# 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.

## 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.

⋆ These authors contributed equally to this work.

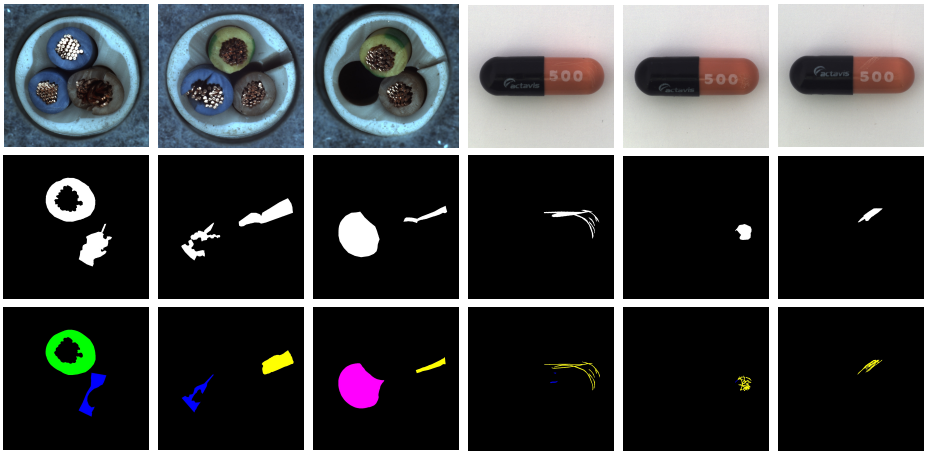

**Fig. 1:** The annotation comparison of the “cable" and “capsule" class in MVTec dataset. The first row shows the defect image. Rows 2 and 3 show the original annotation and our improved annotation. **Best viewed in color.**

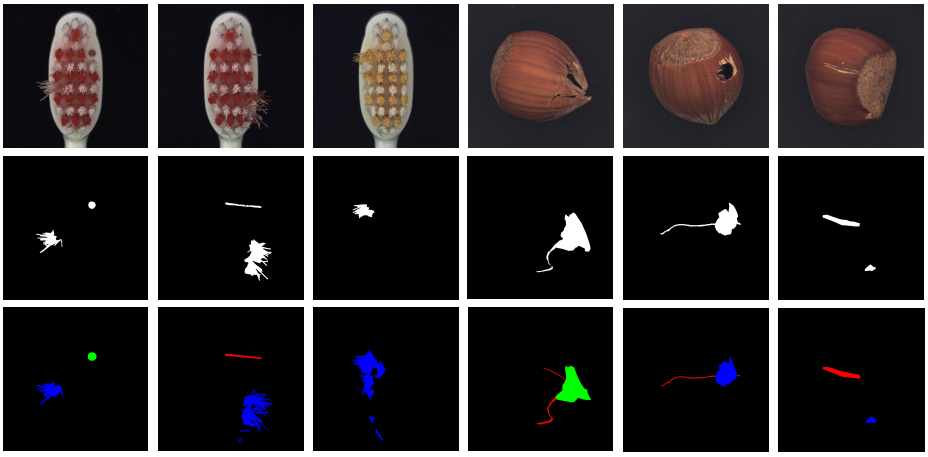

**Fig. 2:** The annotation comparison of the “toothbrush" and “hazelnut" class in MVTec dataset. The first row shows the defect image. Rows 2 and 3 show the original annotation and our improved annotation. **Best viewed in color.**

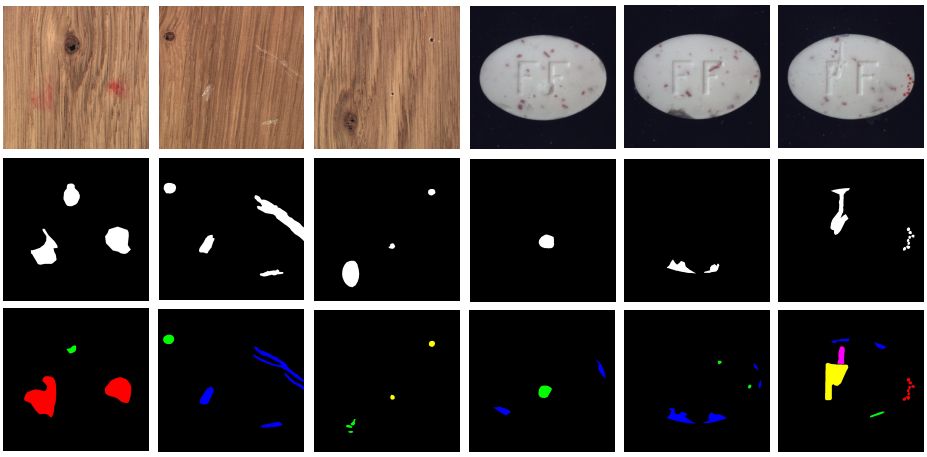

**Fig. 3:** The annotation comparison of the “wood" and “pill" class in MVTec dataset. The first row shows the defect image. Row 2 and 3 show the original annotation and our improved annotation. **Best viewed in color.**

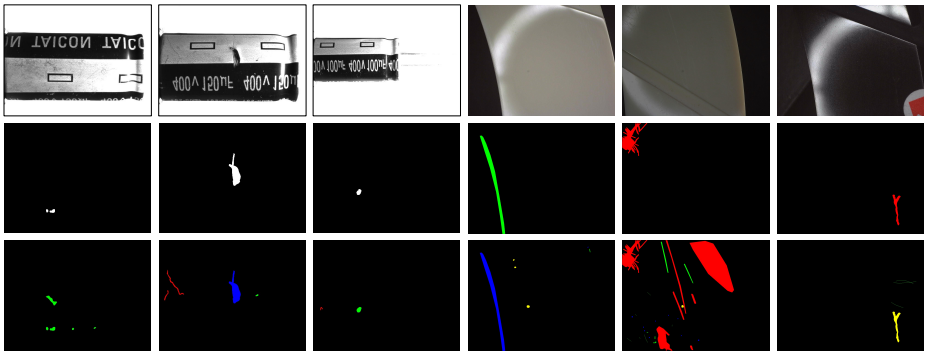

**Fig. 4:** The annotation comparison of the “capacitor" and “ring" class in VISION dataset. The first row shows the defect image. Rows 2 and 3 show the original annotation and our improved annotation. **Best viewed in color.**

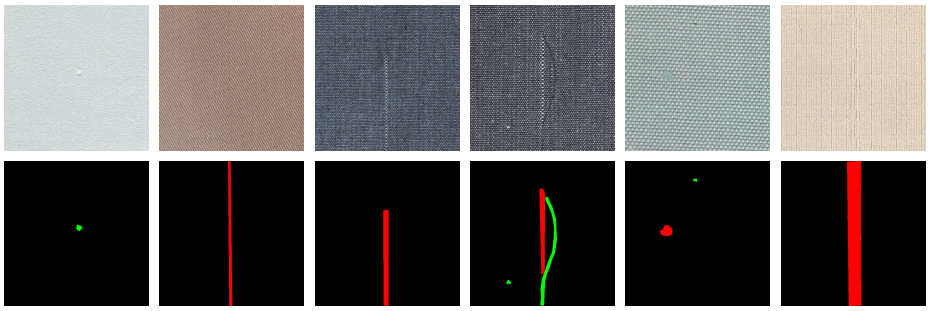

**Fig. 5:** The annotation comparison of the “cotton fabric" class in the COTTON dataset. The first row shows the defect image. Row 2 shows our improved annotation. **Best viewed in color.**

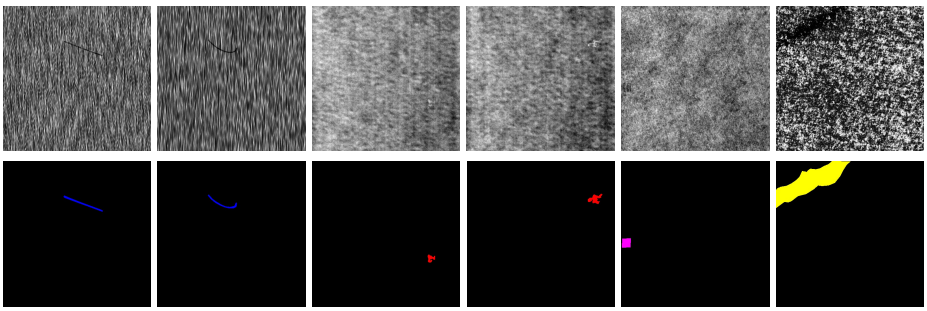

**Fig. 6:** The annotation comparison of the “texture surface" in DAGM dataset. The first row shows the defect image. Row 2 shows our improved annotation. **Best viewed in color.**


**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.**

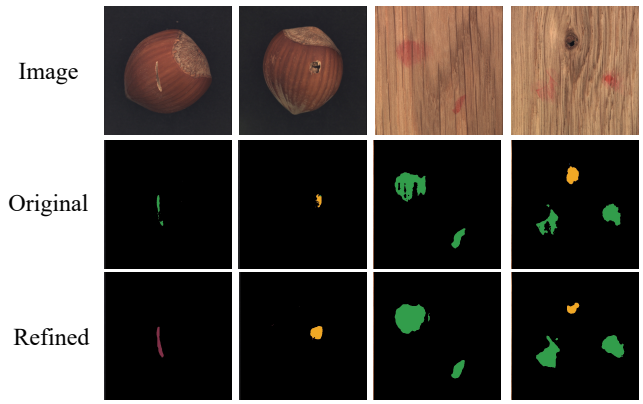


**Fig. 8:** Segmentation result comparison between model trained on our refined dataset and the original dataset of the “hazelnut" and “wood" class in MVTec dataset. “Original” denotes the segmentation masks produced by the model trained on the original dataset. “Refined” denotes the segmentation masks 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.**

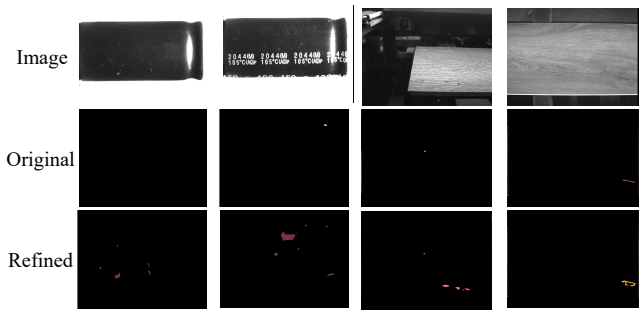


**Fig. 9:** Segmentation result comparison between model trained on our refined dataset and the original dataset of the "Capacitor" and "Wood" class in the VISION dataset. "Original" denotes the segmentation masks produced by the model trained on the original dataset. "Refined" denotes the segmentation 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.**

**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 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 $2 \times 2$ | $H \times W \times C$ | $\frac{H}{2} \times \frac{W}{2} \times 2C$ | None | None |

**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 |

**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 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.**

## References

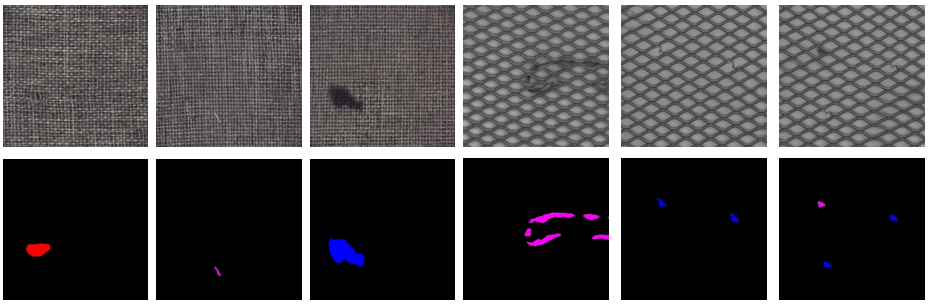

**Fig. 11:** The generated images and masks of the “carpet" and “grid" class. **Best viewed in color.**

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

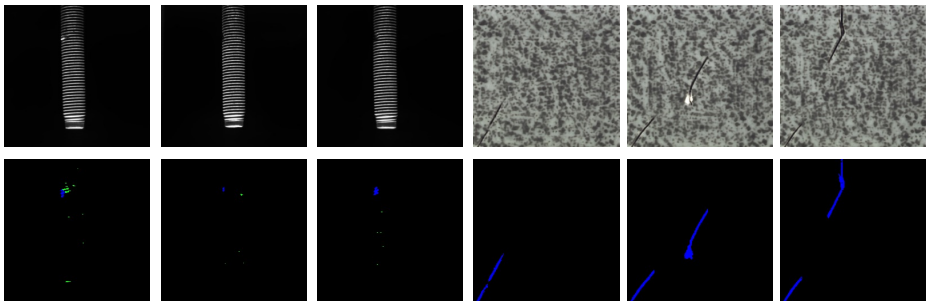

**Fig. 13:** The generated images and masks of the “screw" and “tile" class. **Best viewed in color.**

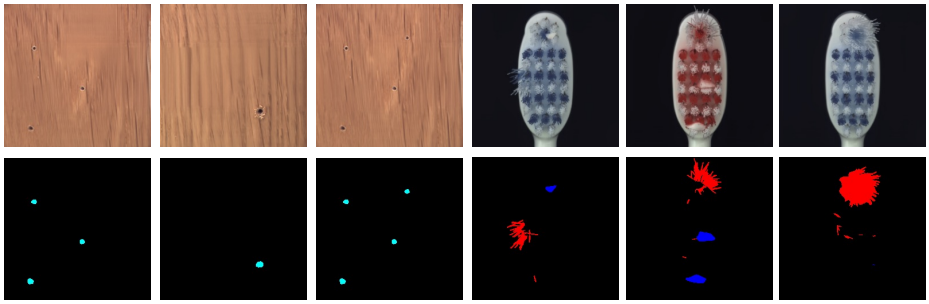

**Fig. 14:** The generated images and masks of the “wood" and “toothbrush" class. **Best viewed in color.**

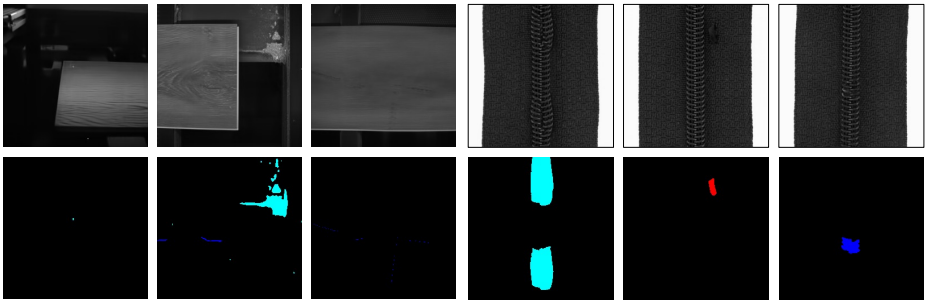

**Fig. 15:** The generated images and masks of the “wood-surface" and “zipper" class. **Best viewed in color.**