Defect Spectrum: A Granular Look of Large-Scale Defect Datasets with Rich Semantics

Shuai Yang[1,2*], Zhifei Chen[1*], Pengguang Chen[3], Xi Fang[3], Yixun Liang[1], Shu Liu[3], and Yingcong Chen[1,2,4]

[1] Hong Kong University of Science and Technology, Guangzhou
[2] HKUST(GZ) - SmartMore Joint Lab
[3] SmartMore. Corp
[4] Hong Kong University of Science and Technology

Abstract. Defect inspection is paramount within the closed-loop manufacturing system. However, existing datasets for defect inspection often lack the precision and semantic granularity required for practical applications. In this paper, we introduce the Defect Spectrum, a comprehensive benchmark that offers **precise**, **semantic-abundant**, and **large-scale** annotations for a wide range of industrial defects. Building on four key industrial benchmarks, our dataset refines existing annotations and introduces rich semantic details, distinguishing multiple defect types within a single image. With our dataset, we were able to achieve an increase of **10.74%** in the Recall rate, and a decrease of **33.10%** in the False Positive Rate (FPR) from the industrial simulation experiment. Furthermore, we introduce Defect-Gen, a two-stage diffusion-based generator designed to create high-quality and diverse defective images, even when working with limited defective data. The synthetic images generated by Defect-Gen significantly enhance the performance of defect segmentation models, achieving an improvement in mIoU scores up to **9.85** on Defect-Spectrum subsets. Overall, The Defect Spectrum dataset demonstrates its potential in defect inspection research, offering a solid platform for testing and refining advanced models. Our project page is in `https://envision-research.github.io/Defect_Spectrum/`.

1 Introduction

Industrial manufacturing is a cornerstone of modern society. In an environment where minute imperfections can result in significant failures, ensuring top-tier quality is imperative. Manufacturing predominantly relies on a closed-loop system, encompassing production, defect inspection, filtering, and analysis, as illustrated in Figure 1.

Within this system, defect inspection plays a pivotal role, interfacing with most stages and ultimately determining product quality. Striking the right balance between identifying defective items and acknowledging sub-optimal ones,

---

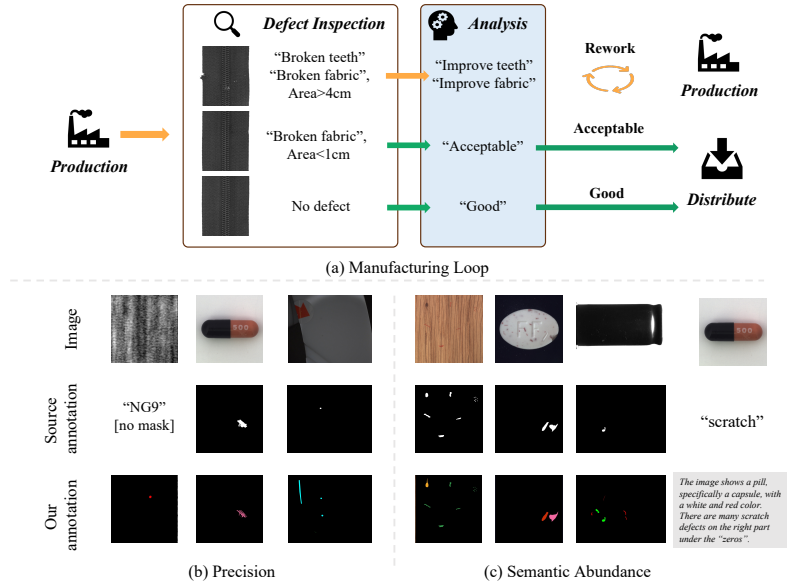[*] These authors contributed equally to this work.

Fig. 1: (a) Identifying the size, position, and type of defects is essential for quality control, as it guides the post-processing of products. Major issues, such as misaligned zipper teeth, necessitate factory rework, whereas minor problems, like fabric snags, can lead to different distribution strategies. This approach ensures the maintenance of product quality and enhances the distribution process. (b) Shows our annotation is finer, and includes those that are omitted in the source annotation. (c) Source annotation [1, 3, 46] ignores multiple defective classes within a single image, while ours provides annotation for each distinct class, shown in different colors. **Best viewed in color.**

based on defect size, position, and type, becomes critical [42]. Taking the "zipper" defect as an illustrative case. A garment zipper where the teeth are misaligned, as depicted in Figure 1 (a). This type of defect, although it might seem minor in terms of size or visibility, critically impacts the garment's functionality, necessitating its return to the factory for correction. However, defects located on the fabric, such as minor snags or slight color variations, require careful consideration of their size and impact. Small-scale fabric defects could be classified within an acceptable range, allowing for differentiated distribution strategies that might include selling these products at a discount, thereby maintaining product flow without compromising overall quality standards. Additionally, documenting the category and location of defects can pave the way for predictive maintenance and provide valuable insights for refining product repair processes [29].

However, current datasets struggle to meet the intricate practical needs of industrial defect inspection. One notable limitation is the insufficient granularity concerning defect types and locations. For instance, anomaly detection datasets

like MVTEC [3] and AeBAD [51] give pixel-level annotations but are restricted to binary masks. Meanwhile, datasets like VISION [1], though more detailed, occasionally miss or misclassify defect instances.

To address these gaps, we introduce the Defect Spectrum, aiming to offer semantics-abundant, precise, and large-scale annotations for a broad spectrum of industrial defects. This empowers practical defect inspection systems to furnish a more thorough and precise analysis, bolstering automated workflows. Building on four key industrial benchmarks, Defect Spectrum offers enhanced annotations through a rigorous labeling endeavor. We have re-evaluated and refined existing defect annotations to ensure a holistic representation. For example, contours of subtle defects, like scratches and pits, are carefully refined for better precision, and missing defects are carefully filled with the help of specialists. Beyond that, our dataset stands out by providing annotations with rich semantics details, distinguishing multiple defect types even within a single image. Lastly, we have incorporated descriptive captions for each sample, aiming to integrate the use of Vision Language Models (VLMs) in upcoming studies. During this endeavor, we employ our innovative annotation tool, Defect-Click. It has largely accelerated our labeling process, emphasizing its utility and efficiency, ensuring meticulous labeling even with the extensive scope of our dataset.

Another palpable challenge is the limited number of defective samples in datasets. For instance, in DAGM, there are only 900 defective images. In MVTEC, although it has 5354 total images, the defectives among them are merely 1258. And even the extensive VISION dataset falls short in comparison to natural image datasets like ImageNet [11] (1 million images) and ADE20k [53, 54] (20k images). To address this, we harness the power of generative models, proposing the "Defect-Gen", a two-stage diffusion-based generator. Our generator exhibits promising performance in image diversity and quality even with a limited number of training data. We show that these generated data could largely boost the performance of existing models in our Data Spectrum benchmark.

To summarize, our contributions are listed as follows.

- We introduce the Defect Spectrum dataset, designed to enhance defect inspection with its **semantics-abundant**, **precise**, and **large-scale annotations**. Unlike existing datasets, we not only refine existing annotations for a more holistic representation but also introduce rich semantics details. This dataset, building on four key industrial benchmarks, goes beyond binary masks to provide more detailed and precise annotations.
- We propose the Defect-Gen, a **two-stage diffusion-based** generator, to tackle the challenges associated with the limited availability of defective samples in datasets. This generator is shown to boost the performance of existing models, by enhancing image diversity and quality even with a limited training set.
- We conducted a comprehensive evaluation on our Defect Spectrum dataset, highlighting its versatility and application across various defect inspection challenges. By doing so, we provide a foundation for researchers to evalu-

ate and develop state-of-the-art models tailored for the intricate needs of industrial defect inspection.

2   Related Work

Industrial Datasets There are several well-used datasets for Industrial Defect Inspection: DAGM2007 [46], AITEX [36], AeBAD [51], BeanTech [27], Cotton-SFDG [20] and KoektorSDD [39] offer commonly seen images that cover a wide array of manufacturing materials; MVTEC [2, 3] is a dataset for benchmarking anomaly detection methods with a focus on industrial inspection; VISION V1 [1] includes a collection of 14 industrial inspection datasets containing multiple objects. A notable shortcoming in the aforementioned industrial datasets is they often lack specificity regarding the defect's type or its precise location. Aiming to refine these issues, we introduce the Defect Spectrum datasets. Further details will be explained in Section 3.

Defect-mask Generation Defect inspection plays a vital role in various industries, including manufacturing, healthcare, and transportation. Previous attempts based on the traditional computer vision method [37] have proven to be robust for detecting small defects, but they all suffer from detecting defects in textures-rich patterns. In recent years, Convolutional Neural Networks(CNNs) [15, 16, 28] based models are commonly used for defect inspection, but limited availability of real-world defect samples remains a challenge. To mitigate such data-insufficiency issue, traditional methods for synthesizing defect images manually destroy normal samples [26] or adopt Computer-Aided Drawing (CAD) [19, 25]. Deep learning-based approaches are generally effective, but they require large amounts of data. GAN-based methods [14, 31, 45, 50] are adopted to perform defect sample synthesis for data augmentation. DefectGAN adopts an encoder-decoder structure to synthesize defects by mimicking defacement and restoration processes. However, it is important to note that GAN-based methods typically require a substantial quantity of real defect data in order to achieve effective results. Recent advancements in Diffusion models [12, 18, 30] demonstrated a superior performance in image generation. However, they tend to reproduce existing samples when trained with scarce data, leading to a lack of diversity. Stable Diffusion [32] is one of the most prevailing methods in this field. Nonetheless, it is not applicable to use a pre-trained stable diffusion model when generating masks. Our proposed approach, on the other hand, is capable of generating defective image-mask pairs with both diversity and high quality, even when trained on limited datasets.

3   Dataset

3.1   Datasets Analysis

In Table 1, we present an analysis of the Defect Spectrum datasets in comparison with other prevalent industrial datasets. Notably, the DAGM2007 and Cotton-

Fabric datasets originally lacked pixel-wise labels, making them less suitable for detailed defect inspection. While datasets like AITEX, AeBAD, BeanTech, and KoektorSDD offer defect masks, they only focus on a limited range of products, offering a restricted number of annotated images and defect categories.

While some high-quality datasets offer a significant volume of images with pixel-level annotations, they are not without their limitations. For instance, there are cases where MVTEC and VISION annotations either miss defects or provide imprecise, coarse labels, as illustrated in Figure 1(b). Additionally, these datasets commonly merge various defect classes into a single homogeneous category. This shortcoming is particularly apparent in the "pill" and "capacitor" examples in Figure 1(c), where the original annotations provide only binary masks that do not differentiate between defects such as "scratch", "crack", and "color point". This approach fails to reflect real-world scenarios, where industrial images frequently exhibit multiple types of defects simultaneously.

To enhance the capabilities for defect detection, Defect Spectrum datasets introduce a comprehensive collection of 3518 high-quality, high-resolution images derived from the aforementioned datasets. These selected images feature a wide variety of objects and defects, ensuring extensive variance and coverage for improved analysis. This curated dataset offers detailed, precise, and diverse category annotations for each image and enriches the data with comprehensive captions to facilitate better contextual understanding. For every product type featured, the Defect Spectrum datasets extend their utility by incorporating realistic synthetic data and their accurate masks, ensuring a thorough and versatile testing ground.

Table 1: Comparison with real-world manufacturing datasets. Defect Spectrum datasets are the second largest one even though excluding our synthetic data. Defect Spectrum is also the most diverse, semantics-abundant, and precise manufacturing benchmark datasets to date. We use * to represent the amount of synthetic data.

	Annotated Defective Images	Defect Type	Pixel-wise Label	Multiple Defective Label	Detailed Caption
AITEX [36]	105	12	✓		
AeBAD [51]	346	4	✓		
BeanTech [27]	290	3	✓		
Cotton-Fabric [20]	89	1			
DAGM2007 [46]	900	6			
KolektorSDD2 [39]	356	1	✓		
MVTec [3]	1258	69	✓		
VISION V1 [1]	4165	44	✓	✓	
VisA [56]	1200	75	✓		
Defect Spectrum	**3518+1920***	**125**	✓	✓	✓

3.2   Annotation Improvements

Our improvements in annotations are mainly in three aspects: precision, semantics-abundance, and detailed caption.

Precision For datasets that were not annotated or merely had image-wise annotations, we have elevated them to meet our standards. We have enriched these datasets with meticulous pixel-level annotations, delineating defect boundaries and assigning a distinct class label to each type of defect. For those datasets that already possessed pixel-wise masks, we enhanced their precision and rectified any imperfections. We undertook efforts to account for any overlooked defects, ensuring exhaustive coverage. For nuanced defects, such as scratches and pits, we refined the contours to achieve heightened accuracy.

Semantics Abundance In contrast to datasets that only offer binary defective masks, Defect Spectrum furnishes annotations with more semantic details, identifying multiple defect types within a single image. We identify that there are 552 multiple defective images and provide their multi-class labels. Moreover, we have re-assessed and fine-tuned the existing defect classes, guaranteeing a more granular and precise categorization. In total, we offer 125 distinct defect classes.

Detailed Caption With the evolution of Vision Language Models (VLMs), we have equipped our datasets by integrating exhaustive captions. It's worth noting that current captioning models, such as BLIP2 [22] and LLaVa [23], often overlook defect information. As a remedy, we manually refined the captions from VLMs and furnished detailed descriptions. These narratives not only identify the objects but also elucidate their specific defects. We anticipate that this enhancement will inspire researchers to increasingly leverage VLMs for defect inspection in forthcoming studies.

3.3   Defect Generation

To tackle the issue of defects scarcity, we turn to the burgeoning field of generative models. By using the limited available data, we propose a two-staged diffusion-based generator, called the "Defect-Gen".

Background Given a set of defective image-mask pairs, we aim to learn a generative model that captures the true data distribution, so that it can generate more pairs to augment the training set. We denote the dataset as $\mathcal{D} = \{(I_1, M_1), (I_2, M_2), \ldots, (I_N, M_N)\}$, where the image $I_i \in \mathbb{R}^{h \times w \times 3}$ and the mask $M_i \in \{0, n\}^{h \times w \times n}$ refer to the defect image and its corresponding defect mask respectively. N is the number of samples in the training set, which is small in practice.
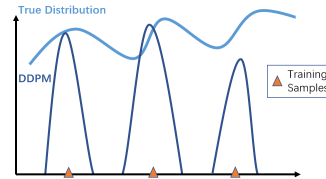

Fig. 2: DDPM predicts high density around training samples and fails to capture the true data distribution.

n denotes the number of defective types in the mask images. Specifically, we convert the mask into a one-hot encoding scheme for each channel separately. We show that with a very small modification, it can generate images with corresponding labels. We perform a channel-wise concatenation between I with the M, i.e., $x = I \oplus M$, where $\oplus$ means concatenation and $x \in \mathbb{R}^{h \times w \times n_{total}}$, and $n_{total} = n_{defect} + 3$. We then treat the x as the *input* to train the generator. This improves the usability of the generative model with negligible computational overhead. In the following, we term x as "image" instead of "image with label" for convenience.
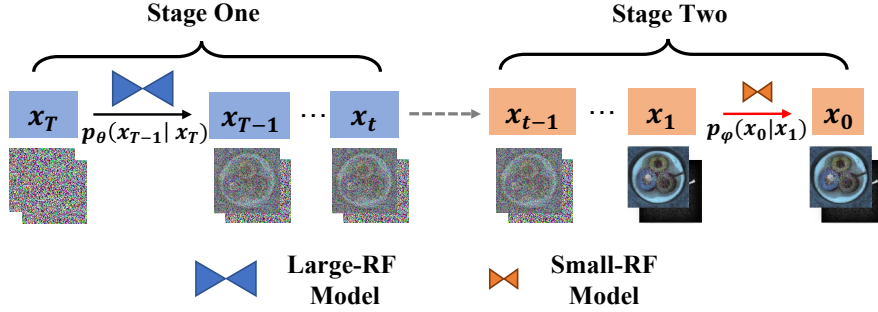

Fig. 3: The inference process of the two staged diffusion models. The input to the large model p_θ is gaussian noise, after the optimal step is reached, the intermediate results containing global information will be used as the input to the small model p_ϕ.

Few-shot Challenges Note that defect images are difficult to collect in practice, and thus, models have to be trained with very few samples. Under this situation, we observe that the generated results lack diversity. To be specific, models tend to memorize the training set. The reason could be that the generative models such as Diffusion models tend to predict high density around training samples and fail to capture the true data distribution, as depicted in Figure 2.

Overfitting Issue The limitation discussed above is not surprising. In statistical learning theory, it is well-known that the generalization capacity of a classification model is positively related to the sample size and negatively related to the dimension. We can reasonably hypothesize that a similar trend also holds in the diffusion model according to the Vapnik–Chervonenkis theory [41]. In this sense, as the data dimension ($h \times w \times n_{total}$) is much larger than the sample
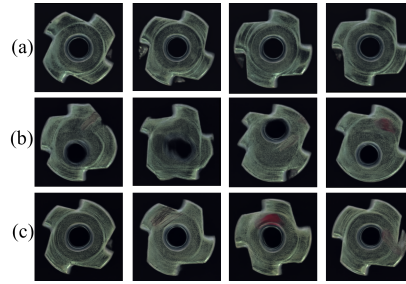

Fig. 4: The visual cases in (a) demonstrate a lack of diversity in using DDPMs. Cases in (b) demonstrated excessive diversity. (c) shows the generated samples using our framework. we maintained the global structure while introducing local variance.

size ($N = 25$ in our setting), the vanilla diffusion model suffers severe overfitting. As shown in Figure 4 (a), DDPM replicates training cases, leading to low diversity generation.

Modeling the Patch-level Distribution To alleviate the aforementioned problem, we propose to model the patch-level distribution instead of the image-level distribution. By treating a patch as one sample, the data dimension ($h_{patch} \times w_{patch} \times n_{total}$) is largely reduced, while the sample size (N_{patch})is significantly increased. This reduces the risk of overfitting. Figure 5. demonstrates the effectiveness of our strategy.

Restraining the Receptive Field Although we can naively replace x with cropped image patches to achieve patch-level modeling, it is hard to use learned patches to reconstruct into a whole image during inference. In other words, if explicitly train a patched generator, we would have to introduce a reconstruction term to merge these patches. Alternatively, we leverage the network architecture to restrain the size of the receptive field to achieve this. Standard U-Net is used in the vanilla diffusion model [18]. It is composed of a series of down-sampling layers. With the reduced number of down-sampling layers, the output receptive fields gradually decrease. This allows the model to only be visible to small patches on the original images. This strategy does not change the position of each patch in one image and thus has the potential to maintain the whole image. Thus, by using a smaller receptive field, patch-level modeling is achieved.

Handling the Global Distortion
While patch-level modeling is effective in overcoming overfitting, it falls short of representing the global structure of the entire image, leading to unrealistic results. This is shown in Figure 4(b). To address this issue, we propose a two-stage diffusion process as depicted in Figure 3. Our approach is inspired by [9], which reveals that different time steps in the diffusion process correspond to distinct levels of information. In the early stages, coarse geometry information is generated, while in later stages, finer information is produced.

Training Samples   Generated Result

Fig. 5: The property of patch-level modeling. The right image is generated from the small-receptive-field model, and the two left images are the two most similar images from the training set.

Specifically, we train two models: one with a small receptive field, which we introduced previously, and another with a larger receptive field. During inference, we use the large-receptive-field model to capture the geometry structure in the early steps, and then switch to the small-receptive-field model to generate
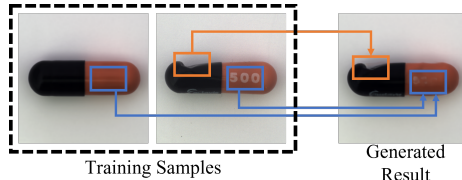
diverse local patches in the remaining steps. The effectiveness of this strategy is demonstrated in Figure 4 (c). Our model has two key hyper-parameters: the switch timestep u and the receptive field of the small model. Both of them can control the trade-off between fidelity and diversity. We use FID to measure the generation fidelity. LPIPS was originally used for measuring the similarity between two images, the lower score indicates a higher similarity and vice versa. In this scenario, to achieve a higher generation diversity with fidelity, we want to maintain a higher LPIPS score with a similar FID score. Due to the page limits, the detailed selection of the switch timestep u and the receptive field of the small model can be found in the Sec.B of the Appendix.

3.4 Auxiliary Annotation Tool

Fig. 6: Comparison between Defect Click and Segment Anything [21]. Progressively annotating a scratched capsule with human clicks: With our "Defect-Click" tool, we can swiftly pinpoint the two scratches. However, when using "Segment Anything", it becomes challenging to accurately identify the defects, as shown in the red box. **Best viewed in color.**

Annotating pixel masks is an exceptionally demanding task in the labeling domain, especially under the stringent standards of Defect Spectrum. It is not feasible to perform such a task from scratch. To alleviate this challenge, we introduce an auxiliary annotation tool, "Defect-Click," designed to conserve the efforts of our specialists.

Defect-Click is an advanced interactive annotation tool designed to automatically segment defect areas based on a user's click point. Distinct from traditional interactive segmentation methods, Defect-Click utilizes its pretrained knowledge of industrial defects to adeptly pinpoint irregular defective regions. Built upon the Focal-Click framework [6,7], we tailored Defect-Click for the industrial defect domain by integrating 21 proprietary labeled datasets, introducing multi-level crop training for small defects, and incorporating edge-sensitive losses during training. The 21 proprietary labeled datasets consist of defective image-mask pairs for industrial inspection. Multi-level crop training means we rescale the training samples randomly to a resolution of [512, 1024, 1536, 2048, 2560, 3072] and then crop 512×512 patches for training. Edge-sensitive losses denote the loss function in Mask2Former [8]. We use the $loss_{cls} : loss_{mask} : loss_{dice} = 2 : 5 : 5$

in practice. These specialized approaches ensure that Defect-Click significantly outperforms other annotation tools in the industrial dataset domain, as showcased in Figure 6. Segment Anything [21] struggles to identify the scratch defect, while Defect-Click clearly delineates the defect's contour.

With the assistance of Defect-Click, we can initially obtain a rough defect mask with merely several clicks and subsequently refine its imperfections. On average, this approach has resulted in a time-saving of about 60%. Even though, this comprehensive annotation project still spans a total of 580 working hours.

4   Experiments

4.1   Benchmarking existing methods

In the realm of industrial defect inspection, there are three primary tasks: defect detection (determining if an image contains a defect), defect classification (identifying the type of defect), and defect segmentation (pinpointing both the boundaries and the type of the defect in the image) [4, 24, 40]. Typical defect detection methods such as Patchcore [34], PADIM [10], and BGAD [48] emphasize identifying the presence of defects but fall short in discerning defect types. Defect classification methods can determine the type of defect but do not provide information about its location or size. Our Defect Spectrum dataset come with detailed and comprehensive annotations, aiming to solve the most complex task. Consequently, we focus on methods that excel in defect segmentation.

Additionally, due to the confidential nature of many industrial products, transferring data externally is often prohibited. This necessitates models that can operate efficiently on local devices. With this in mind, we have handpicked several SOTA segmentation methods and adapted them to a lightweight version. Our baseline includes UNet - small [33], ResNet18 [17] - PSPNet [52], ResNet18 - DeepLabV3+ [5], HRNetV2W18 - small [43], BiseNetV2 [49], ViT - Tiny [13]- Segmenter [38], Segformer - MiT - B0 [47], and HRNet - Mask2Former [8]. The models are abbreviated as follows: UNet (UNet - small), PSP (ResNet18 - PSP-Net), DL (ResNet18 - DeepLabV3+), HR (HRNetw18small), Bise (BiseNetV2), V-T (ViT-Tiny - Segmenter), M-B0 (Segformer - MiT-B0), and M2F (HRNet - Mask2Former).

We present a comprehensive evaluation of the above methods on each sub-set of our Defect Spectrum benchmark. For the performance metric, we choose the mean Intersection over Union (mIoU). Results are shown in Table 2. The consistent performance of DeepLabV3+ across multiple datasets suggests that it's a robust model for various types of defect segmentation tasks. The Transformer-based models seem to be particularly effective for Cotton-Fabric. This might be due to the inherent advantages of Transformers in capturing long-range semantic information, which could be commonly found in "Cotton-Fabric". Performance varies across models for different categories, suggesting no universal solution. Model selection should consider dataset specifics. Some datasets challenge all models, highlighting a need for more research.

Table 2: Quantitative comparison of various defect segmentation methods across different Defect Spectrum reannotated datasets. Results reflect the mIoU. We highlight the best mIoU of each dataset with red color. "DS" is abbreviated for Defect Spectrum.

		CNNs					Transformers		
		UNet	PSP	DL	HR	Bise	V-T	M-B0	M2F
	bottle	43.44	50.20	56.53	45.02	44.92	69.71	40.88	53.20
	cable	47.95	52.50	52.59	50.39	45.24	54.51	58.31	49.72
	capsule	28.05	29.59	35.49	34.02	28.30	33.94	38.95	26.91
	carpet	50.91	53.76	53.75	47.28	44.52	43.70	38.45	47.34
	grid	37.06	42.86	41.18	30.97	33.89	40.08	18.86	24.81
	h_nut	58.84	56.87	61.78	59.31	57.53	55.07	59.60	56.72
Defect	leather	57.56	61.42	54.56	55.45	57.89	47.85	50.80	53.96
Spectrum	m_nut	49.18	46.99	51.08	48.76	55.51	54.68	48.89	39.43
(MVTec)	pill	35.81	36.38	33.83	29.30	27.23	42.65	46.35	27.14
	screw	31.87	38.77	33.36	29.66	19.01	22.54	19.26	21.89
	tile	85.49	82.51	83.02	85.66	84.21	78.29	79.14	83.04
	t_brush	23.96	25.25	25.16	26.25	25.58	33.30	32.22	28.26
	tran.	40.37	44.02	58.23	44.50	45.97	53.60	41.13	50.87
	wood	72.69	67.93	68.21	69.00	67.81	62.62	73.02	59.66
	zipper	54.83	60.95	58.03	55.87	47.15	51.69	60.12	49.47
	mean	**49.88**	**51.41**	**51.58**	**47.99**	**45.40**	**49.31**	**46.45**	**45.70**
	Capa.	57.04	54.01	52.75	54.56	54.36	56.30	59.29	57.27
Defect	Console	35.32	30.77	32.67	31.70	30.45	22.48	25.64	32.50
Spectrum	Ring	52.37	56.16	56.97	60.17	54.06	44.27	52.21	62.09
(VISION)	Screw	53.13	53.91	55.20	52.46	51.76	36.87	47.54	52.05
	Wood	64.75	66.80	66.72	66.79	67.40	53.34	63.43	66.70
	mean	**52.52**	**52.33**	**52.86**	**53.14**	**51.61**	**42.65**	**49.62**	**54.12**
DS-DAGM2007		85.89	85.14	86.82	84.02	83.14	52.42	83.06	85.56
DS-Cotton-Fabric		39.03	48.73	47.55	41.13	46.82	51.29	50.52	64.09

4.2   Generation Quality

Figure 7 provides a qualitative comparison between our generation results and those from other synthesis methods. On the left-hand side, we present different objects to demonstrate the high fidelity of our method. On the right-hand side, we used the two images shown in the "Real defect" to generate samples to demonstrate our high diversity. We observe that the generated models from CycleGAN [55] and DDPM [18] completely failed to learn a diverse defect pattern and thus failed to generate samples with diversity by producing mere duplicates of the training set. On the other hand, sinDiffusion [44] and SinGAN [35] can produce diverse samples but are not visually realistic. More visual cases, including other classes, can be found in the supplementary file. Figure 8 displays the image-mask pairs we generated. Our images are of high quality, and the corresponding masks align well with them.
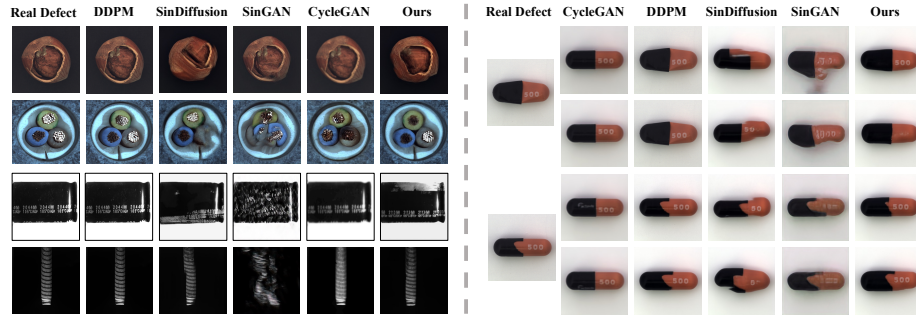
Fig. 7: Qualitative comparison of our method with other image synthesis methods. On the left-hand side, we compared different objects across different datasets to demonstrate the high fidelity of our generation method. On the right-hand side, we show our method can exhibit diversity while maintaining high quality. **Best viewed in color.**
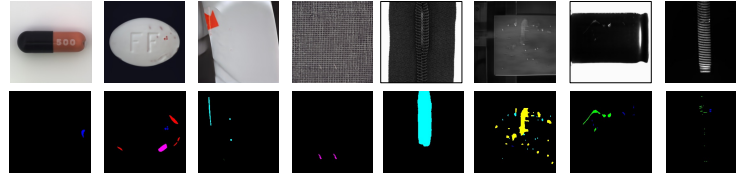

Fig. 8: Qualitative results of our proposed defect generation method. Generated images demonstrated rich semantics, exhibiting high quality. Generated masks precisely reflected defect areas. **Best viewed in color.**

4.3   Synthetic Data for Performance Boost

Boosting SOTA methods with Synthetic Data Results in Table 3 show a large performance increase in both DS-MVTec and DS-Cotton datasets, the increase is comparatively smaller in the DS-VISION dataset, however, such increase is demonstrated in each of the sub-classes. We do not generate extra data for DS-DAGM2007, since it is already a synthetic dataset. The result demonstrates the effectiveness of our synthetic data. We also compared with other generation methods in the ability to boost performance. Detailed comparisons can be found in the supplementary file.

Table 3: Performance (mIoU) comparison between models trained with and without synthetic data. The bolded text indicates results with synthetic data. "DS" is abbreviated for Defect Spectrum.

	DS-MVTec	DS-VISION	DS-Cotton
DeepLabV3+	51.58/**55.55**	52.33/**53.46**	48.73/**58.58**
Mask2Former	45.70/**50.16**	54.12/**55.47**	64.09/**65.39**
MiT-B0	46.45/**56.21**	49.62/**50.75**	50.52/**55.86**

Impact of Synthetic Data In Figure 9, we delve deeper into the impact of varying the quantity of our synthetic data on model performance. Figure 9 (a) shows the performance improvement over different quantities of synthetic data using DeepLabV3+. Interestingly, we found that the transformer-based model (MiT-B0) benefits much more with synthetic data than CNN-based models, as shown in Figure 9 (b).
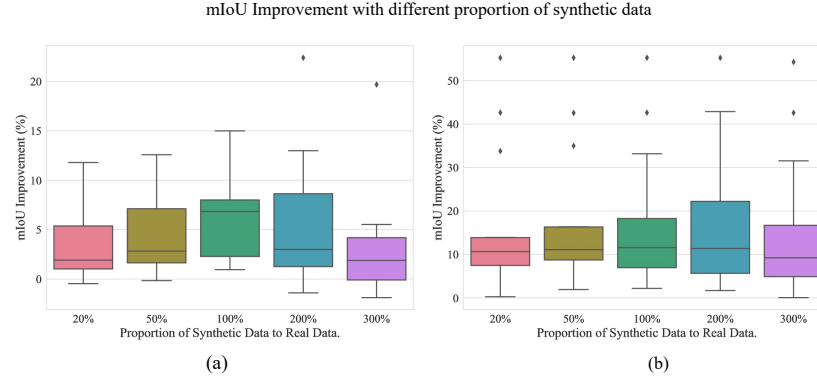
mIoU Improvement with different proportion of synthetic data

Fig. 9: Improvement in mIoU with different proportions of synthetic Data. This experiment is done on Defect Spectrum (MVTec) with DeeplabV3+ and MiT-B0 shown as (a) and (b) respectively.

When using synthetic data that is 20% of the size of the original training set, there is an enhancement in the results. Additionally, it's worth noting that the optimal amount of synthetic data required can vary based on the specific category of images. When using synthetic data that is 200% of the size of the original training set, there is an enhancement in the performance, but results in greater variance. Additionally, the performance starts to decrease after reaching the 300%. On a holistic scale, integrating 100% of the synthetic data appears to be a reasonable choice.

4.4   Comparison between original and Defect Spectrum dataset

Table 4: The quality control benchmark for the objects to be inspected. We show Zipper, pill, and wood as example classes.

Example classes	Standard For Benign Products
Zipper	No defect on teeth; Fabric defect < 4800 pixels
Pill	No cracks; Contamination < 4000 pixels; Color stains < 300 pixels
Wood	No scratch; No dent; Impurities < 250 pixels; Stain < 1000 pixels

The enhancement of our dataset contains two significant modifications: 1) the expansion to include more defect classes, and 2) the improvement of annotation accuracy in both training and validation sets. Given these sig-

Table 5: Comparison of original annotation and defect spectrum annotation in the simulation experiment of manufacturing production. Metrics are reported in image level recall rate and false positive rate (FPR).

Method	↑ Recall (%)	↓ FPR (%)
Original	85.33	49.60
Defect Spectrum (DS)	**96.07**	**16.50**

nificant changes, it becomes impractical to directly assess the performance (like computing mIoU) of our refined model on the original ground truth, and vice versa.

To objectively assess our dataset's superiority, we design a simulation experiment that mirrors real-world manufacturing processes. Manufacturing experts are invited to set a quality control benchmark for the inspected items, as detailed in Table 4. This benchmark specifies unacceptable critical defects and establishes a threshold for minor defects—for instance, any teeth defect in a zipper is intolerable, whereas only extensive fabric defects in a zipper are considered detrimental. Following these criteria, we classify the validation samples as either benign or defective. We then train two segmentation models of the same architecture: one on our refined dataset and the other on the original dataset. Utilizing these segmentation results and the established criteria, we calculated the image-level recall rate ($\frac{TP}{FN+TP}$) and the false positive rate ($\frac{FP}{TN+FP}$). A superior recall rate denotes more effective defective product identification, whereas a reduced false positive rate indicates fewer benign products mistakenly flagged as defective. As shown in Table 5, the model trained with our refined annotations outperforms the one trained with the original dataset in terms of recall rate and false positive rate, thus enhancing product profitability without compromising quality.

Furthermore, we conducted qualitative comparisons across all subsets within the Defect Spectrum. We compared the annotations from the original dataset with those from our refined dataset. Additionally, we evaluated the segmentation model's masks based on the original dataset annotations against those derived from our refined dataset. Both the masks and the annotations exhibit greater accuracy and improved differentiation among defect types in comparison to the original dataset. Visual examples of these annotations and segmentation masks are included in the appendix for further reference.

5   Conclusion

In conclusion, our Defect Spectrum dataset, complemented by the Defect-Gen generator, addresses critical gaps in industrial defect inspection. By providing Semantics-abundant, precise, and large-scale annotations, our contributions will foster advancements in defect inspection methodologies. The potential integration of Vision Language Models, the practical value of labeling assistant Defect-Click, coupled with the Defect-Gen's capability to mitigate data scarcity, sets the stage for more robust defect inspection systems in the future.

References

1. Bai, H., Mou, S., Likhomanenko, T., Cinbis, R.G., Tuzel, O., Huang, P., Shan, J., Shi, J., Cao, M.: Vision datasets: A benchmark for vision-based industrial inspection. arXiv preprint arXiv:2306.07890 (2023)
2. Bergmann, P., Batzner, K., Fauser, M., Sattlegger, D., Steger, C.: The mvtec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. International Journal of Computer Vision **129**(4), 1038–1059 (2021)
3. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9592–9600 (2019)
4. Carvalho, P., Durupt, A., Grandvalet, Y.: A Review of Benchmarks for Visual Defect Detection in the Manufacturing Industry, p. 1527–1538. Springer International Publishing (Sep 2022). `https://doi.org/10.1007/978-3-031-15928-2_133`, `http://dx.doi.org/10.1007/978-3-031-15928-2_133`
5. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 801–818 (2018)
6. Chen, X., Zhao, Z., Yu, F., Zhang, Y., Duan, M.: Conditional diffusion for interactive segmentation. In: ICCV (2021)
7. Chen, X., Zhao, Z., Zhang, Y., Duan, M., Qi, D., Zhao, H.: Focalclick: Towards practical interactive image segmentation (2022)
8. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation (2022)
9. Choi, J., Lee, J., Shin, C., Kim, S., Kim, H., Yoon, S.: Perception prioritized training of diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)
10. Defard, T., Setkov, A., Loesch, A., Audigier, R.: Padim: a patch distribution modeling framework for anomaly detection and localization (2020)
11. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 248–255. IEEE (2009), `https://ieeexplore.ieee.org/abstract/document/5206848/`
12. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Advances in neural information processing systems **34**, 8780–8794 (2021)
13. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale (2021)
14. Du, Z., Gao, L., Li, X.: A new contrastive gan with data augmentation for surface defect recognition under limited data. IEEE Transactions on Instrumentation and Measurement (2022)
15. Faghih-Roohi, S., Hajizadeh, S., Núñez, A., Babuska, R., De Schutter, B.: Deep convolutional neural networks for detection of rail surface defects. In: 2016 International joint conference on neural networks (IJCNN). pp. 2584–2589 (2016)
16. Guo, J., Wang, Q., Li, Y.: Semi-supervised learning based on convolutional neural network and uncertainty filter for façade defects classification. Computer-Aided Civil and Infrastructure Engineering pp. 302–317 (2021)

17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)

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

19. Huang, Q., Wu, Y., Baruch, J., Jiang, P., Peng, Y.: A template model for defect simulation for evaluating nondestructive testing in x-radiography. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans **39**, 466–475 (2009)

20. Incorporated, C.: Standard fabric defect glossary (2023), uRL: `https://www.cottoninc.com/quality-products/textile-resources/fabric-defect-glossary`

21. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything. arXiv:2304.02643 (2023)

22. Li, J., Li, D., Savarese, S., Hoi, S.: BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In: ICML (2023)

23. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning (2023)

24. Lu, F., Yao, X., Fu, C.W., Jia, J.: Removing anomalies as noises for industrial defect localization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16166–16175 (2023)

25. Mery, D., Hahn, D., Hitschfeld, N.: Simulation of defects in aluminium castings using cad models of flaws and real x-ray images. Insight-Non-Destructive Testing and Condition Monitoring pp. 618–624 (2005)

26. Mery, D., Filbert, D.: Automated flaw detection in aluminum castings based on the tracking of potential defects in a radioscopic image sequence. IEEE Transactions on Robotics and Automation **18**(6), 890–901 (2002)

27. Mishra, P., Verk, R., Fornasier, D., Piciarelli, C., Foresti, G.L.: VT-ADL: A vision transformer network for image anomaly detection and localization. In: 30th IEEE/IES International Symposium on Industrial Electronics (ISIE) (June 2021)

28. Mundt, M., Majumder, S., Murali, S., Panetsos, P., Ramesh, V.: Meta-learning convolutional neural architectures for multi-target concrete defect classification with the concrete defect bridge image dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11196–11205 (2019)

29. Ni, C., Yang, K., Xia, X., Lo, D., Chen, X., Yang, X.: Defect identification, categorization, and repair: Better together (2022)

30. Nichol, A.Q., Dhariwal, P.: Improved denoising diffusion probabilistic models. In: International Conference on Machine Learning. pp. 8162–8171. PMLR (2021)

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

32. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models (2021)

33. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation (2015)

34. Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., Gehler, P.: Towards total recall in industrial anomaly detection (2022)

35. Rott Shaham, T., Dekel, T., Michaeli, T.: Singan: Learning a generative model from a single natural image. In: Computer Vision (ICCV), IEEE International Conference on (2019)

36. Silvestre-Blanes, J., Albero-Albero, T., Miralles, I., Pérez-Llorens, R., Moreno, J.: A public fabric database for defect detection methods and results. Autex Research Journal **19**(4), 363–374 (2019). `https://doi.org/doi:10.2478/aut-2019-0035`, `https://doi.org/10.2478/aut-2019-0035`

37. Song, W., Chen, T., Gu, Z., Gai, W., Huang, W., Wang, B.: Wood materials defects detection using image block percentile color histogram and eigenvector texture feature. In: Proceedings of the First International Conference on Information Sciences, Machinery, Materials and Energy. Atlantis Press (2015). `https://doi.org/10.2991/icismme-15.2015.163`, `https://doi.org/10.2991/icismme-15.2015.163`

38. Strudel, R., Garcia, R., Laptev, I., Schmid, C.: Segmenter: Transformer for semantic segmentation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7262–7272 (2021)

39. Tabernik, D., Šela, S., Skvarč, J., Skočaj, D.: Segmentation-based deep-learning approach for surface-defect detection. Journal of Intelligent Manufacturing **31**(3), 759–776 (2020)

40. Tang, J., Lu, H., Xu, X., Wu, R., Hu, S., Zhang, T., Cheng, T.W., Ge, M., Chen, Y.C., Tsung, F.: An incremental unified framework for small defect inspection. In: 18th European Conference on Computer Vision (ECCV) (2024), `https://github.com/jqtangust/IUF`

41. Vapnik, V.N., Chervonenkis, A.Y.: On the uniform convergence of relative frequencies of events to their probabilities. Measures of complexity: festschrift for alexey chervonenkis (2015)

42. Wagner, S.: A literature survey of the quality economics of defect-detection techniques. CoRR **abs/1612.04590** (2016), `http://arxiv.org/abs/1612.04590`

43. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., et al.: Deep high-resolution representation learning for visual recognition. IEEE transactions on pattern analysis and machine intelligence **43**(10), 3349–3364 (2020)

44. Wang, W., Bao, J., Zhou, W., Chen, D., Chen, D., Yuan, L., Li, H.: Sindiffusion: Learning a diffusion model from a single natural image. arXiv preprint arXiv:2211.12445 (2022)

45. Wei, J., Zhang, Z., Shen, F., Lv, C.: Mask-guided generation method for industrial defect images with non-uniform structures. Machines **10**(12), 1239 (2022)

46. Wieler, M., Hahn, T.: Weakly supervised learning for industrial optical inspection. In: DAGM symposium in. vol. 6 (2007)

47. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers (2021)

48. Yao, X., Li, R., Zhang, J., Sun, J., Zhang, C.: Explicit boundary guided semi-push-pull contrastive learning for supervised anomaly detection (2023), `https://arxiv.org/abs/2207.01463`

49. Yu, C., Gao, C., Wang, J., Yu, G., Shen, C., Sang, N.: Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. International Journal of Computer Vision **129**, 3051–3068 (2021)

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

51. Zhang, Z., Zhao, Z., Zhang, X., Sun, C., Chen, X.: Industrial anomaly detection with domain shift: A real-world dataset and masked multi-scale reconstruction. arXiv preprint arXiv:2304.02216 (2023)

52. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2881–2890 (2017)
53. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)
54. Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., Torralba, A.: Semantic understanding of scenes through the ade20k dataset. International Journal of Computer Vision **127**(3), 302–321 (2019)
55. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Computer Vision (ICCV), 2017 IEEE International Conference on (2017)
56. Zou, Y., Jeong, J., Pemula, L., Zhang, D., Dabeer, O.: Spot-the-difference self-supervised pre-training for anomaly detection and segmentation (2022)

Supplementary Material for Defect Spectrum: A Granular Look of Large-Scale Defect Datasets with Rich Semantics

Shuai Yang[1,2*], Zhifei Chen[1*], Pengguang Chen[3], Xi Fang[3], Shu Liu[3], and Yingcong Chen[1,2,4]

[1] Hong Kong University of Science and Technology, Guangzhou
[2] HKUST(GZ) - SmartMore Joint Lab
[3] SmartMore. Corp
[4] Hong Kong University of Science and Technology

In this supplementary, we extended our experiment to incorporate more annotation comparisons with existing datasets in Sec. 1. The detailed generation settings and more quantitative analysis are discussed in Sec. 2. We also include more visual cases in Sec. 3 to demonstrate the capacity of our framework to maintain both fidelity and diversity.

1 Visual Comparison between Original and Defect Spectrum Dataset

In this section, we first present a visual comparison between ours (the last row) and the original datasets' annotation. Figure 1, 2, 3 shows the comparison of the MVTec dataset, we re-classify the defects based on their type and enabled more semantic abundance. As for Figure 4 of the VISION dataset, we refined the original annotation for more granularity. The original DAGM and Cotton datasets contained no pixel-level annotation, thus we provide our annotation as shown in Figure 5, 6. We also demonstrate the efficacy of our refined annotations for defect inspection by employing a segmentation model. As illustrated in Figure 7, 8 and Figure 9, the segmentation model trained on our refined dataset demonstrates enhanced precision and an improved capability to differentiate between various types of defects, compared to its performance when trained on the original dataset.

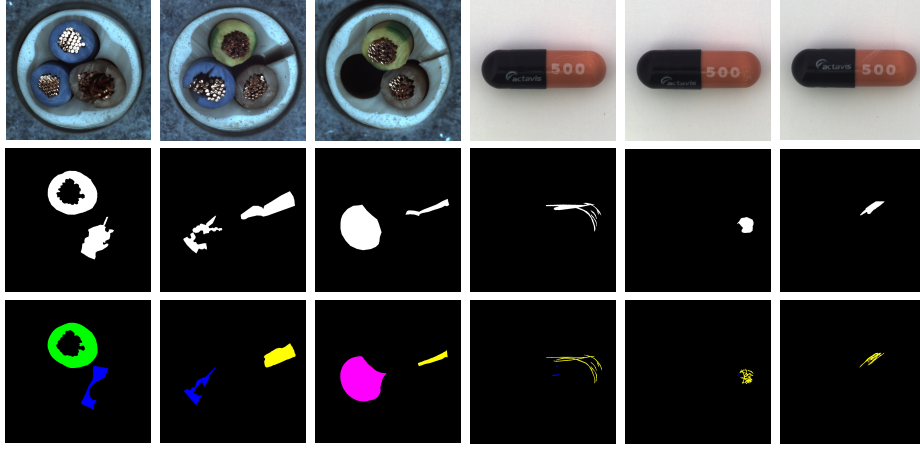[*] These authors contributed equally to this work.

Fig. 1: The annotation comparison of the "cable" and "capsule" class in MVTec dataset. The first row shows the defect image. Rows 2 and 3 show the original annotation and our improved annotation. **Best viewed in color.**
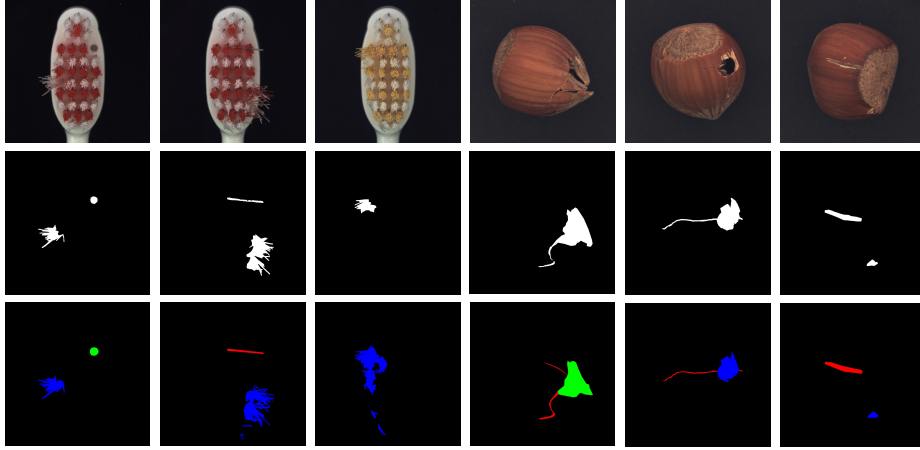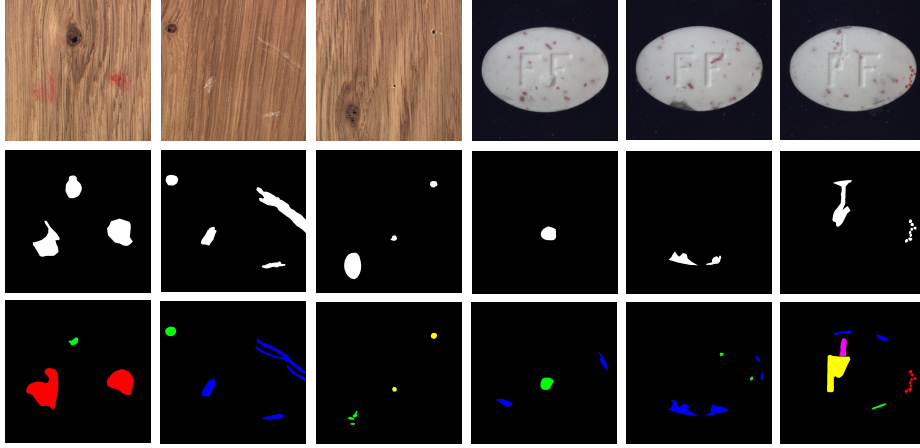

Fig. 2: The annotation comparison of the "toothbrush" and "hazelnut" class in MVTec dataset. The first row shows the defect image. Rows 2 and 3 show the original annotation and our improved annotation. **Best viewed in color.**
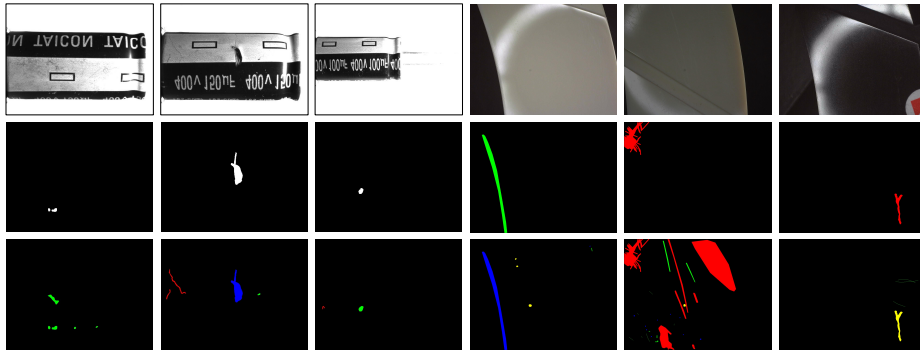
Fig. 3: The annotation comparison of the "wood" and "pill" class in MVTec dataset. The first row shows the defect image. Row 2 and 3 show the original annotation and our improved annotation. **Best viewed in color.**

Fig. 4: The annotation comparison of the "capacitor" and "ring" class in VISION dataset. The first row shows the defect image. Rows 2 and 3 show the original annotation and our improved annotation. **Best viewed in color.**
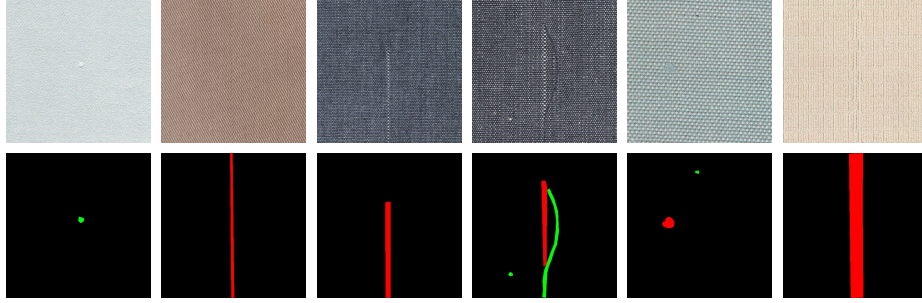
Fig. 5: The annotation comparison of the "cotton fabric" class in the COTTON dataset. The first row shows the defect image. Row 2 shows our improved annotation. **Best viewed in color.**
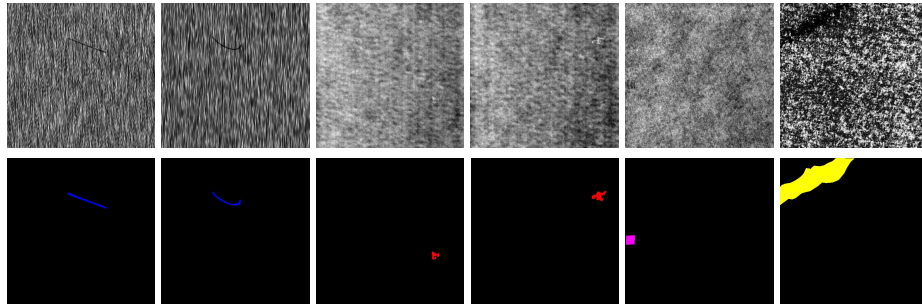

Fig. 6: The annotation comparison of the "texture surface" in DAGM dataset. The first row shows the defect image. Row 2 shows our improved annotation. **Best viewed in color.**
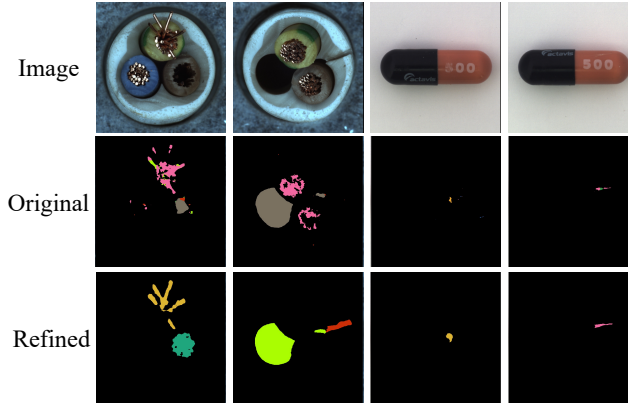
Fig. 7: Segmentation result comparison between model trained on our refined dataset and the original dataset of the "cable" and "capsule" class in MVTec dataset. "Original" denotes the segmentation masks produced by the model trained on the original dataset. "Refined" denotes the segmentation masks produced by the model trained on our refined dataset. We show the model trained with our dataset exhibits improved granularity and high quality. **Best viewed in color.**

2   Defect Generation

Implementation details In this section, we will first elaborate on the architecture of Defect-Gen. Then we will go over the dataset and training settings of our model. Lastly, we quantitatively compared it with other methods to demonstrate the superiority of our method.

Experimental Settings Since there was no train-test split in MVTec AD dataset, to train both large and small diffusion models, we employed 5 images for each defective type per object, which is the same as our segmentation training setting. For VISION, DAGM2007, and Cotton-Fabric, we use the pre-split
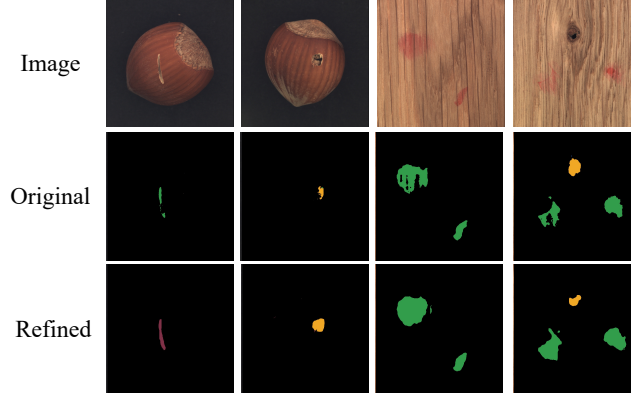
Fig. 8: Segmentation result comparison between model trained on our refined dataset and the original dataset of the "hazelnut" and "wood" class in MVTec dataset. "Original" denotes the segmentation masks produced by the model trained on the original dataset. "Refined" denotes the segmentation masks produced by the model trained on our refined dataset. We show the model trained with our dataset exhibits improved granularity and high quality. **Best viewed in color.**

Fig. 9: Segmentation result comparison between model trained on our refined dataset and the original dataset of the "Capacitor" and "Wood" class in the VISION dataset. "Original" denotes the segmentation masks produced by the model trained on the original dataset. "Refined" denotes the segmentation masks produced by the model trained on our refined dataset. We show the model trained with our dataset exhibits improved granularity and high quality. **Best viewed in color.**
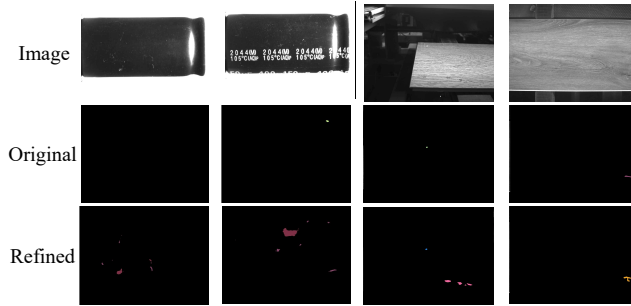
training set. Table 1 to 4 show the architectures of the large and small-receptive-field models. The training of diffusion models is performed on four 3090 GPUs, with a batch size of 2, a learning rate of $1e-4$, and a training iteration number of 150,000. We utilize the Adam optimizer with a weight decay of $2e-3$.

Table 1: Upsampling Block

Layer Type	Input size	Output size	Norm	Activation
ResBlock $\times$ 2	$H \times W \times C$	$H \times W \times C$	GN	SiLU
Interpolation	$H \times W \times C$	$2H \times 2W \times \frac{C}{2}$	None	None

Table 2: Downsampling Block

Layer Type	Input size	Output size	Norm	Activation
ResBlock $\times$ 2	$H \times W \times C$	$H \times W \times C$	GN	SiLU
Avg_pool 2×2	$H \times W \times C$	$\frac{H}{2} \times \frac{W}{2} \times 2C$	None	None

Parameter analysis As we discuss in Sec.3.4.2, our model has two key hyper-parameters: the switch timestep u and the receptive field of the small model. Both of them can control the trade-off between fidelity and diversity. We use FID to measure the generation fidelity. Since there is no existing metric to effectively measure the generation diversity, we used LPIPS score to indicate such. A higher LPIPS score with a similar FID score demonstrated a higher diversity in the dataset. Table 5 shows the FID and LPIPS for different u and receptive fields. As shown, when u increases, fidelity increases while diversity decreases. Similarly, when the receptive field switches from small to large, the same trend occurs. Empirically, we use u=50 and the medium receptive field to achieve a good trade-off between FID and LPIPS.

Table 3: Architecture for Large receptive fields model.

Layer Type	Resolution	# of Channels	Norm	Activation
InConv	256	4	GN	SiLU
DownSampleBlock	256	192	None	None
DownSampleBlock	128	384	None	None
DownSampleBlock	64	768	None	None
DownSampleBlock	16	1536	None	None
UpSampleBlock	16	768	None	None
UpSampleBlock	64	384	None	None
UpSampleBlock	128	192	None	None
UpSampleBlock	256	96	None	None
OutConv	256	4	GN	SiLU

Table 4: Architecture for Small receptive fields model.

Layer Type	Resolution	# of Channels	Norm	Activation
InConv	256	4	GN	SiLU
DownSampleBlock	256	192	None	None
DownSampleBlock	128	384	None	None
UpSampleBlock	128	192	None	None
UpSampleBlock	256	96	None	None
OutConv	256	4	GN	SiLU

Table 5: The table shows the trade-off between diversity and image quality of the capsule class. The column represents 3 different receptive field sizes, large, medium, and small, and the respective down-sampling blocks are 6, 3, 2. The row represents the timesteps(v) used for the small receptive field model.

	u	25	50	75	100	400	700
Small	FID ↓	115.2754	93.2839	80.8040	79.6411	82.5127	78.4115
	LPIPS ↑	0.3981	0.3666	0.3537	0.3523	0.3467	0.3460
Medium	FID ↓	69.9419	57.5374	57.3961	57.8977	57.426	57.006
	LPIPS ↑	0.3473	0.3458	0.3450	0.3417	0.3392	0.3381
Large	FID ↓	59.085	56.6246	56.7247	56.2493	55.7226	54.0529
	LPIPS ↑	0.2914	0.2870	0.2866	0.2853	0.2832	0.2814

Quantitative Evaluation We have compared the segmentation performance boost across different methods on the original MVTec dataset. GAN-based methods were excluded since they hardly generate realistic images, further disrupting the original data distribution. Results for defect segmentation are shown in Table. 6. The first column shows the defect segmentation mIoU score with only the original training data. The rest of each column presents defect segmentation performance with original training data pairs and the augmented pairs generated by different synthesis methods. SinDiffusion dropped the mIoU score, due to the incorrectly structured output images and mislabeled masks. However, it can slightly improve the segmentation performance for certain classes like "Carpet", "Grid", "Leather", "Tile" and "Wood". Since those classes do not contain any industrial parts and thus do not require any global structure information during synthesizing. DDPM-generated samples can boost the performance score, however, due to the lack of diversity during generation, the increase in performance is limited.

Table 6: Quantitative comparison on segmentation performance between sinDiffusion, DDPM, and our method. To demonstrate the effectiveness of our method on other dataset besides Defect Spectrum, the comparison was made on the original MVTec dataset

	w/o any AUG	sinDiffusion	DDPM	Ours
capsule	75.47	76.25	79.21	82.20
bottle	67.54	70.52	67.32	73.75
carpet	67.33	72.89	68.94	74.27
screw	53.12	49.66	60.12	58.78
grid	59.68	61.59	60.68	62.14
cable	46.28	41.75	48.28	49.14
hazelnut	69.25	65.65	69.25	71.46
leather	66.39	66.91	66.39	66.80
metal_nut	69.56	63.5	68.57	74.4
pill	69.71	66.75	70.14	73.19
tile	70.33	72.43	71.23	73.58
toothbrush	68.26	64.26	68.09	70.14
transistor	44.31	47.16	44.37	47.47
wood	65.33	70.25	64.93	68.55
zipper	67.62	63.12	68.61	70.48
mean	64.01	63.51	65.07	67.76

3   Visual Generation Results

We have included more defect generation results along with their masks as shown in Figure 10 to 15 below.
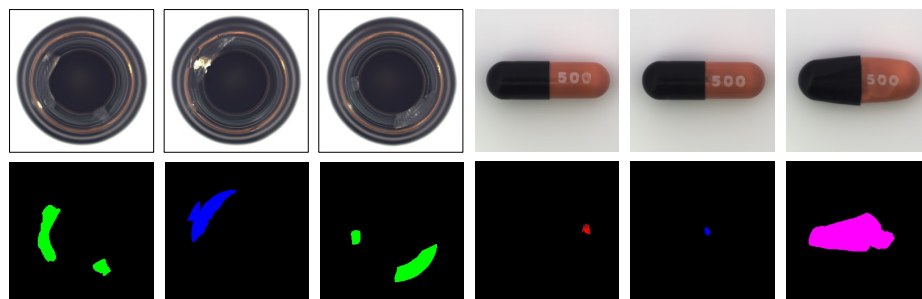

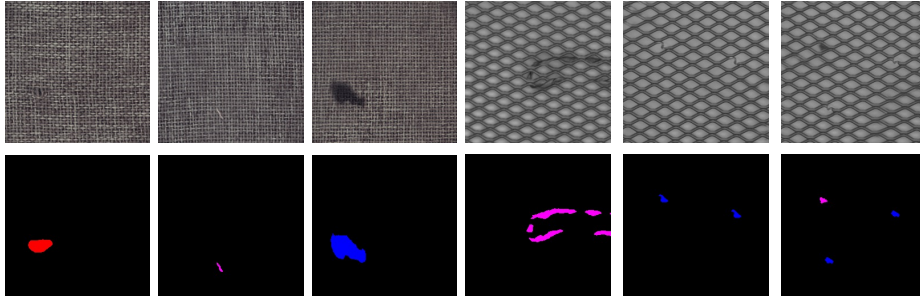**Fig. 10:** The generated images and masks of the "bottle" and "capsule" class. **Best viewed in color.**

Fig. 11: The generated images and masks of the "carpet" and "grid" class. **Best viewed in color.**

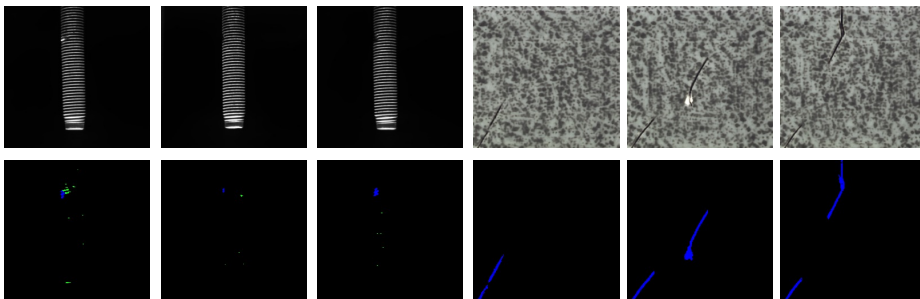**Fig. 12:** The generated images and masks of the "pill" and "ring" class. **Best viewed in color.**

Fig. 13: The generated images and masks of the "screw" and "tile" class. **Best viewed in color.**
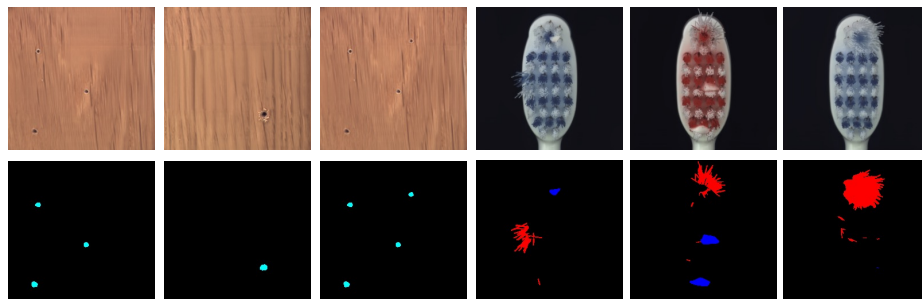
Fig. 14: The generated images and masks of the "wood" and "toothbrush" class. **Best viewed in color.**
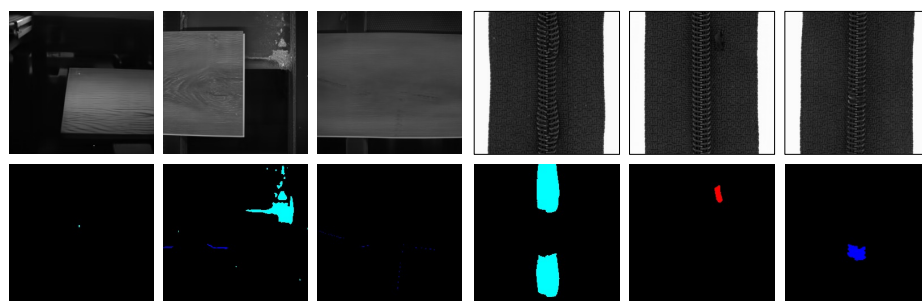

Fig. 15: The generated images and masks of the "wood-surface" and "zipper" class. **Best viewed in color.**