \odot (1-m), \quad (1)$$

其中 x_t 是时间步 t 生成的异常图像，y_0 是输入的正常样本，m 是异常掩码，$q(\cdot)$ 和 $p_\theta(\cdot)$ 分别表示如第 3.1 节所述的扩散前向过程与反向过程。

3.1 预备知识

去噪扩散概率模型（DDPM）（Ho等人2020）在图像生成任务中取得了显著成功。该模型采用前向过程向数据添加噪声，随后通过反向过程学习去噪，从而完成对训练数据分布的拟合。给定训练图像x_0，扩散模型中的前向过程$q(\cdot)$可表述为：

$$\begin{aligned} q\left(x_1, \ldots, x_T \mid x_0\right) &:= \prod_{t=1}^{T} q\left(x_t \mid x_{t-1}\right), \\ q\left(x_t \mid x_{t-1}\right) &:= \mathcal{N}\left(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t \mathbf{I}\right), \end{aligned} \quad (2)$$

其中β_t是时间步t的方差。

反向过程通过迭代预测高斯分布的均值$\mu_\theta(x_t, t)$和方差$\Sigma_\theta\left(x_t, t\right)$ (来近似实现，其中方差在DDPM)中被设为常数：

$$p_\theta\left(x_{t-1} \mid x_t\right) := \mathcal{N}\left(x_{t-1}; \mu_\theta\left(x_t, t\right), \Sigma_\theta\left(x_t, t\right)\right). \quad (3)$$

Textual inversion (Gal et al. 2022) utilizes a pre-trained Latent Diffusion Model to extract the shared content information in few-shot input samples by optimizing text embeddings. With the refined text embeddings as condition c, textual inversion can generate novel images x_0 with similar contents of input images by:

$$x_0 = \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t, c),\ x_T \sim \mathcal{N}(0,1). \tag{4}$$

3.2 Spatial Anomaly Embedding

Disentangle spatial information from anomaly appearance. We aim at controllable anomaly generation with specified anomaly type and location. A direct solution is to control anomaly type by textual embedding learned from textual inversion (Gal et al. 2022), and control anomaly location by the input mask. However, textual inversion tends to capture the location of anomalies along with the anomaly type information, which results in the generated anomalies only distributed in specific locations. To address the issue, we propose to disentangle the textual embedding into two parts, where one part (the spatial embedding e_s) is directly encoded from the anomaly mask to indicate the anomaly location, leaving the rest (the anomaly embedding e_a) to only learn anomaly type information. We name our decomposed textual embedding as Spatial Anomaly Embedding.

Anomaly embedding is a learned textual embedding that represents the anomaly appearance type information. Different from textual inversion method that learns the features of the entire image, in anomaly generation, our model only needs to focus on anomaly areas, without requiring information of the entire image. Therefore, we introduce *masked textual inversion*, where we mask out irrelevant background and normal regions of the anomaly image, and only the anomaly regions are visible to the model. We initialize the anomaly embedding e_a with k tokens and optimize it using the masked diffusion loss:

$$\mathcal{L}_{dif} = \|m \odot (\epsilon - \epsilon_\theta\left(z_t, t, \{e_a, e_s\}\right))\|_2^2, \tag{5}$$

where $\epsilon \sim \mathcal{N}(0,1)$ and z_t is the noised latent code of the input image x at timestep t.

Spatial embedding. To provide accurate spatial information of the anomaly locations, we introduce a spatial encoder E that encodes the input anomaly mask m into spatial embedding e_s, which is in the form of textual embedding and contains precise location information from the mask. Specifically, we input the anomaly mask into ResNet-50 (He et al. 2016) to extract the image features in different layers and fuse them together by Feature Pyramid Networks (Lin et al. 2017). Finally, several fully-connected networks are employed to map the fused features into textual embedding space, with each network predicting one text token, thereby outputting the final spatial embedding e_s with n tokens.

Overall training framework. For each anomaly type i, we employ an anomaly embedding $e_{a,i}$ to extract its appearance information, while all anomaly categories share a common spatial encoder E. For a set of image-mask pairs (x_i, m_i) in the training data, we first input anomaly mask m_i into spatial encoder E to obtain the spatial embedding $e_s = E(m_i)$. Then, we concatenate the anomaly embedding $e_{a,i}$ and the spatial embedding e_s together to obtain our spatial anomaly embedding $e = \{e_a, e_s\}$. Finally, the concatenated textual embedding e is used as the text condition to the diffusion model, and the training process can be formulated as:

$$e_a^*, E^* = \arg\min_{e_a, E} \mathbb{E}_{z\sim\mathcal{E}(x_i), m_i, \epsilon, t}\mathcal{L}_{dif}. \tag{6}$$

where $\mathcal{E}(\cdot)$ is the image encoder of latent diffusion model and $\epsilon \sim \mathcal{N}(0,1)$.

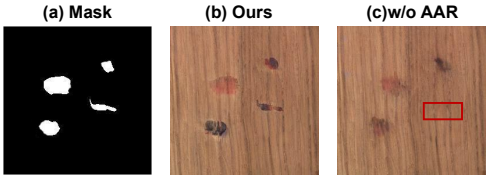

Figure 3: Comparison between the models w/ (Ours) and w/o Adaptive Attention Re-weighting (AAR). The model w/o AAR cannot generate anomalies to fill the entire mask.

3.3 Adaptive Attention Re-Weighting

With the spatial anomaly embedding e, we can use it as the text condition to guide the generation of anomaly images by Eq. (1). However, the generated anomaly images sometimes fail to fill the entire mask, especially when there are multiple anomaly regions in the mask or when the mask has irregular shapes (Fig. 3-a/c). In such cases, the generated anomalies are usually not well aligned with the mask, which limits the improvement in downstream anomaly localization task. To address this problem, we propose an adaptive attention re–weighting mechanism, which allocates more attention to the areas with less noticeable generated anomalies during the denoising process, thereby facilitating better alignment between the generated anomalies and the anomaly masks.

Adaptive attention weight map. Specifically, at the t-th denoising step, we calculate the corresponding $\hat{x}_0 = D(p_\theta(\hat{z}_0|z_t, e))$ (where D is the decoder of LDM). Then, we calculate the pixel-level difference between $\hat{x}_0$ and the normal sample y within the mask m. Based on the difference, we calculate the weight map w_m by the Adaptive Scaling Softmax (ASS) operation:

$$w_m = \|m\|_1 \cdot Softmax(f(\|m \odot y - m \odot \hat{x}_0\|_2^2)), \tag{7}$$

where $f(x) = \frac{1}{x}$ when $x! = 0$ and $f(x) = -\infty$ otherwise. For the regions within the mask that are similar to normal samples, the generated anomalies in these regions are less noticeable. To enhance the anomaly generation effects, these regions are assigned higher weights by Eq. (7) and allocated with more attention by attention re-weighting.

Attention re-weighting. We employ the weight map w_m to adaptively control the cross-attention, in order to guide our model to focus more on the areas with less noticeable generated anomalies. In our cross-attention calculation, Query is calculated from the latent code z_t, and Key and Value are calculated from our spatial anomaly embedding e:

$$Q = W_Q^{(i)} \cdot \varphi_i\left(z_t\right), K = W_K^{(i)} \cdot e, V = W_V^{(i)} \cdot e, \tag{8}$$

文本反转（Gal等人，2022）利用预训练的潜在扩散模型，通过优化文本嵌入来提取少样本输入样本中的共享内容信息。以优化后的文本嵌入作为条件{v*}，文本反转可通过以下方式生成与输入图像内容相似的新图像{v*}：

$$x_0 = \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t, c),\ x_T \sim \mathcal{N}(0,1). \tag{4}$$

3.2 空间异常嵌入

将异常外观与空间信息解耦。我们的目标是实现指定异常类型和位置的可控异常生成。一种直接解决方案是通过文本反转（Gal等人，2022年）学习的文本嵌入来控制异常类型，并通过输入掩码控制异常位置。然而，文本反转往往会同时捕获异常位置和异常类型信息，导致生成的异常仅分布在特定位置。为解决这个问题，我们提出将文本嵌入解耦为两部分：其中一部分（空间嵌入e_s）直接从异常掩码编码以指示异常位置，其余部分（异常嵌入e_a）仅学习异常类型信息。我们将这种分解后的文本嵌入命名为空间异常嵌入。

异常嵌入是一种学习得到的文本嵌入，用于表示异常外观类型信息。与学习整幅图像特征的文本反转方法不同，在异常生成过程中，我们的模型只需关注异常区域，无需获取整幅图像的信息。因此我们引入*masked textual inversion*方案：通过掩码处理异常图像中不相关的背景与正常区域，仅使异常区域对模型可见。我们使用k标记初始化异常嵌入e_a，并采用掩码扩散损失进行优化：

$$\mathcal{L}_{dif} = \|m \odot (\epsilon - \epsilon_\theta\left(z_t, t, \{e_a, e_s\}\right))\|_2^2, \tag{5}$$

其中 $\epsilon \sim \mathcal{N}(0,1)$ 且 z_t 是输入图像 x 在时间步 t 的含噪潜在代码。

空间嵌入。为了提供异常位置的准确空间信息，我们引入了一个空间编码器E，它将输入的异常掩码m编码为空间嵌入e_s。该嵌入采用文本嵌入的形式，并包含来自掩码的精确位置信息。具体而言，我们将异常掩码输入ResNet-50（He等人，2016）以提取不同层的图像特征，并通过特征金字塔网络（Lin等人，2017）将其融合。最后，使用多个全连接网络将融合特征映射到文本嵌入空间，每个网络预测一个文本标记，从而输出具有n个标记的最终空间嵌入e_s。

整体训练框架。针对每种异常类型i，我们采用异常嵌入$e_{a,i}$来提取其外观信息，而所有异常类别共享一个共同的空间编码器E。对于训练数据中的一组图像-掩码对(x_i, m_i)，我们首先将异常掩码m_i输入空间

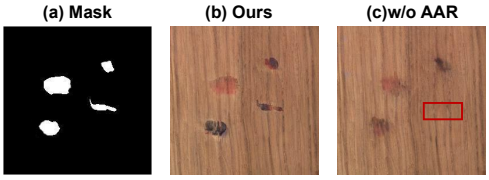

图3：采用自适应注意力重加权机制（Ours）与未采用该机制的模型对比。未配备AAR的模型无法生成完整填充掩码区域的异常内容。

编码器 E 以获取空间嵌入 $e_s = E(m_i)$。接着，我们将异常嵌入 $e_{a,i}$ 和空间嵌入 e_s 拼接起来，得到空间异常嵌入 $e = \{e_a, e_s\}$。最后，将拼接后的文本嵌入 e 作为扩散模型的文本条件，其训练过程可表述为：

$$e_a^*, E^* = \arg\min_{e_a, E} \mathbb{E}_{z\sim\mathcal{E}(x_i), m_i, \epsilon, t}\mathcal{L}_{dif}. \tag{6}$$

其中 $\mathcal{E}(\cdot)$ 是潜在扩散模型的图像编码器，$\epsilon \sim \mathcal{N}(0,1)$ 。

3.3 自适应注意力重加权

通过空间异常嵌入e，我们可以将其作为文本条件，依据公式(1)指导异常图像的生成。然而，生成的异常图像有时无法完整填充整个掩码区域，特别是在掩码包含多个异常区域或具有不规则形状时（图3-a/c）。此类情况下，生成的异常通常无法与掩码良好对齐，这限制了下游异常定位任务的性能提升。为解决该问题，我们提出了一种自适应注意力重加权机制，该机制在去噪过程中为生成异常不够明显的区域分配更多注意力，从而促进生成异常与异常掩码之间实现更佳的对齐效果。

自适应注意力权重图。具体来说，在第t个去噪步骤中，我们计算对应的$\hat{x}_0 = D(p_\theta(\hat{z}_0|z_t, e))$，其中$D$是LDM)的解码器。然后，我们计算$\hat{x}_0$与掩码$m$内正常样本$y$之间的像素级差异。基于该差异，我们通过自适应缩放Softmax（ASS）操作计算权重图w_m：

$$w_m = \|m\|_1 \cdot Softmax(f(\|m \odot y - m \odot \hat{x}_0\|_2^2)), \tag{7}$$

其中，当x时，$f(x) = \frac{1}{x}!$ 若=为0，否则为$f(x) = -\infty$。对于掩码内与正常样本相似的区域，这些区域生成的异常较不明显。为增强异常生成效果，这些区域通过公式(7)被赋予更高权重，并借助注意力重加权机制获得更多关注。

注意力重加权。我们采用权重映射w_m来自适应控制交叉注意力，以引导模型更关注生成异常不明显的区域。在交叉注意力计算中，查询通过潜码z_t计算得到，而键和值则通过我们的空间异常嵌入e计算：

$$Q = W_Q^{(i)} \cdot \varphi_i\left(z_t\right), K = W_K^{(i)} \cdot e, V = W_V^{(i)} \cdot e, \tag{8}$$

Category	DiffAug		CDC		Crop-Paste		SDGAN		Defect-GAN		DFMGAN		Ours	
	IS ↑	IC-L ↑	IS ↑	IC-L ↑	IS ↑	IC-L ↑	IS ↑	IC-L	IS ↑	IC-L ↑	IS ↑	IC-L ↑	IS ↑	IC-L ↑
bottle	1.59	0.03	1.52	0.04	1.43	0.04	1.57	0.06	1.39	0.07	**1.62**	0.12	1.58	**0.19**
cable	1.72	0.07	1.97	0.19	1.74	0.25	1.89	0.19	1.70	0.22	1.96	0.25	**2.13**	**0.41**
capsule	1.34	0.03	1.37	0.06	1.23	0.05	1.49	0.03	**1.59**	0.04	**1.59**	0.11	**1.59**	**0.21**
carpet	1.19	0.06	**1.25**	0.03	1.17	0.11	1.18	0.11	1.24	0.12	1.23	0.13	1.16	**0.24**
grid	1.96	0.06	1.97	0.07	2.00	0.12	1.95	0.10	2.01	0.12	1.97	0.13	**2.04**	**0.44**
hazel_nut	1.67	0.05	1.97	0.05	1.74	0.21	1.85	0.16	1.87	0.19	1.93	0.24	**2.13**	**0.31**
leather	2.07	0.06	1.80	0.07	1.47	0.14	2.04	0.12	**2.12**	0.14	2.06	0.17	1.94	**0.41**
metal nut	1.58	0.29	1.55	0.04	1.56	0.15	1.45	0.28	1.47	0.30	1.49	**0.32**	**1.96**	0.30
pill	1.53	0.05	1.56	0.06	1.49	0.11	1.61	0.07	1.61	0.10	**1.63**	0.16	1.61	**0.26**
screw	1.10	0.10	1.13	0.11	1.12	0.16	1.17	0.10	1.19	0.12	1.12	0.14	**1.28**	**0.30**
tile	1.93	0.09	2.10	0.12	1.83	0.20	2.53	0.21	2.35	0.22	2.39	0.22	**2.54**	**0.55**
toothbrush	1.33	0.06	1.63	0.06	1.30	0.08	1.78	0.03	**1.85**	0.03	1.82	0.18	1.68	**0.21**
transistor	1.34	0.05	1.61	0.13	1.39	0.15	**1.76**	0.13	1.47	0.13	1.64	0.25	1.57	**0.34**
wood	2.05	0.30	2.05	0.03	1.95	0.23	2.12	0.25	2.19	0.29	2.12	0.35	**2.33**	**0.37**
zipper	1.30	0.05	1.30	0.05	1.23	0.11	1.25	0.10	1.25	0.10	1.29	**0.27**	**1.39**	0.25
Average	1.58	0.09	1.65	0.07	1.51	0.14	1.71	0.13	1.69	0.15	1.72	0.20	**1.80**	**0.32**

Table 1: **Comparison on IS and IC-LPIPS on MVTec dataset.** Our model generates the most high-quality and diverse anomaly data, achieving the best IS and IC-LPIPS. **Bold** and underline represent optimal and sub-optimal results, respectively.

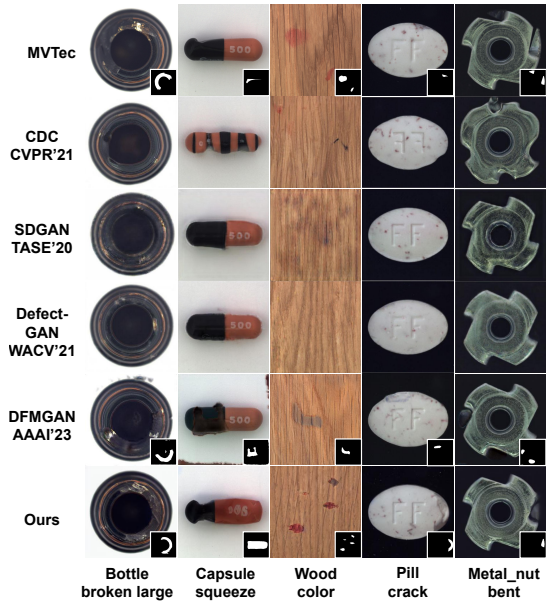

Figure 4: **Comparison on the generation results on MVTec.** Our model generates high quality anomaly images that are accurately aligned with the anomaly masks.

where φ_i is the intermediate representation of the U-Net (ϵ_θ) and the $W^{(i)}$s are the learnable projection matrices. The cross-attention calculation process is then formulated as $Attn(Q, K, V) = m_c \cdot V$, where $m_c = Softmax(\frac{QK^T}{\sqrt{d}})$ is the cross-attention map.

Considering the cross-attention map m_c controls the generated layout and effects, where higher attention leads to stronger generation effects (Hertz et al. 2022), we reweight the cross-attention map by our weight map: $m_c' = m_c \odot w_m$. The new cross-attention map m_c' focuses more on the areas with less noticeable generated anomalies, thereby enhancing the alignment accuracy between the generated anomalies and the input anomaly masks. The re-weighted cross attention is formulated as $RW\text{-}Attn(Q, K, V) = m_c' \cdot V$.

3.4 Mask Generation

Recall that our model requires anomaly masks as inputs. However, the number of real anomaly masks in the training datasets is very few, and the mask data lacks diversity even after augmentation, which motivates us to generate more anomaly masks by learning the real mask distribution. We employ textual inversion to learn a mask embedding e_m, which can be used as text condition to generate extensive anomaly masks. Specifically, we initialize the mask embedding e_m as k' random tokens and optimize it by:

$$e_m^* = \arg\min_{e_m} \mathbb{E}_{z\sim\mathcal{E}(m),\epsilon,t}\left[\|\epsilon - \epsilon_\theta\left(z_t, t, e_m\right)\|_2^2\right]. \quad (9)$$

With the learned mask embedding, we can generate extensive anomaly masks for each type of anomaly.

4 Experiments

4.1 Experiment Settings

Dataset. we conduct experiments on the widely used MVTec (Bergmann et al. 2019) dataset. We employ one-third of the anomaly data with the lowest ID numbers as the training set, reserving the remaining two-thirds for testing.

Implementation details. We assign $k = 8$ tokens for anomaly embedding, $n = 4$ tokens for spatial embedding, and $k' = 4$ tokens for mask embedding. For each type of anomaly, we generate 1000 anomalous image-mask pairs for the downstream anomaly inspection tasks. More details are recorded in the supplementary material.

Metric. ***1)*** **For generation,** due to the limited anomaly data, FID (Heusel et al. 2017) and KID (Bińkowski et al. 2018) are not reliable since the overfitted model tends to yield better scores (best) (Duan et al. 2023). Therefore, we employ Inception Score **(IS)**, which is independent of the given anomaly data, for a direct assessment of generation quality; we also introduce Intra-cluster pairwise LPIPS distance (**IC-LPIPS**) (Ojha et al. 2021) to measure the generation diversity. ***2)*** **for anomaly inspection,** we utilize **AUROC**, Average Precision (**AP**), and the $\mathbf{F_1}$**-max** score to evaluate the accuracy of anomaly detection and localization.

Category	DiffAug		CDC		Crop-Paste		SDGAN		Defect-GAN		DFMGAN		Ours	
	IS ↑	IC-L ↑	IS ↑	IC-L ↑	IS ↑	IC-L ↑	IS ↑	IC-L	IS ↑	IC-L ↑	IS ↑	IC-L ↑	IS ↑	IC-L ↑
bottle	1.59	0.03	1.52	0.04	1.43	0.04	1.57	0.06	1.39	0.07	**1.62**	0.12	1.58	**0.19**
cable	1.72	0.07	1.97	0.19	1.74	0.25	1.89	0.19	1.70	0.22	1.96	0.25	**2.13**	**0.41**
capsule	1.34	0.03	1.37	0.06	1.23	0.05	1.49	0.03	**1.59**	0.04	**1.59**	0.11	**1.59**	**0.21**
carpet	1.19	0.06	**1.25**	0.03	1.17	0.11	1.18	0.11	1.24	0.12	1.23	0.13	1.16	**0.24**
grid	1.96	0.06	1.97	0.07	2.00	0.12	1.95	0.10	2.01	0.12	1.97	0.13	**2.04**	**0.44**
hazel_nut	1.67	0.05	1.97	0.05	1.74	0.21	1.85	0.16	1.87	0.19	1.93	0.24	**2.13**	**0.31**
leather	2.07	0.06	1.80	0.07	1.47	0.14	2.04	0.12	**2.12**	0.14	2.06	0.17	1.94	**0.41**
metal nut	1.58	0.29	1.55	0.04	1.56	0.15	1.45	0.28	1.47	0.30	1.49	**0.32**	**1.96**	0.30
pill	1.53	0.05	1.56	0.06	1.49	0.11	1.61	0.07	1.61	0.10	**1.63**	0.16	1.61	**0.26**
screw	1.10	0.10	1.13	0.11	1.12	0.16	1.17	0.10	1.19	0.12	1.12	0.14	**1.28**	**0.30**
tile	1.93	0.09	2.10	0.12	1.83	0.20	2.53	0.21	2.35	0.22	2.39	0.22	**2.54**	**0.55**
toothbrush	1.33	0.06	1.63	0.06	1.30	0.08	1.78	0.03	**1.85**	0.03	1.82	0.18	1.68	**0.21**
transistor	1.34	0.05	1.61	0.13	1.39	0.15	**1.76**	0.13	1.47	0.13	1.64	0.25	1.57	**0.34**
wood	2.05	0.30	2.05	0.03	1.95	0.23	2.12	0.25	2.19	0.29	2.12	0.35	**2.33**	**0.37**
zipper	1.30	0.05	1.30	0.05	1.23	0.11	1.25	0.10	1.25	0.10	1.29	**0.27**	**1.39**	0.25
Average	1.58	0.09	1.65	0.07	1.51	0.14	1.71	0.13	1.69	0.15	1.72	0.20	**1.80**	**0.32**

表1：MVTec数据集上IS和IC-LPIPS的对比结果。我们的模型生成了最优质、最多样化的异常数据，在IS和IC-LPIPS指标上均取得最佳表现。粗体及下划线分别代表最优与次优结果。

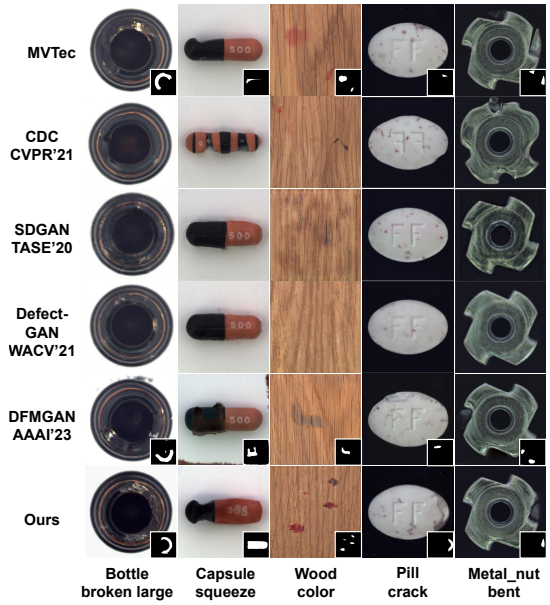

图4：MVTec数据集上的生成效果对比。我们的模型生成了与异常掩码精确对齐的高质量异常图像。

其中 φ_i 是 U-Net (ϵ_θ) 的中间表示，$W^{(i)}$ 是可学习的投影矩阵。交叉注意力计算过程随后被表述为 $Attn(Q,K,V) = m_c \cdot V$，其中 $m_c = Softmax(\frac{QK^T}{\sqrt{d}})$ 是交叉注意力图。

考虑到交叉注意力图m_c控制生成布局与效果，且注意力值越高生成效果越强（Hertz等人，2022），我们通过权重图$m_c' = m_c \odot w_m$对交叉注意力图进行重加权。新的交叉注意力图m_c'更聚焦于生成异常不明显的区域，从而提升生成异常与输入异常掩码之间的对齐精度。重加权交叉注意力的计算公式为$RW\text{-}Attn(Q,K,V) = m_c' \cdot V.$

3.4 掩码生成

回想一下，我们的模型需要异常掩码作为输入。然而，训练数据集中真实异常掩码的数量非常少，且即使经过数据增强，掩码数据仍缺乏多样性，这促使我们通过学习真实掩码分布来生成更多异常掩码。我们采用文本倒置技术来学习掩码嵌入{v*}，该嵌入可作为文本条件用于生成大量异常掩码。具体而言，我们将掩码嵌入{v*}初始化为{v*}个随机标记，并通过以下方式对其进行优化：

$$e_m^* = \arg\min_{e_m} \mathbb{E}_{z\sim\mathcal{E}(m),\epsilon,t}\left[\|\epsilon - \epsilon_\theta\left(z_t, t, e_m\right)\|_2^2\right]. \quad (9)$$

通过学习到的掩码嵌入，我们可以为每种异常类型生成大量的异常掩码。

4 实验

4.1 实验设置

数据集。我们在广泛使用的MVTec（Bergmann等人，2019）数据集上进行实验。我们采用ID号最低的三分之一异常数据作为训练集，其余三分之二保留用于测试。

实现细节。我们为异常嵌入分配$k = 8$个标记，为空间嵌入分配$n = 4$个标记，为掩码嵌入分配$k' = 4$个标记。针对每种异常类型，我们生成1000个异常图像-掩码对用于下游异常检测任务。更多细节记录在补充材料中。

指标。***1)*** 在生成任务中，由于异常数据有限，FID（Heusel等人，2017）和KID（Bińkowski等人，2018）的可靠性不足——因为过拟合的模型往往会获得更高（最优）的评分（Duan等人，2023）。因此我们采用与特定异常数据无关的初始分数（IS）来直接评估生成质量；同时引入集群内成对LPIPS距离（IC-LPIPS）（Ojha等人，2021）来衡量生成多样性。***2)*** 在异常检测方面，我们使用AUROC、平均精度（AP）以及$\mathbf{F_1}$-max分数来评估异常检测与定位的准确性。

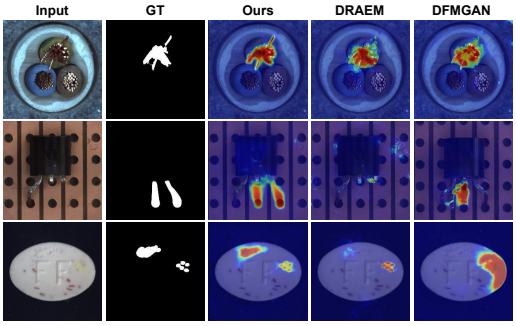

Figure 5: **Quantitative anomaly localization comparison** with an U-Net trained on the data generated by DRAEM, DFMGAN and our model. It shows that our model achieves the best anomaly localization results.

Category	DRAEM			PRN			DFMGAN			Ours		
	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max
bottle	96.7	80.2	74.0	97.5	76.4	71.3	98.9	90.2	83.9	**99.4**	**94.1**	**87.3**
cable	80.3	21.8	28.3	94.5	64.4	61.0	97.2	81.0	75.4	**99.2**	**90.8**	**83.5**
capsule	76.2	25.5	32.1	95.6	45.7	47.9	79.2	26.0	35.0	**98.8**	**57.2**	**59.8**
carpet	92.6	43.0	41.9	96.4	69.6	65.6	90.6	33.4	38.1	**98.6**	**81.2**	**74.6**
grid	**99.1**	**59.3**	58.7	98.9	58.6	**58.9**	75.2	14.3	20.5	98.3	52.9	54.6
hazelnut	98.8	73.6	68.5	98.0	73.9	68.2	99.7	95.2	89.5	**99.8**	**96.5**	**90.6**
leather	98.5	67.6	65.0	99.4	58.1	54.0	98.5	68.7	66.7	**99.8**	**79.6**	**71.0**
metal nut	96.9	84.2	74.5	97.9	93.0	87.1	99.3	98.1	**94.5**	**99.8**	**98.7**	94.0
pill	95.8	45.3	53.0	98.3	55.5	72.6	81.2	67.8	72.6	**99.8**	**97.0**	**90.8**
screw	91.0	30.1	35.7	94.0	47.7	49.8	58.8	2.2	5.3	**97.0**	**51.8**	**50.9**
tile	98.5	93.2	87.8	98.5	91.8	84.4	**99.5**	**97.1**	**91.6**	99.2	93.9	86.2
toothbrush	93.8	29.5	28.4	96.1	46.4	46.2	96.4	75.9	72.6	**99.2**	**76.5**	**73.4**
transistor	76.5	31.7	24.2	94.9	68.6	68.4	96.2	81.2	77.0	**99.3**	**92.6**	**85.7**
wood	98.8	**87.8**	**80.9**	96.2	74.2	67.4	95.3	70.7	65.8	**98.9**	84.6	74.5
zipper	93.4	65.4	64.7	98.4	79.0	73.7	92.9	65.6	64.9	**99.4**	**86.0**	**79.2**
Average	92.2	54.1	53.1	96.9	66.2	64.7	90.0	62.7	62.1	**99.1**	**81.4**	**76.3**

Table 2: **Comparison on pixel-level anomaly localization on MVTec dataset** by training an U-Net on the generated data from DRAEM, PRN, DFMGAN and our model.

Category	DRAEM			PRN			DFMGAN			Ours		
	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max
bottle	99.3	99.8	**98.9**	94.9	98.4	94.1	99.3	99.8	97.7	**99.8**	**99.9**	**98.9**
cable	72.1	83.2	79.2	86.3	92.0	84.0	95.9	97.8	93.8	**100**	**100**	**100**
capsule	93.2	98.7	94.0	84.9	95.8	94.3	92.8	98.5	94.5	**99.7**	**99.9**	**98.7**
carpet	95.3	98.7	93.4	92.6	97.8	92.1	67.9	87.9	87.3	**96.7**	**98.8**	**94.3**
grid	**99.8**	**99.9**	**98.8**	96.6	98.9	95.0	73.0	90.4	85.4	98.4	99.5	98.7
hazelnut	**100**	**100**	**100**	93.6	96.0	94.1	99.9	**100**	99.0	99.8	99.9	98.9
leather	**100**	**100**	**100**	99.1	99.7	97.6	99.9	**100**	99.2	**100**	**100**	**100**
metal_nut	97.8	99.6	97.6	97.8	99.5	96.9	99.3	99.8	99.2	**100**	**100**	**100**
pill	94.4	98.9	95.8	88.8	97.8	93.2	68.7	91.7	91.4	**98.0**	**99.6**	**97.0**
screw	88.5	96.3	89.3	84.1	94.7	87.2	22.3	64.7	85.3	**96.8**	**97.9**	**95.5**
tile	**100**	**100**	**100**	91.1	96.9	89.3	**100**	**100**	**100**	**100**	**100**	**100**
toothbrush	99.4	99.8	97.6	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
transistor	79.6	80.5	71.4	88.2	88.9	84.0	90.8	92.5	88.9	**100**	**100**	**100**
wood	**100**	**100**	**100**	77.5	92.7	86.7	98.4	99.4	98.8	98.4	99.4	98.8
zipper	**100**	**100**	**100**	98.7	99.7	97.6	99.7	99.9	99.4	99.9	**100**	99.4
Average	94.6	97.0	94.4	91.6	96.6	92.4	87.2	94.8	94.7	**99.2**	**99.7**	**98.7**

Table 3: **Comparison on image-level anomaly detection.**

4.2 Comparison in Anomaly Generation

Baseline. The compared anomaly generation methods can be classified into 2 groups: ***1)*** the models (Crop&Paste (Lin et al. 2021), DRAEM (Zavrtanik, Kristan, and Skočaj 2021), PRN (Zhang et al. 2023a) and DFMGAN (Duan et al. 2023)) that can generate anomalous image-mask pairs, which are employed to compare anomaly detection and localization; ***2)*** the models (DiffAug (Zhao et al. 2020), CDC (Ojha et al. 2021), Crop&Paste, SDGAN (Niu et al. 2020), DefectGAN (Zhang et al. 2021) and DFMGAN) that can generate specific anomaly types, which are employed to compare anomaly generation quality and classification.

Category	DiffAug	CDC	Crop&Paste	SDGAN	Defect-GAN	DFMGAN	Ours
bottle	48.84	38.76	52.71	48.84	53.49	56.59	**90.70**
cable	21.36	39.06	32.81	21.88	21.36	45.31	**67.19**
capsule	34.67	28.89	32.89	30.22	32.00	37.23	**66.67**
carpet	35.48	25.27	27.96	21.50	29.03	47.31	**58.06**
grid	28.33	35.83	28.33	30.83	27.50	40.83	**42.50**
hazelnut	65.28	54.86	59.03	43.75	61.11	81.94	**85.42**
leather	40.74	43.38	34.39	38.10	42.33	49.73	**61.90**
metalnut	58.85	48.44	59.89	44.27	56.77	**64.58**	59.38
pill	29.86	21.88	26.74	20.49	28.47	29.52	**59.38**
screw	25.10	32.92	28.81	26.75	28.81	37.45	**48.15**
tile	59.65	48.54	68.42	42.69	26.90	74.85	**84.21**
transistor	38.09	29.76	41.67	32.14	35.72	52.38	**60.71**
wood	41.27	28.57	47.62	30.95	24.60	49.21	**71.43**
zipper	22.76	14.63	26.42	21.54	18.70	27.64	**69.51**
Average	39.31	35.06	40.55	32.43	34.77	49.61	**66.09**

Table 4: **Comparison on anomaly classification accuracy** trained on the generated data by the anomaly generation models with a ResNet-18.

Anomaly generation quality. We compare our model with DiffAug, CDC, Crop&Paste, SDGAN, DefectGAN and DFMGAN on anomaly generation quality and diversity in Tab. 1. Since DRAEM and PRN crop random textures to imitate anomalies, we cannot compute IC-LPIPS for them. For each anomaly category, we allocate one-third of the anomaly data for training and generate 1000 anomaly images to compute IS and IC-LPIPS. It demonstrates that our model generates anomaly data with both the highest quality and diversity.

Moreover, we exhibit the generated anomalies in Fig. 4. It can be seen that our model excels in producing high-quality authentic anomalies that accurately align with their corresponding masks. In contrast, CDC yields visually perplexing outcomes, particularly for structural anomaly categories like capsule-squeeze. SDGAN and DefectGAN yield poor outputs, frequently encountering difficulties in generating anomalies such as pill-crack. The state-of-the-art model DFMGAN sometimes struggles to produce authentic anomalies and fails to keep the alignment between the generated anomalies and masks, as shown in metal nut-bent. More results are presented in supplementary material.

Anomaly generation for anomaly detection and localization. We compare the performance of our approach with existing anomaly generation methods in downstream anomaly detection and localization. Due to the inability of DiffAug and SDGAN to generate anomaly masks, we only compare our method with Crop&Paste, DRAEM, PRN and DFMGAN. For each method, we generate 1000 images per anomaly category and train an U-Net (Ronneberger, Fischer, and Brox 2015) alongside normal samples for anomaly localization. The localization outcomes are aggregated using average pooling to derive confidence scores for image-level anomaly detection (the same as DREAM). We compute pixel-level metrics including AUROC, AP, F_1-max. The results, as presented in Tab. 2, illustrate that our model outperforms other anomaly generation models at most conditions.Furthermore, we also evaluate image-level AUROC, AP, and F_1-max scores in Tab. 3. It demonstrates our model has the best anomaly detection performance compared to other methods. We also compare the qualitative results on anomaly localization in Fig. 5, which shows our superior performance in localizing the anomalies.

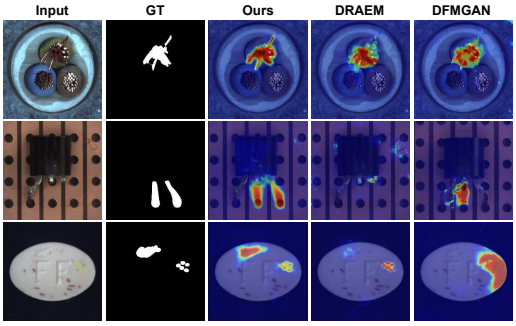

图5：与基于DRAEM、DFMGAN及本模型生成数据训练的U-Net进行定量异常定位对比。结果表明，本模型取得了最优的异常定位效果。

Category	DRAEM			PRN			DFMGAN			Ours		
	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max
bottle	96.7	80.2	74.0	97.5	76.4	71.3	98.9	90.2	83.9	**99.4**	**94.1**	**87.3**
cable	80.3	21.8	28.3	94.5	64.4	61.0	97.2	81.0	75.4	**99.2**	**90.8**	**83.5**
capsule	76.2	25.5	32.1	95.6	45.7	47.9	79.2	26.0	35.0	**98.8**	**57.2**	**59.8**
carpet	92.6	43.0	41.9	96.4	69.6	65.6	90.6	33.4	38.1	**98.6**	**81.2**	**74.6**
grid	**99.1**	**59.3**	58.7	98.9	58.6	**58.9**	75.2	14.3	20.5	98.3	52.9	54.6
hazelnut	98.8	73.6	68.5	98.0	73.9	68.2	99.7	95.2	89.5	**99.8**	**96.5**	**90.6**
leather	98.5	67.6	65.0	99.4	58.1	54.0	98.5	68.7	66.7	**99.8**	**79.6**	**71.0**
metal nut	96.9	84.2	74.5	97.9	93.0	87.1	99.3	98.1	**94.5**	**99.8**	**98.7**	94.0
pill	95.8	45.3	53.0	98.3	55.5	72.6	81.2	67.8	72.6	**99.8**	**97.0**	**90.8**
screw	91.0	30.1	35.7	94.0	47.7	49.8	58.8	2.2	5.3	**97.0**	**51.8**	**50.9**
tile	98.5	93.2	87.8	98.5	91.8	84.4	**99.5**	**97.1**	**91.6**	99.2	93.9	86.2
toothbrush	93.8	29.5	28.4	96.1	46.4	46.2	96.4	75.9	72.6	**99.2**	**76.5**	**73.4**
transistor	76.5	31.7	24.2	94.9	68.6	68.4	96.2	81.2	77.0	**99.3**	**92.6**	**85.7**
wood	98.8	**87.8**	**80.9**	96.2	74.2	67.4	95.3	70.7	65.8	**98.9**	84.6	74.5
zipper	93.4	65.4	64.7	98.4	79.0	73.7	92.9	65.6	64.9	**99.4**	**86.0**	**79.2**
Average	92.2	54.1	53.1	96.9	66.2	64.7	90.0	62.7	62.1	**99.1**	**81.4**	**76.3**

表2：通过在DRAEM、PRN、DFMGAN及本模型生成数据上训练U-Net，对MVTec数据集中像素级异常定位的对比结果。

Category	DRAEM			PRN			DFMGAN			Ours		
	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max
bottle	99.3	99.8	**98.9**	94.9	98.4	94.1	99.3	99.8	97.7	**99.8**	**99.9**	**98.9**
cable	72.1	83.2	79.2	86.3	92.0	84.0	95.9	97.8	93.8	**100**	**100**	**100**
capsule	93.2	98.7	94.0	84.9	95.8	94.3	92.8	98.5	94.5	**99.7**	**99.9**	**98.7**
carpet	95.3	98.7	93.4	92.6	97.8	92.1	67.9	87.9	87.3	**96.7**	**98.8**	**94.3**
grid	**99.8**	**99.9**	**98.8**	96.6	98.9	95.0	73.0	90.4	85.4	98.4	99.5	98.7
hazelnut	**100**	**100**	**100**	93.6	96.0	94.1	99.9	**100**	99.0	99.8	99.9	98.9
leather	**100**	**100**	**100**	99.1	99.7	97.6	99.9	**100**	99.2	**100**	**100**	**100**
metal_nut	97.8	99.6	97.6	97.8	99.5	96.9	99.3	99.8	99.2	**100**	**100**	**100**
pill	94.4	98.9	95.8	88.8	97.8	93.2	68.7	91.7	91.4	**98.0**	**99.6**	**97.0**
screw	88.5	96.3	89.3	84.1	94.7	87.2	22.3	64.7	85.3	**96.8**	**97.9**	**95.5**
tile	**100**	**100**	**100**	91.1	96.9	89.3	**100**	**100**	**100**	**100**	**100**	**100**
toothbrush	99.4	99.8	97.6	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
transistor	79.6	80.5	71.4	88.2	88.9	84.0	90.8	92.5	88.9	**100**	**100**	**100**
wood	**100**	**100**	**100**	77.5	92.7	86.7	98.4	99.4	98.8	98.4	99.4	98.8
zipper	**100**	**100**	**100**	98.7	99.7	97.6	99.7	99.9	99.4	99.9	**100**	99.4
Average	94.6	97.0	94.4	91.6	96.6	92.4	87.2	94.8	94.7	**99.2**	**99.7**	**98.7**

表3：图像级异常检测对比。

4.2 异常生成对比

基线。所比较的异常生成方法可分为两类：***1)*** 能够生成异常图像-掩码对的模型（Crop&Paste (Lin et al. 2021)、DRAEM (Zavrtanik, Kristan和Skočaj 2021)、PRN (Zhang et al. 2023a) 和 DFMGAN (Duan et al. 2023)），这些模型用于比较异常检测与定位性能；***2)*** 能够生成特定异常类型的模型（DiffAug (Zhao et al. 2020)、CDC (Ojha et al. 2021)、Crop&Paste、SDGAN (Niu et al. 2020)、Defect-GAN (Zhang et al. 2021) 和 DFMGAN），这些模型用于比较异常生成质量与分类效果。

Category	DiffAug	CDC	Crop&Paste	SDGAN	Defect-GAN	DFMGAN	Ours
bottle	48.84	38.76	52.71	48.84	53.49	56.59	**90.70**
cable	21.36	39.06	32.81	21.88	21.36	45.31	**67.19**
capsule	34.67	28.89	32.89	30.22	32.00	37.23	**66.67**
carpet	35.48	25.27	27.96	21.50	29.03	47.31	**58.06**
grid	28.33	35.83	28.33	30.83	27.50	40.83	**42.50**
hazelnut	65.28	54.86	59.03	43.75	61.11	81.94	**85.42**
leather	40.74	43.38	34.39	38.10	42.33	49.73	**61.90**
metalnut	58.85	48.44	59.89	44.27	56.77	**64.58**	59.38
pill	29.86	21.88	26.74	20.49	28.47	29.52	**59.38**
screw	25.10	32.92	28.81	26.75	28.81	37.45	**48.15**
tile	59.65	48.54	68.42	42.69	26.90	74.85	**84.21**
transistor	38.09	29.76	41.67	32.14	35.72	52.38	**60.71**
wood	41.27	28.57	47.62	30.95	24.60	49.21	**71.43**
zipper	22.76	14.63	26.42	21.54	18.70	27.64	**69.51**
Average	39.31	35.06	40.55	32.43	34.77	49.61	**66.09**

表4：基于ResNet-18的异常生成模型所生成数据训练的异常分类准确率对比。

异常生成质量。我们在表1中比较了本模型与DiffAug、CDC、Crop&Paste、SDGAN、DefectGAN和DFMGAN在异常生成质量与多样性方面的表现。由于DRAEM和PRN采用随机纹理裁剪来模拟异常，无法计算其IC-LPIPS指标。针对每个异常类别，我们分配三分之一异常数据用于训练，并生成1000张异常图像以计算IS和IC-LPIPS。实验表明，本模型生成的异常数据同时具备最优质量与多样性。

此外，我们在图4中展示了生成的异常样本。可以看出，我们的模型在生成高质量真实异常方面表现卓越，这些异常能精准对应其掩码位置。相比之下，CDC方法产生了视觉上令人困惑的结果，尤其在胶囊挤压等结构性异常类别中更为明显。SDGAN和DefectGAN生成的异常质量较差，在生成药片裂纹等异常时经常遇到困难。先进模型DFMGAN有时难以生成真实异常，且无法保持生成异常与掩码之间的对齐关系，如金属螺母弯曲案例所示。更多结果详见补充材料。

异常检测与定位的异常生成方法。我们将本方法与现有异常生成技术在后续异常检测与定位任务中的表现进行对比。由于DiffAug和SDGAN无法生成异常掩码，我们仅将本方法与Crop&Paste、DRAEM、PRN及DFMGAN进行对比。针对每种方法，我们为每个异常类别生成1000张图像，并配合正常样本训练U-Net网络（Ronneberger, Fischer, and Brox 2015）以执行异常定位。通过平均池化聚合定位结果，得出图像级异常检测的置信度评分（与DREAM方法一致）。我们计算了像素级指标包括AUROC、AP、F_1-max。如表2所示，结果表明在多数情况下我们的模型优于其他异常生成模型。此外，我们还在表3中评估了图像级AUROC、AP及F_1-max分数，证明本模型相比其他方法具有最佳的异常检测性能。图5展示了异常定位的定性结果对比，凸显了我们在异常定位方面的卓越性能。

Category	Unsupervised							Supervised			
	KDAD	CFLOW	DRAEM	SSPCAB	CFA	RD4AD	PatchCore	DevNet	DRA	PRN	Ours
bottle	94.7/50.5	98.8/49.9	99.1/88.5	98.9/88.6	98.9/50.9	98.8/51.0	97.6/75.0	96.7/67.9	91.7/41.5	**99.4**/92.3	99.3/**94.1**
cable	79.2/11.6	98.9/72.6	94.8/61.4	93.1/52.1	98.4/79.8	98.8/77.0	96.8/65.9	97.9/67.6	86.1/34.8	98.8/78.9	**99.2/90.8**
capsule	96.3/ 9.9	**99.5**/64.0	97.6/47.9	90.4/48.7	98.9/**71.1**	99.0/60.5	98.6/46.6	91.1/46.6	88.5/11.0	98.5/62.2	98.8/57.2
carpet	91.5/45.8	**99.7**/67.0	96.3/62.5	92.3/49.1	99.1/47.7	99.4/46.0	98.7/65.0	94.6/19.6	98.2/54.0	99.0/**82.0**	98.6/81.2
grid	89.0/ 7.6	99.1/**87.8**	99.5/53.2	**99.6**/58.2	98.6/82.9	98.0/75.4	97.2/23.6	90.2/44.9	86.2/28.6	98.4/45.7	98.3/52.9
hazelnut	95.0/34.2	97.9/67.2	99.5/88.1	99.6/94.5	98.5/80.2	94.2/57.2	97.6/55.2	76.9/46.8	88.8/20.3	99.7/93.8	**99.8/96.5**
leather	98.2/26.7	99.2/**91.1**	98.8/68.5	97.2/60.3	96.2/60.9	96.6/53.5	98.9/43.4	94.3/66.2	97.2/ 5.1	99.7/69.7	**99.8**/79.6
metal nut	81.7/30.6	98.8/78.2	**98.7**/91.6	99.3/95.1	98.6/74.6	97.3/53.8	97.5/86.6	93.3/57.4	80.3/30.6	99.7/98.0	**99.8/98.7**
pill	90.1/23.1	98.9/60.3	**97**.7/44.8	96.5/48.1	98.8/67.9	98.4/58.1	**97.0**/75.9	98.9/79.9	79.6/22.1	99.5/91.3	**99.8/97.0**
screw	95.4/ 5.9	98.8/45.7	**99.7/72.9**	99.1/62.0	98.7/61.4	99.1/51.8	98.7/34.2	66.5/21.1	51.0/ 5.1	97.5/44.9	97.0/51.8
tile	78.6/26.7	98.0/86.7	99.4/96.4	99.2/96.3	98.6/92.6	97.4/78.2	94.9/56.0	88.7/63.9	91.0/54.4	**99.6/96.5**	99.2/93.9
toothbrush	95.6/20.0	99.1/56.9	97.3/49.2	97.5/38.9	98.4/61.7	99.0/63.1	97.6/37.1	96.3/52.4	74.5/ 4.8	**99.6/78.1**	99.1/76.5
transistor	76.0/25.9	98.8/40.6	92.2/56.0	85.3/36.5	98.6/82.9	**99.6**/50.3	91.8/66.7	55.2/ 4.4	79.3/11.2	98.4/85.6	99.3/**92.6**
wood	88.3/24.7	98.9/47.2	97.6/81.6	97.2/77.1	97.6/25.6	**99.3**/39.1	95.7/54.3	93.1/47.9	82.9/21.0	97.8/82.6	98.9/**84.6**
zipper	95.1/30.5	96.5/63.9	98.6/73.6	98.1/78.2	95.9/53.9	**99.7**/52.7	98.5/63.1	92.4/53.1	96.8/42.3	98.8/77.6	99.4/**86.0**
Average	89.6/24.9	98.7/65.3	97.7/69.0	96.2/65.5	98.3/66.3	98.3/57.8	97.1/56.6	86.4/49.3	84.8/25.7	99.0/78.6	**99.1/81.4**

Table 5: **Comparison on pixel-level anomaly localization (AUROC/AP)** between the simple U-Net trained on our generated dataset and the existing anomaly detection methods with their official codes or pre-trained models.

Method			Metric		
SAE	Masked $\mathcal{L}$	AAR	AUROC	AP	F_1-max
			81.3	31.1	46.5
✓			90.3	51.2	60.7
✓	✓		95.0	64.9	68.8
	✓	✓	95.5	67.5	68.9
✓	✓	✓	**99.1**	**81.4**	**76.3**

Table 6: **Ablation study** on our spatial anomaly embedding (SAE), masked diffusion loss (Masked $\mathcal{L}$) and adaptive attention re-weighting mechanism (AAR).

Anomaly generation for anomaly classification. To further validate the generation quality of our model, we employ the generated anomalies to train a downstream anomaly classification model. Specifically, we adopt the experiment setting in DFMGAN, which trains a ResNet-34 (He et al. 2016) on the generated dataset and test the classification accuracy on the remaining shared test dataset. The comparison results are shown in Tab. 4. It can be seen that our model outperforms all other models in almost all types of components and the average accuracy (**66.09%**) surpasses that of the second-ranked DFMGAN (49.61%) by a margin of **16.48%**.

4.3 Comparison with Anomaly Detection Models

To further validate the efficacy of our model, we conduct a comparative experiment with the state-of-the-art anomaly detection methods CFLOW (Gudovskiy, Ishizaka, and Kozuka 2022), DRAEM (Zavrtanik, Kristan, and Skočaj 2021), CFA (Lee, Lee, and Song 2022), RD4AD (Deng and Li 2022), PatchCore (Roth et al. 2022), DevNet (Pang et al. 2021), DRA (Ding, Pang, and Shen 2022) and PRN (Zhang et al. 2023a). We employ their official codes or pre-trained models and evaluate them on the same testing dataset that we use. It is worth noting that due to the absence of the open-source code for PRN, we utilize the data provided in its paper. The comparison results on pixel-level AUROC and AP are presented in Tab. 5. It can be seen that although our model is only a simple U-Net, with the help of our generated anomaly data, it has a good performance in anomaly localization with the highest AP of **81.4%** and AUROC of **99.1%**, indicating the profound significance of our generated data for downstream anomaly inspection tasks.

4.4 Ablation Study

We evaluate the effectiveness of our components: spatial anomaly embedding (SAE), masked diffusion loss (Masked $\mathcal{L}$) and adaptive attention re-weighting mechanism (AAR). Not that the models without SAE employ only an anomaly embedding trained by textual inversion. We train 5 models: ***1)*** with none of these components; ***2)*** only SAE; ***3)*** SAE + masked $\mathcal{L}$; ***4)*** masked $\mathcal{L}$ + AAR and ***5)*** the full model (ours). We employ these models to generate 1000 anomalous image-mask pairs and train an U-Net for anomaly localization. We compare the pixel-level localization results in Tab. 6. It demonstrates that the omission of any of the proposed modules leads to a noticeable decline in the model's performance on anomaly localization, which validates the efficacy of the proposed modules. For more experiments, please refer to the supplementary material.

5 Conclusion

In this paper, we propose *Anomalydiffusion*, a novel anomaly generation model which generates anomalous image-mask pairs. We disentangle anomaly information into anomaly appearance and location information represented by anomaly embedding and spatial embedding in the textual space of LDM. Moreover, we also introduce an adaptive attention re-weighting mechanism, which helps our model focus more on the areas with less noticeable generated anomalies, thus improving the alignment between the generated anomalies and masks. Extensive experiments show that our model outperforms the existing anomaly generation methods and our generated anomaly data effectively improves the performance of the downstream anomaly inspection tasks. In future work, we would explore the application of a more potent diffusion model to enhance the resolution of the generated anomalies, which could further improve the performance.

Category	Unsupervised							Supervised			
	KDAD	CFLOW	DRAEM	SSPCAB	CFA	RD4AD	PatchCore	DevNet	DRA	PRN	Ours
bottle	94.7/50.5	98.8/49.9	99.1/88.5	98.9/88.6	98.9/50.9	98.8/51.0	97.6/75.0	96.7/67.9	91.7/41.5	**99.4**/92.3	99.3/**94.1**
cable	79.2/11.6	98.9/72.6	94.8/61.4	93.1/52.1	98.4/79.8	98.8/77.0	96.8/65.9	97.9/67.6	86.1/34.8	98.8/78.9	**99.2/90.8**
capsule	96.3/ 9.9	**99.5**/64.0	97.6/47.9	90.4/48.7	98.9/**71.1**	99.0/60.5	98.6/46.6	91.1/46.6	88.5/11.0	98.5/62.2	98.8/57.2
carpet	91.5/45.8	**99.7**/67.0	96.3/62.5	92.3/49.1	99.1/47.7	99.4/46.0	98.7/65.0	94.6/19.6	98.2/54.0	99.0/**82.0**	98.6/81.2
grid	89.0/ 7.6	99.1/**87.8**	99.5/53.2	**99.6**/58.2	98.6/82.9	98.0/75.4	97.2/23.6	90.2/44.9	86.2/28.6	98.4/45.7	98.3/52.9
hazelnut	95.0/34.2	97.9/67.2	99.5/88.1	99.6/94.5	98.5/80.2	94.2/57.2	97.6/55.2	76.9/46.8	88.8/20.3	99.7/93.8	**99.8/96.5**
leather	98.2/26.7	99.2/**91.1**	98.8/68.5	97.2/60.3	96.2/60.9	96.6/53.5	98.9/43.4	94.3/66.2	97.2/ 5.1	99.7/69.7	**99.8**/79.6
metal nut	81.7/30.6	98.8/78.2	**98.7**/91.6	99.3/95.1	98.6/74.6	97.3/53.8	97.5/86.6	93.3/57.4	80.3/30.6	99.7/98.0	**99.8/98.7**
pill	90.1/23.1	98.9/60.3	**97**.7/44.8	96.5/48.1	98.8/67.9	98.4/58.1	**97.0**/75.9	98.9/79.9	79.6/22.1	99.5/91.3	**99.8/97.0**
screw	95.4/ 5.9	98.8/45.7	**99.7/72.9**	99.1/62.0	98.7/61.4	99.1/51.8	98.7/34.2	66.5/21.1	51.0/ 5.1	97.5/44.9	97.0/51.8
tile	78.6/26.7	98.0/86.7	99.4/96.4	99.2/96.3	98.6/92.6	97.4/78.2	94.9/56.0	88.7/63.9	91.0/54.4	**99.6/96.5**	99.2/93.9
toothbrush	95.6/20.0	99.1/56.9	97.3/49.2	97.5/38.9	98.4/61.7	99.0/63.1	97.6/37.1	96.3/52.4	74.5/ 4.8	**99.6/78.1**	99.1/76.5
transistor	76.0/25.9	98.8/40.6	92.2/56.0	85.3/36.5	98.6/82.9	**99.6**/50.3	91.8/66.7	55.2/ 4.4	79.3/11.2	98.4/85.6	99.3/**92.6**
wood	88.3/24.7	98.9/47.2	97.6/81.6	97.2/77.1	97.6/25.6	**99.3**/39.1	95.7/54.3	93.1/47.9	82.9/21.0	97.8/82.6	98.9/**84.6**
zipper	95.1/30.5	96.5/63.9	98.6/73.6	98.1/78.2	95.9/53.9	**99.7**/52.7	98.5/63.1	92.4/53.1	96.8/42.3	98.8/77.6	99.4/**86.0**
Average	89.6/24.9	98.7/65.3	97.7/69.0	96.2/65.5	98.3/66.3	98.3/57.8	97.1/56.6	86.4/49.3	84.8/25.7	99.0/78.6	**99.1/81.4**

表5：基于我们生成数据训练的简易U-Net在像素级异常定位上的性能对比（AUROC/AP）已编辑数据集以及现有的异常检测方法，附带官方代码或预训练模型。

Method			Metric		
SAE	Masked $\mathcal{L}$	AAR	AUROC	AP	F_1-max
			81.3	31.1	46.5
✓			90.3	51.2	60.7
✓	✓		95.0	64.9	68.8
	✓	✓	95.5	67.5	68.9
✓	✓	✓	**99.1**	**81.4**	**76.3**

表6：关于我们的空间异常嵌入（SAE）、掩蔽扩散损失（Masked {v*}）和自适应注意力重加权机制（AAR）的消融研究。

用于异常分类的异常生成。为了进一步验证我们模型的生成质量，我们利用生成的异常样本来训练下游异常分类模型。具体而言，我们采用DFMGAN中的实验设置：在生成数据集上训练ResNet-34网络（He et al. 2016），并在剩余共享测试数据集上评估分类准确率。对比结果如表4所示。可以看出，在几乎所有组件类型中我们的模型均优于其他模型，且平均准确率（66.09%）较第二名DFMGAN（49.61%）高出16.48%。

4.3 与异常检测模型的比较

为进一步验证模型效能，我们与前沿异常检测方法进行了对比实验，包括CFLOW（Gudovskiy等人2022）、DRAEM（Zavrtanik等人2021）、CFA（Lee等人2022）、RD4AD（Deng等人2022）、PatchCore（Roth等人2022）、DevNet（Pang等人2021）、DRA（Ding等人2022）及PRN（Zhang等人2023a）。实验采用其官方代码或预训练模型，并在我们使用的同一测试集上评估。需说明的是，由于PRN未公开源代码，我们采用了其论文中提供的数据。像素级AUROC与AP的对比结果如表5所示。可见尽管我们的模型仅是简单U-Net架构，在生成异常数据的辅助下，其异常检测性能表现优异。定位方面取得了最高81.4%的AP值和99.1%的AUROC值，这表明我们生成的数据对下游异常检测任务具有深远意义。

4.4 消融实验

我们评估了各模块的有效性：空间异常嵌入（SAE）、掩码扩散损失（掩码$\mathcal{L}$）和自适应注意力重加权机制（AAR）。需注意，未使用SAE的模型仅采用通过文本反转训练的异常嵌入。我们训练了5个模型：***1)***不包含任何组件；***2)***仅含SAE；***3)***含SAE与掩码$\mathcal{L}$；***4)***含掩码$\mathcal{L}$ + 与AAR；***5)***为完整模型（我们的方法）。使用这些模型生成1000组异常图像-掩码对，并训练U-Net进行异常定位。表6展示了像素级定位结果的对比，结果表明省略任一提出的模块都会导致异常定位性能显著下降，从而验证了所提模块的有效性。更多实验请参阅补充材料。

5 结论

本文提出了一种新颖的异常生成模型*Anomalydiffusion*，该模型能够生成异常图像-掩码对。我们将异常信息解耦为异常外观和位置信息，分别通过LDM文本空间中的异常嵌入和空间嵌入进行表征。此外，我们还引入了自适应注意力重加权机制，该机制有助于模型更关注生成异常不明显的区域，从而提升生成异常与掩码之间的对齐精度。大量实验表明，我们的模型优于现有异常生成方法，且生成的异常数据能有效提升下游异常检测任务的性能。未来工作中，我们将探索应用更强大的扩散模型来提升生成异常的分辨率，从而进一步提高性能。

Acknowledgments

This work was supported by National Natural Science Foundation of China (62302297, 72192821, 62272447), Young Elite Scientists Sponsorship Program by CAST (2022QNRC001), Shanghai Sailing Program (22YF1420300), Beijing Natural Science Foundation (L222117), the Fundamental Research Funds for the Central Universities (YG2023QNB17), Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), Shanghai Science and Technology Commission (21511101200), CCF-Tencent Open Research Fund (RAGR20220121).

References

Avrahami, O.; Lischinski, D.; and Fried, O. 2022. Blended diffusion for text-driven editing of natural images. In *CVPR*, 18208–18218.

Bergmann, P.; Fauser, M.; Sattlegger, D.; and Steger, C. 2019. MVTec AD–A comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*, 9592–9600.

Bińkowski, M.; Sutherland, D. J.; Arbel, M.; and Gretton, A. 2018. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*.

Cao, Y.; Wan, Q.; Shen, W.; and Gao, L. 2022. Informative knowledge distillation for image anomaly segmentation. *Knowledge-Based Systems*, 248: 108846.

Cao, Y.; Xu, X.; Sun, C.; Cheng, Y.; Du, Z.; Gao, L.; and Shen, W. 2023. Segment Any Anomaly without Training via Hybrid Prompt Regularization. *arXiv preprint arXiv:2305.10724*.

Chen, X.; Han, Y.; and Zhang, J. 2023. A Zero-/Few-Shot Anomaly Classification and Segmentation Method for CVPR 2023 VAND Workshop Challenge Tracks 1&2: 1st Place on Zero-shot AD and 4th Place on Few-shot AD. *arXiv preprint arXiv:2305.17382*.

Chen, X.; Zhang, J.; Tian, G.; He, H.; Zhang, W.; Wang, Y.; Wang, C.; Wu, Y.; and Liu, Y. 2023. CLIP-AD: A Language-Guided Staged Dual-Path Model for Zero-shot Anomaly Detection. *arXiv preprint arXiv:2311.00453*.

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

Ding, C.; Pang, G.; and Shen, C. 2022. Catching both gray and black swans: Open-set supervised anomaly detection. In *CVPR*, 7388–7398.

Duan, Y.; Hong, Y.; Niu, L.; and Zhang, L. 2023. Few-Shot Defect Image Generation via Defect-Aware Feature Manipulation. In *AAAI*, volume 37, 571–578.

Gal, R.; Alaluf, Y.; Atzmon, Y.; Patashnik, O.; Bermano, A. H.; Chechik, G.; and Cohen-Or, D. 2022. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *NIPS*, 27.

Gu, Z.; Liu, L.; Chen, X.; Yi, R.; Zhang, J.; Wang, Y.; Wang, C.; Shu, A.; Jiang, G.; and Ma, L. 2023. Remembering Normality: Memory-guided Knowledge Distillation for Unsupervised Anomaly Detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 16401–16409.

Gudovskiy, D.; Ishizaka, S.; and Kozuka, K. 2022. Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 98–107.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Hertz, A.; Mokady, R.; Tenenbaum, J.; Aberman, K.; Pritch, Y.; and Cohen-Or, D. 2022. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NIPS*, 30.

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

Ho, J.; and Salimans, T. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

Hu, T.; Zhang, J.; Liu, L.; Yi, R.; Kou, S.; Zhu, H.; Chen, X.; Wang, Y.; Wang, C.; and Ma, L. 2023. Phasic Content Fusing Diffusion Model with Directional Distribution Consistency for Few-Shot Model Adaption. In *ICCV*, 2406–2415.

Huang, C.; Guan, H.; Jiang, A.; Zhang, Y.; Spratling, M.; and Wang, Y.-F. 2022. Registration based few-shot anomaly detection. In *ECCV*, 303–319. Springer.

Jeong, J.; Zou, Y.; Kim, T.; Zhang, D.; Ravichandran, A.; and Dabeer, O. 2023. Winclip: Zero-/few-shot anomaly classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19606–19616.

Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2020. Analyzing and improving the image quality of stylegan. In *CVPR*, 8110–8119.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Lee, S.; Lee, S.; and Song, B. C. 2022. Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. *IEEE Access*, 10: 78446–78454.

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

Li, Y.; Zhang, R.; Lu, J.; and Shechtman, E. 2020. Few-shot image generation with elastic weight consolidation. *arXiv preprint arXiv:2012.02780*.

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

致谢

本研究得到国家自然科学基金（62302297、72192821、62272447）、中国科协青年人才托举工程（2022QNRC001）、上海市扬帆计划（22YF1420300）、北京市自然科学基金（L222117）、中央高校基本科研业务费（YG2023QNB17）、上海市科技重大专项（2021SHZDZX0102）、上海市科学技术委员会（21511101200）及CCF-腾讯犀牛鸟科研基金（RAGR20220121）的资助。

参考文献

Avrahami, O.; Lischinski, D.; and Fried, O. 2022. 自然图像文本驱动编辑的混合扩散方法。于 *CVPR*，18208–18218。 Bergmann, P.; Fauser, M.; Sattlegger, D.; and Steger, C. 2019. MVTec AD——一个用于无监督异常检测的综合真实数据集。于 *CVPR*，9592–9600。 Bińkowski, M.; Sutherland, D. J.; Arbel, M.; and Gretton, A. 2018. 解密MMD生成对抗网络。*arXiv preprint arXiv:1801.01401*。 Cao, Y.; Wan, Q.; Shen, W.; and Gao, L. 2022. 面向图像异常分割的信息化知识蒸馏。*Knowledge-Based Systems*，248: 108846。 Cao, Y.; Xu, X.; Sun, C.; Cheng, Y.; Du, Z.; Gao, L.; and Shen, W. 2023. 通过混合提示正则化实现无需训练的任意异常分割。*arXiv preprint arXiv:2305.10724*。 Chen, X.; Han, Y.; and Zhang, J. 2023. 零样本/少样本异常分类与分割方法：在CVPR 2023 VAND挑战赛赛道1和2中获零样本异常检测冠军及少样本异常检测第四名。*arXiv preprint arXiv:2305.17382*。 Chen, X.; Zhang, J.; Tian, G.; He, H.; Zhang, W.; Wang, Y.; Wang, C.; Wu, Y.; and Liu, Y. 2023. CLIP-AD：一种语言引导的分阶段双路径零样本异常检测模型。*arXiv preprint arXiv:2311.00453*。 Deng, H.; and Li, X. 2022. 通过单类嵌入反向蒸馏实现异常检测。于 *CVPR*，9737–9746。 Ding, C.; Pang, G.; and Shen, C. 2022. 捕捉灰天鹅与黑天鹅：开放集监督异常检测。于 *CVPR*，7388–7398。 Duan, Y.; Hong, Y.; Niu, L.; and Zhang, L. 2023. 通过缺陷感知特征操控实现少样本缺陷图像生成。于 *AAAI*，第37卷，571–578。 Gal, R.; Alaluf, Y.; Atzmon, Y.; Patashnik, O.; Bermano, A. H.; Chechik, G.; and Cohen-Or, D. 2022. 一图值一词：通过文本反演实现文本到图像生成的个性化。*arXiv preprint arXiv:2208.01618*。 Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. 生成对抗网络。*NIPS*，27。 顾铮; 刘磊; 陈晓; 易仁; 张健; 王宇; 王超; 舒昂; 江光; 马龙. 2023. 记忆常态：基于记忆引导的无监督异常检测知识蒸馏. 于 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 16401–16409. 古多夫斯基; 石坂聪; 小冢和也. 2022. Cflow-ad：通过条件归一化流实现具有定位能力的实时无监督异常检测. 于 *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 98–107. 何恺明; 张祥雨; 任少卿; 孙剑. 2016. 深度残差学习在图像识别中的应用. 于 *CVPR*, 770–778. 赫兹; 莫卡迪; 特南鲍姆; 阿伯曼; 普里奇; 科恩-奥尔. 2022. 通过跨注意力控制实现提示到提示的图像编辑. *arXiv preprint arXiv:2208.01626*. 霍伊塞尔; 拉姆绍尔; 温特希纳; 内斯勒; 霍赫赖特. 2017. 采用双时间尺度更新规则训练的GAN收敛至局部纳什均衡. *NIPS*, 30. 何; 贾因; 阿比尔. 2020. 去噪扩散概率模型. *Advances in Neural Information Pro- cessing Systems*, 33: 6840–6851. 何; 萨利曼斯. 2022. 无分类器扩散引导. *arXiv preprint arXiv:2207.12598*. 胡涛; 张健; 刘磊; 易仁; 寇帅; 朱慧; 陈晓; 王宇; 王超; 马龙. 2023. 面向少样本自适应训练的相位内容融合扩散模型与方向分布一致性. 于 *ICCV*, 2406–2415. 黄超; 关浩; 江安; 张宇; 斯普拉特林; 王亦飞. 2022. 基于配准的少样本异常检测. 于 *ECCV*, 303–319. 斯普林格出版社. 郑镇瑄; 郭宇; 金泰亨; 张丹; 拉维钱德兰; 达比尔. 2023. Winclip：零样本/少样本异常分类与分割. 于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19606–19616. 卡拉斯莱内; 艾塔拉; 赫尔sten; 莱赫蒂宁; 艾拉. 2020. StyleGAN图像质量分析与改进. 于 *CVPR*, 8110–8119. 金马; 韦林. 2013. 变分贝叶斯自编码器. *arXiv preprint arXiv:1312.6114*. 李承烈; 李成旭; 宋秉辰. 2022. CFA：基于耦合超球面的目标导向异常定位特征自适应. *IEEE Access*, 10: 78446–78454. 李晨龙; 孙健; 尹俊; 普菲斯特. 2021. CutPaste：用于异常检测与定位的自监督学习. 于 *CVPR*, 9664–9674. 李毅; 张锐; 卢健; 谢克特曼. 2020. 基于弹性权重巩固的少样本图像生成. *arXiv preprint arXiv:2012.02780*. 梁煜; 张健; 赵帅; 吴锐; 刘洋; 潘晟. 2023. 全频段通道选择表征在无监督异常检测中的应用. *IEEE Transactions on Image Processing*.

Lin, D.; Cao, Y.; Zhu, W.; and Li, Y. 2021. Few-shot defect segmentation leveraging abundant defect-free training samples through normal background regularization and crop-and-paste operation. In *ICME*, 1–6. IEEE.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *CVPR*, 2117–2125.

Mo, S.; Cho, M.; and Shin, J. 2020. Freeze the discriminator: a simple baseline for fine-tuning gans. *arXiv preprint arXiv:2002.10964*.

Nichol, A. Q.; and Dhariwal, P. 2021. Improved denoising diffusion probabilistic models. In *ICML*, 8162–8171. PMLR.

Niu, S.; Li, B.; Wang, X.; and Lin, H. 2020. Defect image sample generation with GAN for improving defect recognition. *IEEE Transactions on Automation Science and Engineering*, 17(3): 1611–1622.

Ojha, U.; Li, Y.; Lu, J.; Efros, A. A.; Lee, Y. J.; Shechtman, E.; and Zhang, R. 2021. Few-shot image generation via cross-domain correspondence. In *CVPR*, 10743–10752.

Pang, G.; Ding, C.; Shen, C.; and Hengel, A. v. d. 2021. Explainable deep few-shot anomaly detection with deviation networks. *arXiv preprint arXiv:2108.00462*.

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

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. 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*, 234–241. Springer.

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

Ruiz, N.; Li, Y.; Jampani, V.; Pritch, Y.; Rubinstein, M.; and Aberman, K. 2023. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 22500–22510.

Schlegl, T.; Seeböck, P.; Waldstein, S. M.; Langs, G.; and Schmidt-Erfurth, U. 2019. f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks. *Medical image analysis*, 54: 30–44.

Schlegl, T.; Seeböck, P.; Waldstein, S. M.; Schmidt-Erfurth, U.; and Langs, G. 2017. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, 146–157. Springer.

Schuhmann, C.; Vencu, R.; Beaumont, R.; Kaczmarczyk, R.; Mullis, C.; Katta, A.; Coombes, T.; Jitsev, J.; and Komatsuzaki, A. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*.

Tran, N.-T.; Tran, V.-H.; Nguyen, N.-B.; Nguyen, T.-K.; and Cheung, N.-M. 2021. On data augmentation for gan training. *IEEE Transactions on Image Processing*, 30: 1882–1897.

Wang, Y.; Peng, J.; Zhang, J.; Yi, R.; Wang, Y.; and Wang, C. 2023. Multimodal Industrial Anomaly Detection via Hybrid Fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8032–8041.

Wang, Y.; Yi, R.; Tai, Y.; Wang, C.; and Ma, L. 2022. Ctl-gan: Few-shot artistic portraits generation with contrastive transfer learning. *arXiv preprint arXiv:2203.08612*.

Zavrtanik, V.; Kristan, M.; and Skočaj, D. 2021. Draem-a discriminatively trained reconstruction embedding for surface anomaly detection. In *ICCV*, 8330–8339.

Zhang, G.; Cui, K.; Hung, T.-Y.; and Lu, S. 2021. Defect-GAN: High-fidelity defect synthesis for automated defect inspection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2524–2534.

Zhang, H.; Wu, Z.; Wang, Z.; Chen, Z.; and Jiang, Y.-G. 2023a. Prototypical residual networks for anomaly detection and localization. In *CVPR*, 16281–16291.

Zhang, J.; Chen, X.; Xue, Z.; Wang, Y.; Wang, C.; and Liu, Y. 2023b. Exploring Grounding Potential of VQA-oriented GPT-4V for Zero-shot Anomaly Detection. *arXiv preprint arXiv:2311.02612*.

Zhao, S.; Liu, Z.; Lin, J.; Zhu, J.-Y.; and Han, S. 2020. Differentiable augmentation for data-efficient gan training. *NIPS*, 33: 7559–7570.

林德、曹阳、朱伟、李毅。2021。通过正常背景正则化和裁剪粘贴操作利用丰富无缺陷训练样本进行小样本缺陷分割。发表于*ICME*，第1–6页。IEEE。

林铁彦, Dollár, P.、Girshick, R.、何恺明、Haribaran, B.、Belongie, S.。2017。用于目标检测的特征金字塔网络。发表于*CVPR*，第2117–2125页。

莫晟、赵敏、申宰昊, 2020。冻结判别器：一种微调GAN的简单基线。*arXiv preprintarXiv:2002.10964*。

尼科尔, A. Q.、达里瓦尔, P.。2021。改进的去噪扩散概率模型。发表于*ICML*，第8162–8171页。PMLR。

牛帅、李博、王旭、林辉。2020。使用GAN生成缺陷图像样本以改进缺陷识别。*IEEE Transactions on Automation Science and Engineering*，第17卷第3期：第1611–1622页。

奥贾, D.、李毅、卢健、Efros, A. A.、Lee, Y. J.、Shechtman, E.、张睿。2021。通过跨域对应关系进行小样本图像生成。发表于*CVPR*，第10743–10752页。

庞国、丁超、沈晨、Hengel, A. v. d.。2021。基于偏差网络的可解释深度小样本异常检测。*arXiv preprint arXiv:2108.00462*。

龙巴赫, R.、布拉特曼, A.、洛伦兹, D.、埃瑟, P.、奥默, B.。2022。基于潜在扩散模型的高分辨率图像合成。发表于*CVPR*，第10684–10695页。

龙内伯格, O.、菲舍尔, P.、布罗克斯, T.。2015。U-Net：用于生物医学图像分割的卷积网络。发表于*Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*，第234–241页。Springer。

罗斯, K.、佩穆拉, L.、塞佩达, J.、舒尔科夫, B.、布罗克斯, T.、盖勒, P.。2022。实现工业异常检测的完全召回。发表于*CVPR*，第14318–14328页。

鲁伊斯, N.、李毅、詹帕尼, V.、普里奇, Y.、鲁宾斯坦, M.、阿伯曼, K.。2023。Dreambooth：针对主体驱动生成的文本到图像扩散模型微调。发表于*CVPR*，第22500–22510页。

施莱格尔, T.、泽博克, P.、瓦尔德斯坦, S. M.、朗斯, G.、施密特-埃尔富特, U.。2019。f-AnoGAN：基于生成对抗网络的快速无监督异常检测。*Medical image analysis*，第54卷：第30–44页。

施莱格尔, T.、泽博克, P.、瓦尔德斯坦, S. M.、施密特-埃尔富特, U.、朗斯, G.。2017。利用生成对抗网络进行无监督异常检测以指导标记发现。发表于*International conference on information processing in medical imaging*，第146–157页。Springer。

舒曼, C.、文库, R.、博蒙特, R.、卡兹马契克, R.、穆利斯, C.、卡塔, A.、库姆斯, T.、吉采夫, J.、小松崎, A.。2021。Laion-400m：包含4亿个图文对的CLIP过滤开源数据集。*arXiv preprintarXiv:2111.02114*。

陈南胜、陈文鸿、阮文伯、阮德功、张南明。2021。论GAN训练中的数据增强。*IEEE Transactions on Image Processing*，第30卷：第1882–1897页。

王, Y.; 彭, J.; 张, J.; 易, R.; 王, Y.; 与 王, C. 2023. 基于混合融合的多模态工业异常检测。于*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8032–8041.

王, Y.; 易, R.; 邰, Y.; 王, C.; 与 马, L. 2022. Ctl-gan: 基于对比迁移学习的少样本艺术肖像生成。*arXiv preprint arXiv:2203.08612*.

扎夫尔塔尼克, V.; 克里斯蒂安, M.; 与 斯科卡伊, D. 2021. Draem——一种判别性训练的重构嵌入表面异常检测方法。于 *ICCV*, 8330–8339.

张, G.; 崔, K.; 洪, T.-Y.; 与 陆, S. 2021. Defect-GAN: 用于自动缺陷检测的高保真缺陷合成。于*Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2524–2534.

张 H.; 吴, Z.; 王, Z.; 陈, Z.; 与 蒋, Y.-G. 2023a. 用于异常检测与定位的原型残差网络。于 *CVPR*, 16281–16291.

张, X.; 陈, X.; 薛, Z.; 王, Y.; 王, C.; 与 刘, Y. 2023b. 探索面向VQA的GPT-4V在零样本异常检测中的接地潜力。*arXiv preprint arXiv:2311.02612*.

赵, S.; 刘, Z.; 林, J.; 朱, J.-Y.; 与 韩, S. 2020. 数据高效GAN训练的可微分增广。*NIPS*, 33: 7559–7570.

A Overview

This supplementary material consists of:

- Details of the data augmentation method (Sec. B).
- More implementation details (Sec. C).
- More ablation studies (Sec. D).
- Comparison between our Spatial Anomaly Embedding and Prompt-to-Prompt (Sec. E).
- More qualitative comparison results with the anomaly generation methods (Sec. F).
- More quantitative comparison results with the anomaly generation methods (Sec. G).

B Data Augmentation

Due to the limited number of samples for each anomaly category, typically less than 10 images are available for training. This constraint makes it challenging for our anomaly embedding to completely eliminate spatial information, as it still tends to generate anomalies at the positions observed in the training images. Additionally, when the training data for the spatial encoder is scarce, the model becomes susceptible to overfitting, making it difficult to accurately generate anomalies at the correct positions.

To address these issues, we employ a data augmentation approach during training. For paired image-mask data, we perform random cropping, translation, and rotation on both the image and its corresponding mask. By recording the maximum and minimum coordinates of the anomaly region in the image, we ensure that the anomaly remains within the image during data augmentation. This data augmentation process effectively disrupts the spatial information within the training data, causing the anomaly embedding to lose its focus on recording positions and instead concentrate solely on the anomaly appearance. Simultaneously, the spatial encoder benefits from having enough augmented data for training, boosting its ability in position encoding.

C Implementation Details

C.1 Training Details

Training spatial anomaly embedding. For each anomaly type, an anomaly embedding e_a is assigned, while a shared spatial encoder E is employed across all anomaly categories. Each anomaly embedding e_a is composed of 8 tokens, and the spatial embedding e_s comprises 4 tokens. The batch size is set at 4, and the learning rate is 0.005. During each training iteration, we randomly sample 4 anomalous image-mask pairs from all the anomaly categories. We train all the anomaly embedding and spatial encoder at the same time for 300K iterations in 3 days on an NVIDIA GeForce RTX 3090 24GB GPU.

Training mask embedding. For each anomaly type, we assign a mask embedding e_m for it. To enhance the diversity of the generated masks, each mask embedding consists of 2 tokens, preventing it from overfitting. Furthermore, with a batch size of 4 and a learning rate of 0.005, each mask embedding is trained for 30Kiterations.

Mask Generation. With the trained mask embedding, we input it as a text condition to guide the generation process of latent diffusion model (Rombach et al. 2022). Specifically, we employ the classifier-free guidance (Ho and Salimans 2022) to generate masks:

$$\hat{\epsilon}_\theta\left(x_t \mid e_m\right) = \epsilon_\theta\left(x_t\right) + s \cdot \left(\epsilon_\theta\left(x_t, e_m\right) - \epsilon_\theta\left(x_t\right)\right), \quad (10)$$

where s is set 5 (the same as Textual Inversion).

C.2 Metrics

In the quantitative experiments, we employ the following metrics to measure the model performance.

- **Inception Score(IS)** quantifies the quality and diversity of generated images by computing the exponential of the negative of the KL divergence between the marginal distribution of generated images and the conditional distribution of class labels predicted by an Inception model. A higher IS score represents a better generation quality and diversity.
- **Intra-cluster Pairwise LPIPS Distance (IC-LPIPIS)** (Ojha et al. 2021) clusters the generated images into k groups based on LPIPS distance to k target samples, and then compute the average mean LPIPS distances to corresponding target samples within each cluster. A higher IC-LPIPS indicates a better generation diversity.
- **Area Under the Receiver Operating Characteristic (AUROC)** measures the performance of a binary classification model by evaluating its ability to distinguish between true positive and false positive rates across different probability thresholds. A higher AUROC means better anomaly detection and localization performance.
- **Average Precision (AP, which is also known as PR-AUC)** assesses the precision-recall curve for a classification model, calculating the average precision of the model across different recall levels, providing a summary of its overall performance. A higher AP means better anomaly detection and localization results.
- $\mathbf{F_1}$**-max** is a variant of the F_1 score that maximizes both precision and recall by selecting the threshold that yields the highest F1 score when evaluating a binary classification model. A higher $\mathbf{F_1}$-max represents better anomaly detection and localization results

D More Ablation Studies

D.1 Ablation on Spatial Anomaly Embedding

We aim to seek a text embedding that guides the latent diffusion model in generating anomalies within a given anomaly mask. However, textual inversion tends to capture the location of anomalies along with the anomaly type information, which results in the generated anomalies only distributed in specific locations. Therefore, we propose spatial anomaly embedding e, consisting of an anomaly embedding e_a (for appearance) and a spatial embedding e_s (for location), which disentangles the spatial information from anomaly appearance. To further validate this theory, we directly employ

A 概述

本补充材料包括:

- 数据增强方法的详情（附录 B）。
- 更多实现细节（见附录 C）。
- 更多消融研究（见附录 D）。
- 我们的空间异常嵌入与Prompt-to-Prompt（附录E）之间的比较。
- 与异常生成方法（第F节）的更多定性比较结果。
- 与异常生成方法（第G节）的更多定量比较结果。

B 数据增强

由于每个异常类别的样本数量有限，通常只有不到10张图像可用于训练。这一限制使得我们的异常嵌入难以完全消除空间信息，因为它仍然倾向于在训练图像中观察到的位置生成异常。此外，当空间编码器的训练数据稀缺时，模型容易过拟合，导致难以在正确位置准确生成异常。

为解决上述问题，我们在训练过程中采用了数据增强方法。对于配对的图像-掩码数据，我们对图像及其对应掩码同时进行随机裁剪、平移和旋转操作。通过记录图像中异常区域的最大最小坐标，我们确保数据增强过程中异常区域始终保留在图像范围内。这种数据增强处理有效破坏了训练数据中的空间信息，使得异常嵌入不再专注于记录位置信息，而是集中关注异常外观特征。与此同时，空间编码器因获得足够多的增强训练数据而受益，从而提升了其位置编码能力。

C 实现细节

C.1 训练细节

训练空间异常嵌入。针对每种异常类型，会分配一个异常嵌入e_a，而所有异常类别共享一个空间编码器E。每个异常嵌入e_a由8个标记组成，空间嵌入e_s包含4个标记。批处理大小设置为4，学习率为0.005。在每个训练迭代中，我们从所有异常类别中随机采样4个异常图像-掩码对。我们在NVIDIA GeForce RTX 3090 24GB GPU上同时训练所有异常嵌入和空间编码器，持续3天完成30万次迭代。

训练掩码嵌入。对于每种异常类型，我们为其分配一个掩码嵌入e_m。为了增强生成掩码的多样性，每个掩码嵌入由2个标记组成，以防止过拟合。此外，在批大小为4、学习率为0.005的条件下，每个掩码嵌入训练30K次迭代。

掩码生成。利用训练好的掩码嵌入，我们将其作为文本条件输入，以指导潜在扩散模型（Rombach等人，2022年）的生成过程。具体而言，我们采用无分类器引导技术（Ho和Salimans，2022年）来生成掩码:

$$\hat{\epsilon}_\theta\left(x_t \mid e_m\right)=\epsilon_\theta\left(x_t\right)+s \cdot\left(\epsilon_\theta\left(x_t, e_m\right)-\epsilon_\theta\left(x_t\right)\right), \quad (10)$$

其中 s 设为 5（与 Textual Inversion 相同）。

C.2 指标

在定量实验中，我们采用以下指标来衡量模型性能。

- 初始分数(IS)通过计算生成图像的边缘分布与Inception模型预测的类别标签条件分布之间KL散度负值的指数，来量化生成图像的质量和多样性。IS分数越高，代表生成质量与多样性越佳。
- 簇内成对LPIPS距离（IC-LPIPS）（Ojha等人2021）基于与k个目标样本的LPIPS距离将生成图像聚类为k个组，随后计算各聚类内与对应目标样本的平均LPIPS距离。IC-LPIPS值越高，表明生成多样性越佳。
- 接收者操作特征曲线下面积（AUROC）通过评估二元分类模型在不同概率阈值下区分真阳性率和假阳性率的能力，来衡量其性能。AUROC值越高，表示异常检测和定位性能越好。
- 平均精度（AP，也称为PR-AUC）评估分类模型的精确率-召回率曲线，通过计算模型在不同召回率水平下的平均精度，来综合衡量其整体性能。AP值越高，表明异常检测和定位的效果越好。
- $\mathbf{F_1}$-max 是 F_1 分数的一个变体，它通过选择在评估二元分类模型时能产生最高 F1 分数的阈值，来同时最大化精确率和召回率。更高的 $\mathbf{F_1}$-max 值代表更好的异常检测和定位结果。

D 更多消融研究

D.1 空间异常嵌入消融实验

我们的目标是寻找一种文本嵌入，能够引导潜在扩散模型在给定的异常掩码内生成异常。然而，文本反转往往会同时捕获异常位置与异常类型信息，导致生成的异常仅分布在特定区域。因此，我们提出空间异常嵌入e，其中包含表征外观的异常嵌入e_a ()和表征位置的空间嵌入e_s ()，从而将空间信息从异常外观中解耦。为进一步验证该理论，我们直接采用

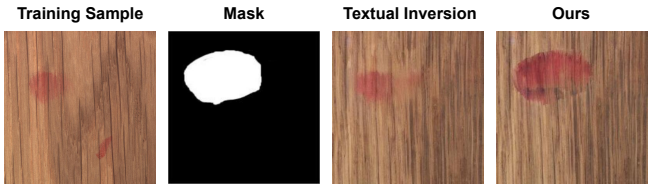

Figure 6: **Comparison results between Textual Inversion (Anomaly Embedding only) and our model (Spatial Anomaly Embedding).** The generated result of Textual Inversion tends to generate anomalies at the location as the same as the training sample.

Anomaly Rate	10%	20%	**30% (Ours)**	40%	50%
AUROC ↑	96.2	98.1	**99.1**	99.0	98.7
AP ↑	64.2	75.5	**81.4**	81.1	80.0

Table 7: **ablation study on the rate of anomalies.**

text inversion to train an anomaly embedding and use the trained embedding to generate anomalies with a given mask by blended latent diffusion (the same as our generation process). The generated results are shown in Fig. 6. It can be seen that the generated result by Textual Inversion tends to generate anomalies at the location the same as the training sample, which limits its application in anomaly generation where anomalies can be located at arbitrary positions.

D.2 Ablation on the rate of anomalies

We conduct additional experiments with the rate of anomalies as 10%, 20%, 30% (**Ours**), 40%, and 50% and test the performance on anomaly localization measured by AUROC and AP. The performance of anomaly localization is shown in Tab. 7. It can be seen that AP decreases quickly when the anomaly rate falls below 30%. This is attributed to the limited availability of training data for most categories, often comprising only 1-2 instances, making it challenging for the model to capture the anomaly information. Conversely, when the rate exceeds 30%, the model performance is similar. This indicates that our model can effectively learn sufficient anomaly information without a heavy reliance on an abundance of training samples.

D.3 Ablation on the hyperparameters

We conduct ablation studies on the length of the anomaly embedding l_a and spatial embedding l_s in Tab. 8. Specifically, we train models with different l_a and l_s and then employ their generated data to train a UNet to localize the anomalies. It can be seen that when increasing l_s and l_a, the total parameter number of the model rises, but the final performance in the downstream anomaly localization task is similar, which demonstrates that our model is not snesitive to the hyperparameters.

D.4 Ablation on SAE and AAR

We conduct more ablation studies on the effectiveness of our spatial anomaly embedding (SAE) and adaptive attention re-weighting mechanism (AAR) by adding SAE and AAR separately. We compare our model with 3 models: ***1)*** w/o AAR&SAE, ***2)*** AAR only, and ***3)*** SAE only in generating glue anomalies to the leather in the Fig. 7. It shows that the model without AAR&SAE cannot generate authentic anomalies or fill anomaly mask. While adding SAE improves anomaly authenticity, it doesn't fill the mask. Moreover, incorporating AAR fills the mask but sacrifices authenticity. In contrast, our model (SAE + AAR) effectively generates authentic anomalies filling the mask.

Model	AUROC ↑	AP ↑	F_1-max ↑	PRO ↑
$L_s = 4, l_a = 8$ (**Ours**)	**99.1**	**81.4**	**76.3**	**94.0**
$L_s = 4, l_a = 16$	98.8	80.6	75.1	93.2
$L_s = 8, l_a = 8$	99.0	80.9	75.8	93.5
$L_s = 8, l_a = 16$	**99.1**	81.2	75.9	93.8

Table 8: **Ablation study on the anomaly embedding l_a and spatial embedding l_s**

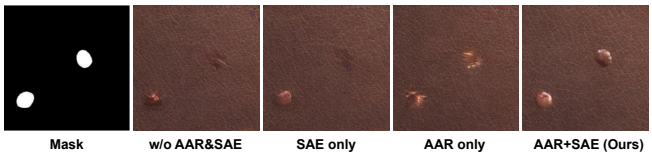

Figure 7: **Ablation study on SAE and AAR.**

E Comparison with Prompt-to-Prompt

Prompt-to-Prompt (Hertz et al. 2022) proposed a method that allows modifying generated images by altering corresponding text descriptions. For instance, when transforming "a cat sits on the street" to "a dog sits on the street," Prompt-to-Prompt replaces the cross-attention map of "dog" with that of "cat", which transforms the cat in the original image into a dog while maintaining nearly unchanged content in other regions, achieving controlled image generation with specific positions. However, Prompt-to-Prompt requires a text corresponding to the original image for generation, which is unavailable in anomaly generation.

It seems that Prompt-to-prompt presents a potential solution for controlling generation positions through the manipulation of cross-attention maps. A direct solution is to resize the mask m to match and substitute the cross-attention map m_c of anomaly embedding, thus controlling the generation location. However, even though the new cross-attention map m_c' seemingly dictates anomaly location, it could conflict with the values V in the cross-attention module. Since V is designed for the original cross-attention map m_c, the semantic information of V in the newly enforced mask m_c' might not align with the semantics in original mask m_c, consequently leading to a unstable generated results.

To verify it, we conduct reconstruction experiments on real anomalies, comparing the results of Textual Inversion + Prompt-to-Prompt with our approach (Spatial Anomaly Embedding). Specifically, we sample a real anomaly image I as ground truth and mask out the anomaly parts for generation. The results are shown in Fig. 8. Textual Inversion + Prompt-to-Prompt can not generate anomalies as authentic

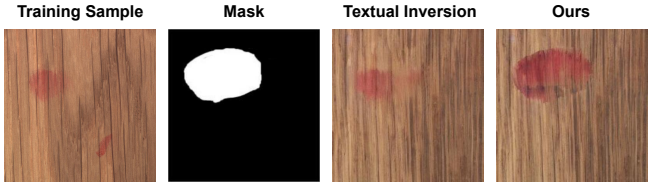

图6：文本反转（仅异常嵌入）与我们模型（空间异常嵌入）的对比结果。文本反转的生成结果倾向于在训练样本相同位置生成异常。

Anomaly Rate	10%	20%	**30% (Ours)**	40%	50%
AUROC ↑	96.2	98.1	**99.1**	99.0	98.7
AP ↑	64.2	75.5	**81.4**	81.1	80.0

表7：异常率消融研究。

文本反转用于训练异常嵌入，并利用训练后的嵌入通过混合潜在扩散（与我们的生成过程相同）在给定掩码下生成异常。生成结果如图6所示。可以看出，文本反转生成的异常倾向于出现在与训练样本相同的位置，这限制了其在异常位置可任意设定的异常生成中的应用。

D.2 异常率消融实验

我们在异常比例为10%、20%、30%（本方法）、40%和50%的条件下进行了补充实验，并通过AUROC和AP指标评估异常定位性能。异常定位性能如表7所示。可以看出当异常比例低于30%时，AP值会快速下降。这主要是因为大多数类别的训练数据有限（通常仅含1-2个样本），导致模型难以捕捉异常信息。相反，当异常比例超过30%时，模型表现趋于稳定。这表明我们的模型能够有效学习到足够的异常信息，而无需过度依赖大量训练样本。

D.3 超参数消融实验

我们在表格8中对异常嵌入l_a和空间嵌入l_s的长度进行了消融实验。具体而言，我们使用不同的l_a和l_s训练模型，然后利用它们生成的数据训练UNet来定位异常。可以看出，当增加l_s和l_a时，模型的总参数量会上升，但在下游异常定位任务中的最终性能表现相近，这表明我们的模型对超参数不敏感。

D.4 SAE与AAR的消融实验

我们通过添加空间异常嵌入（SAE）和自适应注意力重加权机制（AAR），对SAE和AAR的有效性进行了更多的消融实验。

Model	AUROC ↑	AP ↑	F_1-max ↑	PRO ↑
$L_s = 4, l_a = 8$ (**Ours**)	**99.1**	**81.4**	**76.3**	**94.0**
$L_s = 4, l_a = 16$	98.8	80.6	75.1	93.2
$L_s = 8, l_a = 8$	99.0	80.9	75.8	93.5
$L_s = 8, l_a = 16$	**99.1**	81.2	75.9	93.8

表8：关于异常嵌入l_a和空间嵌入l_s的消融研究

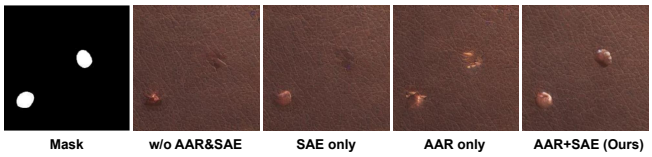

图 7：关于 SAE 和 AAR 的消融研究。

我们分别对AAR进行了测试。在图7中生成皮革的胶水异常时，将我们的模型与3个模型进行比较：不含AAR&SAE的***1)***、仅含AAR的***2)***以及仅含SAE的***3)***。实验表明，不含AAR&SAE的模型无法生成真实异常或填充异常掩码。虽然添加SAE能提升异常真实性，但仍无法填充掩码。此外，引入AAR能填充掩码却会牺牲真实性。相比之下，我们的模型（SAE + AAR）能有效生成填充掩码的真实异常。

E 与 Prompt-to-Prompt 的对比

Prompt-to-Prompt（Hertz等人，2022）提出了一种通过修改对应文本描述来调整生成图像的方法。例如，当将“一只猫坐在街上”转换为“一只狗坐在街上”时，该方法用“猫”的交叉注意力图替换“狗”的对应图，从而将原始图像中的猫转换为狗，同时保持其他区域内容几乎不变，实现了特定位置的可控图像生成。然而，Prompt-to-Prompt需要原始图像对应的文本进行生成，而这在异常生成场景中无法获取。

Prompt-to-prompt方法似乎提供了一种通过操控交叉注意力图来控制生成位置的潜在解决方案。直接解决方案是将掩码m调整大小以匹配并替换异常嵌入的交叉注意力图m_c，从而控制生成位置。然而，即使新的交叉注意力图m_c'看似能指定异常位置，它仍可能与交叉注意力模块中的数值V产生冲突。由于V是为原始交叉注意力图m_c设计的，新强制应用的掩码m_c'中V的语义信息可能与原始掩码m_c的语义不一致，最终导致生成结果不稳定。

为验证这一点，我们在真实异常图像上进行了重建实验，将Textual Inversion {v*}与Prompt-to-Prompt方法的结果与我们提出的空间异常嵌入方法进行对比。具体而言，我们采样真实异常图像{v*}作为基准真值，并遮挡异常部分进行生成。结果如图8所示，Textual Inversion {v*}与Prompt-to-Prompt方法无法生成如真实异常般逼真的效果。

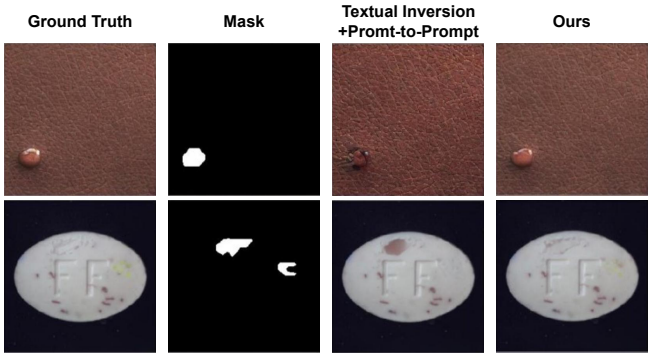

Figure 8: **Comparison results between Textual Inversion + Prompt-to-Prompt and our model (Spatial Anomaly Embedding) in anomaly generation.**

Method	Metric			
	AUROC	AP	F_1-max	PRO
Textual Inversion+ Prompt-to-Prompt	91.2	55.1	64.4	73.5
Ours	**99.1**	**81.4**	**76.4**	**94.0**

Table 9: **Comparison with Textual Inversion + Prompt-to-Prompt on anomaly localization.**

as ours. And its generated anomalies are quite different from the ground truth, indicating that replacing the cross-attention map directly cannot generate satisfying anomalies. Moreover, we also conduct a quantitative experiment, where we generate anomalous image-mask pairs to support the downstream anomaly localization task. We follow the experiment settings in the main paper, in which we train an U-Net on the generated data and compare the localization accuracy. The results are recorded in Tab. 9. Our model outperforms Textual Inversion + Prompt-to-Prompt significantly.

F More qualitative experiments

We give a more comprehensive comparison with the existing anomaly generation methods DiffAug (Zhao et al. 2020), CDC (Ojha et al. 2021), Crop&Paste, SDGAN (Niu et al. 2020), Defect-GAN (Zhang et al. 2021) and DFMGAN (Duan et al. 2023). We exhibit the generation results of all the anomaly generation methods across all components in Fig. 9. Our model demonstrates remarkable proficiency in generating high-quality, authentic anomalies that are precisely aligned with the corresponding masks. In contrast, Crop&Paste exhibits limited diversity in generating various anomaly types. DiffAug displays evident overfitting tendencies towards the training samples (the image in the lower-right corner). CDC yields visually perplexing results, particularly for structural anomaly categories like capsule-squeeze. SDGAN and DefectGAN yield poor outputs, frequently encountering challenges in generating anomalies such as pill-crack. The state-of-the-art model DFMGAN occasionally struggles to create authentic anomalies and fails in maintaining alignment between the generated anomalies and masks, as observed in the case of metal nut-bent. In comparison, our model generates anomalies with the highest diversity and authenticity, and the generated anomalies align with the masks accurately, which can effectively support the downstream anomaly inspection tasks.

G More Quantitative experiments

More comparison with the anomaly generation models. In this section, we provide supplementary experiments to complement those presented in the main paper. Specifically, in addition to the methods covered in the main paper, we include Crop&Paste (Lin et al. 2021) for comparison and we additionally introduce the Per Region Overlap (PRO) metric to provide a more comprehensive evaluation on anomaly localization. The experiment settings are the same as that in the main paper, where we train an U-net on the generated anomaly data. The pixel-level anomaly localization results are shown in Tab. 10 and the image-level anomaly detection results are shown in Tab. 11. The quantitative results demonstrate that our model outperforms all the other anomaly generation methods in terms of both anomaly localization and detection, indicating our good anomaly generation quality and diversity.

More comparison with the anomaly localization models. In this section, we further compare the anomaly detection methods with F_1-max score on anomaly localization. The results are shown in Table 12. It can be seen that our model achieves the best performance in anomaly localization with F_1-max.

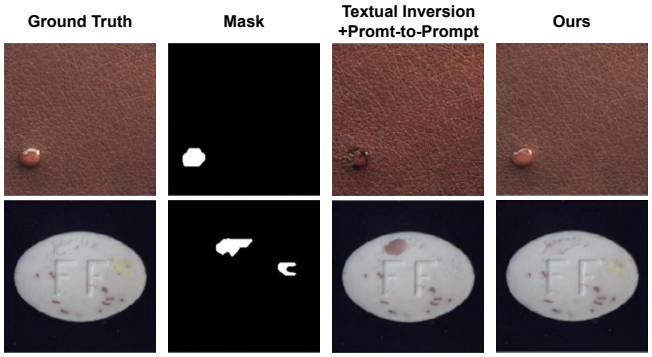

图8：文本反转{v*}、Prompt-to-Prompt与我们的模型（空间异常嵌入）在异常生成方面的对比结果。

Method	Metric			
	AUROC	AP	F_1-max	PRO
Textual Inversion+ Prompt-to-Prompt	91.2	55.1	64.4	73.5
Ours	**99.1**	**81.4**	**76.4**	**94.0**

表9：与Textual Inversion + Prompt-to-Prompt在异常定位上的对比

与我们的方法相同。其生成的异常与真实情况大相径庭，这表明直接替换交叉注意力图无法生成令人满意的异常。此外，我们还进行了定量实验，通过生成异常图像-掩码对来支持下游异常定位任务。我们遵循主论文中的实验设置，在生成数据上训练U-Net并比较定位精度。结果记录在表9中。我们的模型显著优于Textual Inversion +和Prompt-to-Prompt方法。

F 更多定性实验

我们与现有的异常生成方法进行了更全面的比较，包括DiffAug（Zhao等人，2020）、CDC（Ojha等人，2021）、Crop&Paste、SDGAN（Niu等人，2020）、Defect-GAN（Zhang等人，2021）和DFM-GAN（Duan等人，2023）。图9展示了所有异常生成方法在各组件上的生成结果。我们的模型在生成高质量、真实且与对应掩码精确对齐的异常方面表现出卓越能力。相比之下，Crop&Paste在生成多样化异常类型时表现有限；DiffAug对训练样本（右下角图像）表现出明显过拟合倾向；CDC会产生视觉上令人困惑的结果，尤其对胶囊挤压等结构异常类别；SDGAN和DefectGAN生成效果较差，在生成药片裂纹等异常时经常遇到困难；当前最先进的DFMGAN模型有时难以创建真实异常，且无法保持生成异常与掩码的对齐关系。

与金属螺母弯曲案例中观察到的情况和掩码相比，我们的模型生成的异常具有最高的多样性和真实性，且生成的异常与掩码精确吻合，能有效支持下游异常检测任务。

G 更多定量实验

与异常生成模型的进一步比较。在本节中，我们提供了补充实验以完善主论文中的内容。具体而言，除主论文已涵盖的方法外，我们额外引入Crop&Paste（Lin等人2021）进行对比，并首次采用区域重叠度（PRO）指标以更全面评估异常定位性能。实验设置与主论文保持一致，即在生成的异常数据上训练U-net网络。像素级异常定位结果展示在表10中，图像级异常检测结果呈现在表11中。定量结果表明，我们的模型在异常定位和检测方面均优于其他所有异常生成方法，这印证了我们生成异常数据具备优良的质量与多样性。

与异常定位模型的更多比较。在本节中，我们进一步比较了异常检测方法在异常定位任务中使用F_1-max得分的情况。结果如表12所示。可以看出，我们的模型在使用F_1-max进行异常定位时取得了最佳性能。

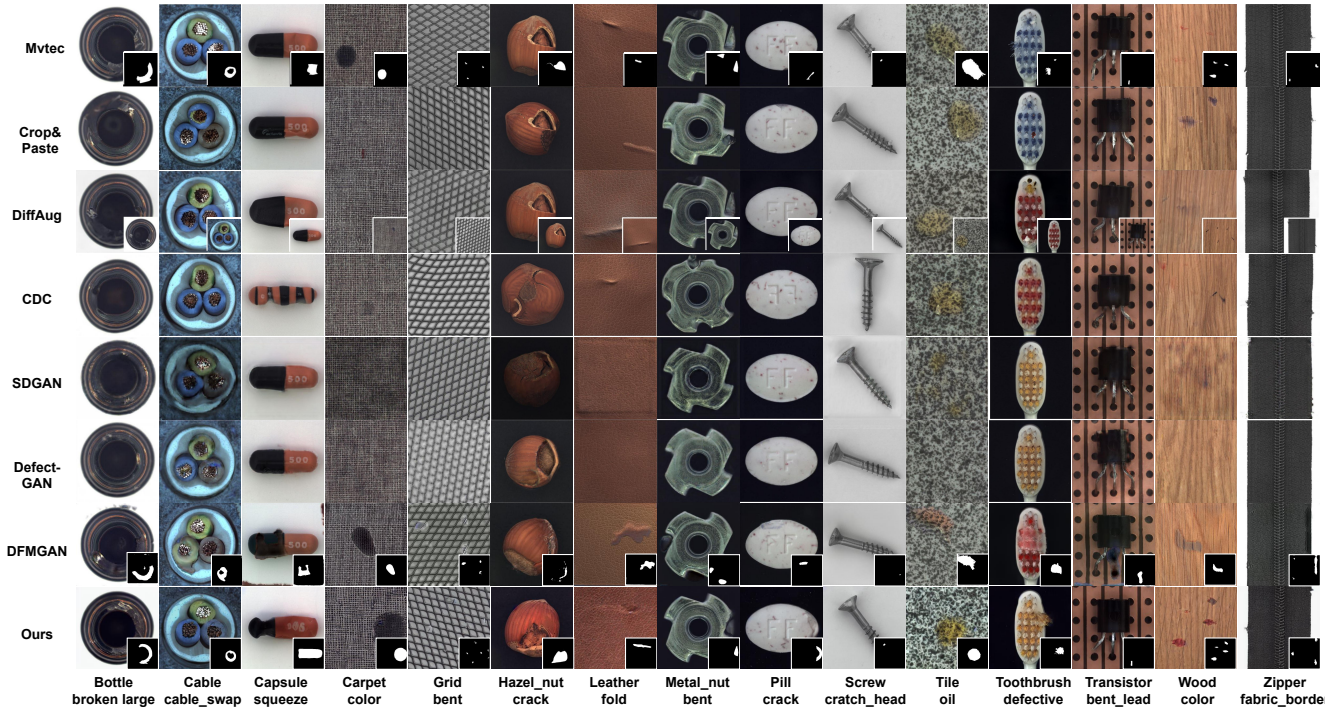

Figure 9: **Qualitative comparison on the anomaly generation quality.** Note that the generated anomlalies by DiffAug is the same as the training samples (images in lower-right corner)

Category	Crop&Paste				DRAEM				PRN				DFMGAN				Ours			
	AUC	AP	F_1-max	PRO	AUC	AP	F_1-max	PRO	AUC	AP	F_1-max	PRO	AUC	AP	F_1-max	PRO	AUC	AP	F_1-max	PRO
bottle	94.5	67.4	63.5	77.8	96.7	80.2	74.0	91.2	97.5	76.4	71.3	88.5	98.9	90.2	83.9	91.7	**99.4**	**94.1**	**87.3**	**94.3**
cable	96.0	75.3	69.3	87.1	80.3	21.8	28.3	58.2	94.5	64.4	61.0	79.7	97.2	81.0	75.4	84.9	**99.2**	**90.8**	**83.5**	**95.0**
capsule	95.3	49.2	51.1	89.5	76.2	25.5	32.1	81.1	95.6	45.7	47.9	89.7	79.2	26.0	35.0	66.1	**98.8**	**57.2**	**59.8**	**95.4**
carpet	83.7	36.6	39.7	62.9	92.6	43.0	41.9	80.0	96.4	69.6	65.6	90.6	90.6	33.4	38.1	76.5	**98.6**	**81.2**	**74.6**	**91.6**
grid	84.7	13.1	22.4	70.2	**99.1**	**59.3**	58.7	**95.8**	98.9	58.6	**58.9**	**95.8**	75.2	14.3	20.5	52.3	98.3	52.9	54.6	92.3
hazelnut	88.5	38.0	42.8	74.1	98.8	73.6	68.5	95.9	98.0	73.9	68.2	92.7	99.7	95.2	89.5	96.4	**99.8**	**96.5**	**90.6**	**97.1**
leather	97.5	76.0	70.8	95.7	98.5	67.6	65.0	96.7	99.4	58.1	54.0	97.5	98.5	68.7	66.7	96.0	**99.8**	**79.6**	**71.0**	**98.2**
metal nut	96.3	84.2	74.0	67.2	96.9	84.2	74.5	90.4	97.9	93.0	87.1	85.0	99.3	98.1	**94.5**	88.0	**99.8**	**98.7**	94.0	**94.8**
pill	81.5	17.8	24.3	57.4	95.8	45.3	53.0	83.7	98.3	55.5	72.6	88.2	81.2	67.8	72.6	56.5	**99.8**	**97.0**	**90.8**	**97.3**
screw	93.4	31.2	36.0	**83.9**	91.0	30.1	35.7	78.1	94.0	47.7	49.8	83.8	58.8	2.2	5.3	41.8	**97.0**	**51.8**	**50.9**	80.3
tile	94.0	79.3	74.5	79.2	98.5	93.2	87.8	95.3	98.5	91.8	84.4	91.3	**99.5**	**97.1**	**91.6**	**97.5**	99.2	93.9	86.2	96.1
toothbrush	89.3	30.9	34.6	66.6	93.8	29.5	28.4	75.1	96.1	46.4	46.2	83.1	96.4	75.9	72.6	74.3	**99.2**	**76.5**	**73.4**	**91.4**
transistor	85.9	52.5	52.1	64.5	76.5	31.7	24.2	54.3	94.9	68.6	68.4	70.0	96.2	81.2	77.0	65.5	**99.3**	**92.6**	**85.7**	**96.2**
wood	84.0	45.7	48.0	57.9	98.8	**87.8**	**80.9**	**94.7**	96.2	74.2	67.4	82.1	95.3	70.7	65.8	89.9	**98.9**	84.6	74.5	94.3
zipper	94.8	47.6	51.4	83.4	93.4	65.4	64.7	84.6	98.4	79.0	73.7	93.7	92.9	65.6	64.9	83.0	**99.4**	**86.0**	**79.2**	**96.3**
Average	90.4	48.4	49.4	74.3	92.2	54.1	53.1	83.1	96.9	66.2	64.7	87.4	90.0	62.7	62.1	76.3	**99.1**	**81.4**	**76.3**	**94.0**

Table 10: **Comparison on the pixel-level anomaly localization** with AUC, AP, F_1-max and PRO metrics by training an U-Net on the generated datasets produced by Crop&Paste, DRAEM, PRN, DFMGAN and our model. **Bold** and underline represent optimal and sub-optimal results, respectively.

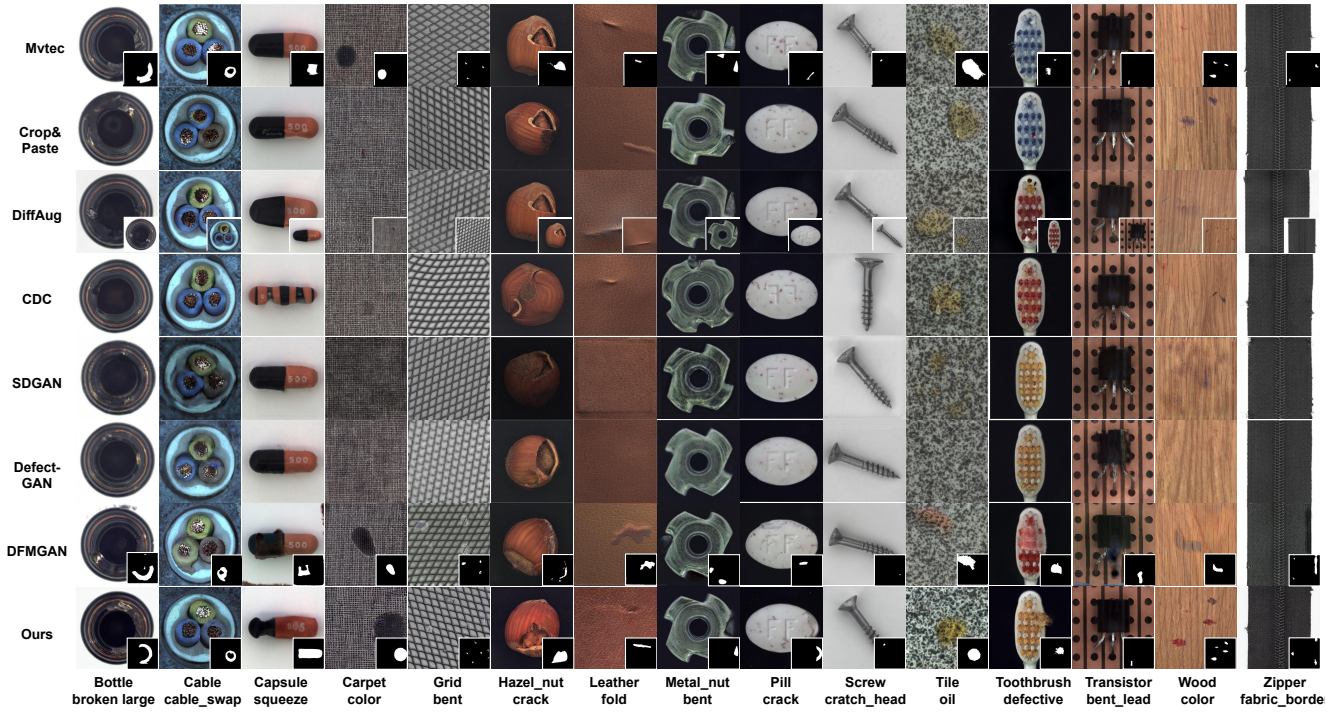

图9：异常生成质量的定性比较。请注意，DiffAug生成的异常具有 他与训练样本相同（右下角图像）

Category	Crop&Paste				DRAEM				PRN				DFMGAN				Ours			
	AUC	AP	F_1-max	PRO	AUC	AP	F_1-max	PRO	AUC	AP	F_1-max	PRO	AUC	AP	F_1-max	PRO	AUC	AP	F_1-max	PRO
bottle	94.5	67.4	63.5	77.8	96.7	80.2	74.0	91.2	97.5	76.4	71.3	88.5	98.9	90.2	83.9	91.7	**99.4**	**94.1**	**87.3**	**94.3**
cable	96.0	75.3	69.3	87.1	80.3	21.8	28.3	58.2	94.5	64.4	61.0	79.7	97.2	81.0	75.4	84.9	**99.2**	**90.8**	**83.5**	**95.0**
capsule	95.3	49.2	51.1	89.5	76.2	25.5	32.1	81.1	95.6	45.7	47.9	89.7	79.2	26.0	35.0	66.1	**98.8**	**57.2**	**59.8**	**95.4**
carpet	83.7	36.6	39.7	62.9	92.6	43.0	41.9	80.0	96.4	69.6	65.6	90.6	90.6	33.4	38.1	76.5	**98.6**	**81.2**	**74.6**	**91.6**
grid	84.7	13.1	22.4	70.2	**99.1**	**59.3**	58.7	**95.8**	98.9	58.6	**58.9**	**95.8**	75.2	14.3	20.5	52.3	98.3	52.9	54.6	92.3
hazelnut	88.5	38.0	42.8	74.1	98.8	73.6	68.5	95.9	98.0	73.9	68.2	92.7	99.7	95.2	89.5	96.4	**99.8**	**96.5**	**90.6**	**97.1**
leather	97.5	76.0	70.8	95.7	98.5	67.6	65.0	96.7	99.4	58.1	54.0	97.5	98.5	68.7	66.7	96.0	**99.8**	**79.6**	**71.0**	**98.2**
metal nut	96.3	84.2	74.0	67.2	96.9	84.2	74.5	90.4	97.9	93.0	87.1	85.0	99.3	98.1	**94.5**	88.0	**99.8**	**98.7**	94.0	**94.8**
pill	81.5	17.8	24.3	57.4	95.8	45.3	53.0	83.7	98.3	55.5	72.6	88.2	81.2	67.8	72.6	56.5	**99.8**	**97.0**	**90.8**	**97.3**
screw	93.4	31.2	36.0	**83.9**	91.0	30.1	35.7	78.1	94.0	47.7	49.8	83.8	58.8	2.2	5.3	41.8	**97.0**	**51.8**	**50.9**	80.3
tile	94.0	79.3	74.5	79.2	98.5	93.2	87.8	95.3	98.5	91.8	84.4	91.3	**99.5**	**97.1**	**91.6**	**97.5**	99.2	93.9	86.2	96.1
toothbrush	89.3	30.9	34.6	66.6	93.8	29.5	28.4	75.1	96.1	46.4	46.2	83.1	96.4	75.9	72.6	74.3	**99.2**	**76.5**	**73.4**	**91.4**
transistor	85.9	52.5	52.1	64.5	76.5	31.7	24.2	54.3	94.9	68.6	68.4	70.0	96.2	81.2	77.0	65.5	**99.3**	**92.6**	**85.7**	**96.2**
wood	84.0	45.7	48.0	57.9	98.8	**87.8**	**80.9**	**94.7**	96.2	74.2	67.4	82.1	95.3	70.7	65.8	89.9	**98.9**	84.6	74.5	94.3
zipper	94.8	47.6	51.4	83.4	93.4	65.4	64.7	84.6	98.4	79.0	73.7	93.7	92.9	65.6	64.9	83.0	**99.4**	**86.0**	**79.2**	**96.3**
Average	90.4	48.4	49.4	74.3	92.2	54.1	53.1	83.1	96.9	66.2	64.7	87.4	90.0	62.7	62.1	76.3	**99.1**	**81.4**	**76.3**	**94.0**

表10：通过在使用Crop&Paste、DRAEM、PRN、DFMGAN及本模型生成的数据集上训练U-Net，在像素级异常定位任务中采用AUC、AP、F_1-max和PRO指标的对比结果。粗体与下划线分别表示最优与次优结果。

Category	Crop&Paste			DRAEM			PRN			DFMGAN			Ours		
	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max
bottle	85.4	95.1	90.9	99.3	99.8	**98.9**	94.9	98.4	94.1	99.3	99.8	97.7	**99.8**	**99.9**	**98.9**
cable	93.3	96.1	91.6	72.1	83.2	79.2	86.3	92.0	84.0	95.9	97.8	93.8	**100**	**100**	**100**
capsule	77.1	94.1	90.4	93.2	98.7	94.0	84.9	95.8	94.3	92.8	98.5	94.5	**99.7**	**99.9**	**98.7**
carpet	57.7	84.3	87.3	95.3	98.7	93.4	92.6	97.8	92.1	67.9	87.9	87.3	**96.7**	**98.8**	**94.3**
grid	83.0	94.1	87.6	**99.8**	**99.9**	**98.8**	96.6	98.9	95.0	73.0	90.4	85.4	98.4	99.5	98.7
hazelnut	68.8	85.0	78.0	**100**	**100**	**100**	93.6	96.0	94.1	99.9	**100**	99.0	99.8	99.9	98.9
leather	91.9	97.5	90.9	**100**	**100**	**100**	99.1	99.7	97.6	99.9	**100**	99.2	**100**	**100**	**100**
metal nut	92.2	98.1	93.3	97.8	99.6	97.6	97.8	99.5	96.9	99.3	99.8	99.2	**100**	**100**	**100**
pill	51.7	87.1	91.4	94.4	98.9	95.8	88.8	97.8	93.2	68.7	91.7	91.4	**98**	**99.6**	**97**
screw	59.3	81.9	86.0	88.5	96.3	89.3	84.1	94.7	87.2	22.3	64.7	85.3	**96.8**	**97.9**	**95.5**
tile	73.8	91.1	83.8	**100**	**100**	**100**	91.1	96.9	89.3	**100**	**100**	**100**	**100**	**100**	**100**
toothbrush	81.2	91.0	88.9	99.4	99.8	97.6	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
transistor	85.9	81.8	80.0	79.6	80.5	71.4	88.2	88.9	84.0	90.8	92.5	88.9	**100**	**100**	**100**
wood	49.5	81.2	86.6	**100**	**100**	**100**	77.5	92.7	86.7	98.4	99.4	98.8	98.4	99.4	98.8
zipper	59.4	82.8	88.9	**100**	**100**	**100**	98.7	99.7	97.6	99.7	99.9	99.4	99.9	**100**	99.4
Average	74.0	89.4	87.7	94.6	97.0	94.4	91.6	96.6	92.4	87.2	94.8	94.7	**99.2**	**99.7**	**98.7**

Table 11: **Comparison on the image-level anomaly detection** with AUC, AP and F_1-max metrics by training an U-Net on the generated datasets produced by Crop&Paste, DRAEM, PRN, DFMGAN and our model.

F_1-max	KDAD	CFLOW	DREAM	SSPCAB	CFA	RD4AD	PatchCore	DevNet	DRA	Ours
bottle	50.9	9.5	83.0	80.3	75.9	82.1	78.6	64.6	53.5	**87.3**
cable	18.3	10.5	58.5	51.2	76.3	65.2	68.5	54.9	55.3	**83.5**
capsule	15.1	6.9	48.9	49.5	57.0	60.4	56.7	38.7	47.7	**60.8**
carpet	54.2	3.5	60.0	47.1	48.3	67.8	67.9	52.3	42.3	**74.6**
grid	10.9	3.2	56.3	58.4	32.2	**59.9**	49.1	42.9	50.1	54.6
hazelnut	37.5	3.9	80.6	88.9	61.4	70.0	68.1	22.4	47.2	**90.6**
leather	30.4	3.9	63.2	58.1	53.8	67.2	54.7	32.1	19.8	**71.0**
metal nut	34.2	30.7	84.4	87.8	87.1	77.0	86.0	65.2	64.6	**94.0**
pill	29.9	17.6	62.6	46.5	79.5	63.7	73.5	22.8	45.5	**90.8**
screw	8.3	0.9	**66.9**	63.8	37.8	58.7	47.2	14.8	0.7	50.9
tile	27.8	26.6	**90.8**	88.5	77.8	71.8	69.4	69.9	61.4	86.2
toothbursh	25.1	4.7	47.5	37.2	62.1	58.7	63.8	35.1	22.6	**73.4**
transistor	26.7	19.9	55.2	34.8	76.3	59.8	64.2	28.9	33.2	**85.7**
wood	25.2	10.0	**75.1**	68.7	48.6	61.3	60.3	51.9	49.9	74.5
zipper	26.8	4.5	68.2	73.7	65.8	69.4	70.0	45.6	56.9	**79.2**
average	28.1	10.4	66.7	62.3	61.4	66.2	65.2	42.8	43.4	**76.4**

Table 12: **Comparison on anomaly localization with $\mathbf{F_1}$-max.**

Category	Crop&Paste			DRAEM			PRN			DFMGAN			Ours		
	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max	AUC	AP	F_1-max
bottle	85.4	95.1	90.9	99.3	99.8	**98.9**	94.9	98.4	94.1	99.3	99.8	97.7	**99.8**	**99.9**	**98.9**
cable	93.3	96.1	91.6	72.1	83.2	79.2	86.3	92.0	84.0	95.9	97.8	93.8	**100**	**100**	**100**
capsule	77.1	94.1	90.4	93.2	98.7	94.0	84.9	95.8	94.3	92.8	98.5	94.5	**99.7**	**99.9**	**98.7**
carpet	57.7	84.3	87.3	95.3	98.7	93.4	92.6	97.8	92.1	67.9	87.9	87.3	**96.7**	**98.8**	**94.3**
grid	83.0	94.1	87.6	**99.8**	**99.9**	**98.8**	96.6	98.9	95.0	73.0	90.4	85.4	98.4	99.5	98.7
hazelnut	68.8	85.0	78.0	**100**	**100**	**100**	93.6	96.0	94.1	99.9	**100**	99.0	99.8	99.9	98.9
leather	91.9	97.5	90.9	**100**	**100**	**100**	99.1	99.7	97.6	99.9	**100**	99.2	**100**	**100**	**100**
metal nut	92.2	98.1	93.3	97.8	99.6	97.6	97.8	99.5	96.9	99.3	99.8	99.2	**100**	**100**	**100**
pill	51.7	87.1	91.4	94.4	98.9	95.8	88.8	97.8	93.2	68.7	91.7	91.4	**98**	**99.6**	**97**
screw	59.3	81.9	86.0	88.5	96.3	89.3	84.1	94.7	87.2	22.3	64.7	85.3	**96.8**	**97.9**	**95.5**
tile	73.8	91.1	83.8	**100**	**100**	**100**	91.1	96.9	89.3	**100**	**100**	**100**	**100**	**100**	**100**
toothbrush	81.2	91.0	88.9	99.4	99.8	97.6	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
transistor	85.9	81.8	80.0	79.6	80.5	71.4	88.2	88.9	84.0	90.8	92.5	88.9	**100**	**100**	**100**
wood	49.5	81.2	86.6	**100**	**100**	**100**	77.5	92.7	86.7	98.4	99.4	98.8	98.4	99.4	98.8
zipper	59.4	82.8	88.9	**100**	**100**	**100**	98.7	99.7	97.6	99.7	99.9	99.4	99.9	**100**	99.4
Average	74.0	89.4	87.7	94.6	97.0	94.4	91.6	96.6	92.4	87.2	94.8	94.7	**99.2**	**99.7**	**98.7**

表11：通过在使用Crop&Paste、DRAEM、PRN、DFMGAN及本模型生成的训练集上训练U-Net，采用AUC、AP和F_1-max指标进行图像级异常检测的性能对比。

F_1-max	KDAD	CFLOW	DREAM	SSPCAB	CFA	RD4AD	PatchCore	DevNet	DRA	Ours
bottle	50.9	9.5	83.0	80.3	75.9	82.1	78.6	64.6	53.5	**87.3**
cable	18.3	10.5	58.5	51.2	76.3	65.2	68.5	54.9	55.3	**83.5**
capsule	15.1	6.9	48.9	49.5	57.0	60.4	56.7	38.7	47.7	**60.8**
carpet	54.2	3.5	60.0	47.1	48.3	67.8	67.9	52.3	42.3	**74.6**
grid	10.9	3.2	56.3	58.4	32.2	**59.9**	49.1	42.9	50.1	54.6
hazelnut	37.5	3.9	80.6	88.9	61.4	70.0	68.1	22.4	47.2	**90.6**
leather	30.4	3.9	63.2	58.1	53.8	67.2	54.7	32.1	19.8	**71.0**
metal nut	34.2	30.7	84.4	87.8	87.1	77.0	86.0	65.2	64.6	**94.0**
pill	29.9	17.6	62.6	46.5	79.5	63.7	73.5	22.8	45.5	**90.8**
screw	8.3	0.9	**66.9**	63.8	37.8	58.7	47.2	14.8	0.7	50.9
tile	27.8	26.6	**90.8**	88.5	77.8	71.8	69.4	69.9	61.4	86.2
toothbursh	25.1	4.7	47.5	37.2	62.1	58.7	63.8	35.1	22.6	**73.4**
transistor	26.7	19.9	55.2	34.8	76.3	59.8	64.2	28.9	33.2	**85.7**
wood	25.2	10.0	**75.1**	68.7	48.6	61.3	60.3	51.9	49.9	74.5
zipper	26.8	4.5	68.2	73.7	65.8	69.4	70.0	45.6	56.9	**79.2**
average	28.1	10.4	66.7	62.3	61.4	66.2	65.2	42.8	43.4	**76.4**

表12：基于$\mathbf{F_1}$-ma的异常定位对比 x。