SUPPLEMENTARY MATERIAL FOR DEFECT SPECTRUM: A GRANULAR LOOK OF LARGE-SCALE DEFECT DATASETS WITH RICH SEMANTICS

Anonymous authors
Paper under double-blind review

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

A EXTENDED SYNTHETIC DATA EXPERIMENTS

We extended our experiments to further demonstrate the effectiveness of the usage of the synthetic data by incorporating 3 more baselines: DeepLabv3+, Mask2Former, and Mit-B0. Results in Table 1 show a large performance increase in both MVTec and Cotton datasets, the increase is comparatively smaller in the VISION dataset, however, such increase is demonstrated in each of the sub-classes.

Table 1: Performance (mIoU) comparison between models trained with and without synthetic data. The bolded text indicates results with synthetic data.

	MVTec	VISION	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**

B INCREASING RATIO OF SYNTHETIC DATA

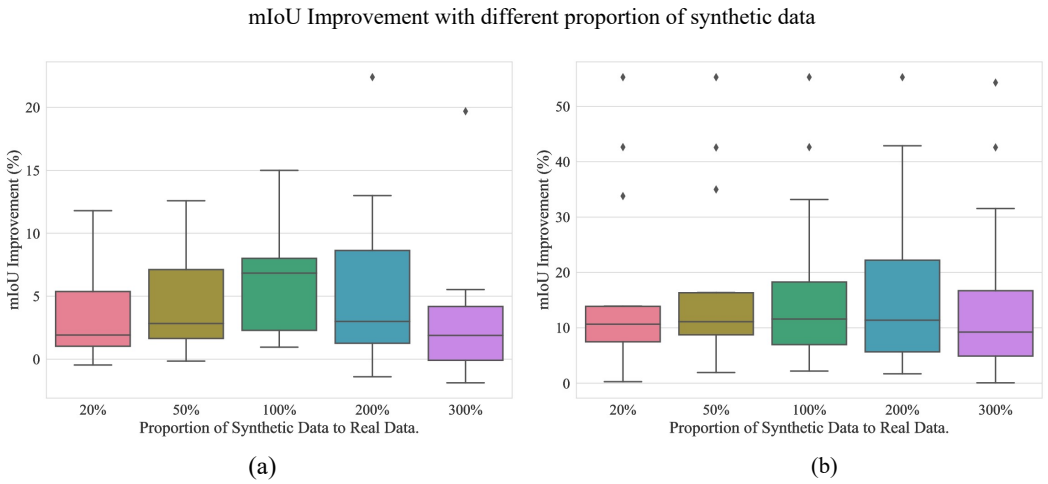

Figure 1: Improvement in mIoU with different proportions of synthetic Data. This experiment is done on Defect Spectrum (MVTec) with DeeplabV3+ and MiT-B0 shown as (a) and (b) respectively.

We further increase the synthetic data ratio to test the impact it has on model performance. Figure 1 shows the performance improvement over different quantities of synthetic data using DeepLabV3+

and Mit-B0. 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%.

Table 2: Uniform Upsampling strategy

	DeepLabV3+	MiT-B0
Baseline	51.58	46.45
With Synthetic	**54.87**	**56.21**

C SAMPLING STRATEGY

We employ a new sampling strategy that uniformly up-scales all training sets to 150 images. So those with limited data gain more synthetic data and vice versa. As shown in Table 2, with this sampling strategy, MiT-B0 models show a great leap in performance, from 46.45 to 56.21, and it even surpasses the original SOTA DeepLabV3+.

D COMPARISON BETWEEN REFINED AND BASELINE DATASET

As shown in Figure 3, we compared the segmentation model trained on the baseline dataset and our refined dataset. The results with our annotation demonstrate a better granularity while having rich semantics (different defective classes).

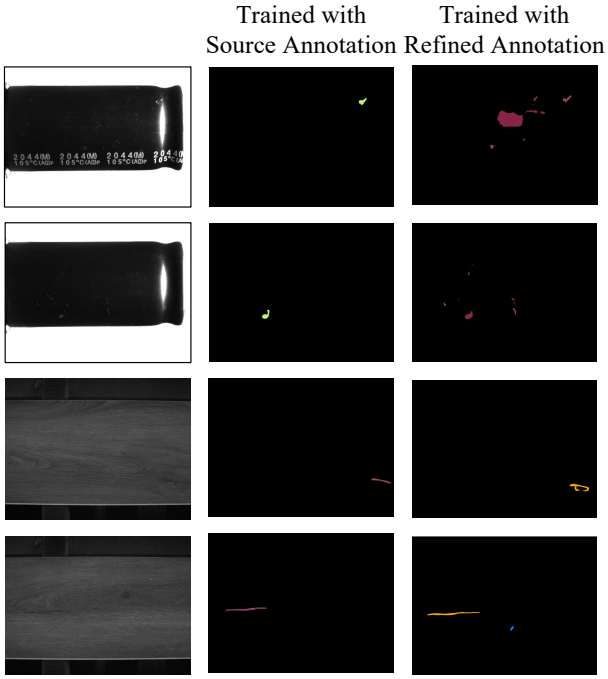

Figure 2: Qualitative comparison between the segmentation model trained on our refined dataset and the base dataset. We show our method can exhibit diversity while maintaining high quality. **Best viewed in color.**

E Comparison between Unsupervised AD and Ours

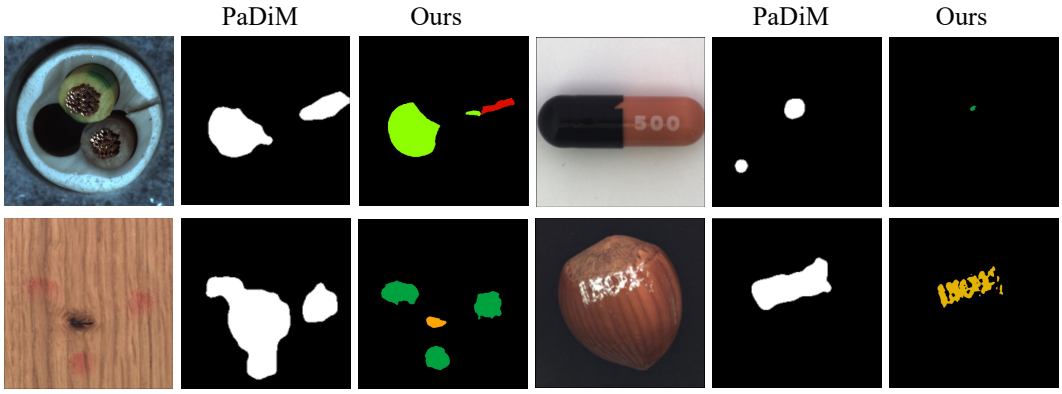

Figure 3: Qualitative comparison between the unsupervised AD baseline PaDiM and ours. Ours is trained on refined MVTec using MiT-B0. The segmentation results using our refined dataset highlighted different defective types with superior granularity. **Best viewed in color.**

F Incorporating normal data

We conducted additional experiments on refined-MVTec to analyze the impact of integrating normal data. Our findings indicate that adding normal data indeed improves the mIoU, addressing the issue of over-penalizing non-defective areas.

Table 3: Combining different percentage of normal(defect-free) data. The source indicates the refined MVTec training set without any normal data.

	Source	+20% normal	+100% normal	+200% normal	+300% normal
Mean	51.58	**53.87**	53.06	53.38	53.04

G Annotation Comparision

In this section, we present a visual comparison between ours (the last row) and the original datasets' annotation. Figure 4, 5, 6 shows the comparison of the MVTec dataset, we re-classify the defects based on their type and enabled more semantic abundance. As for Figure 7 of 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 8, 9.

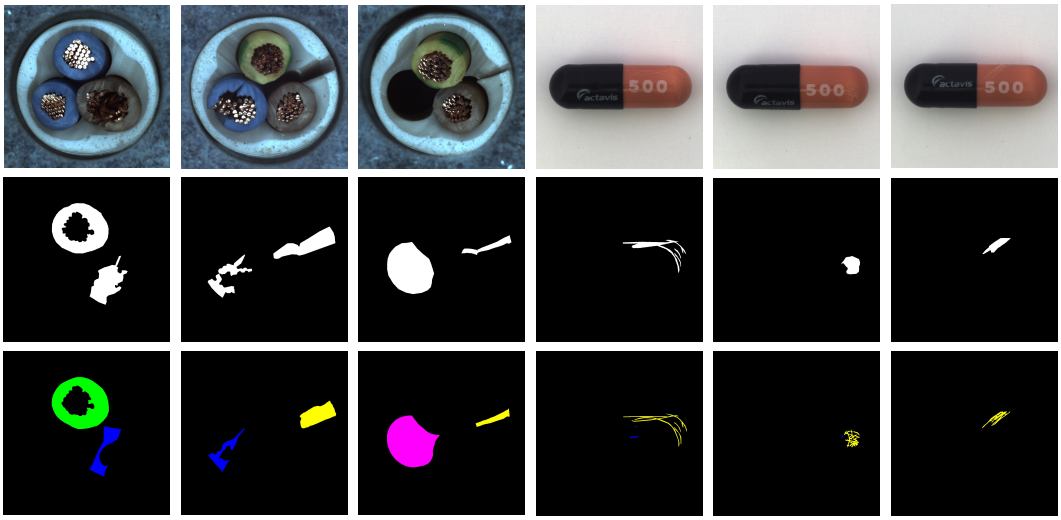

Figure 4: The annotation comparison of the "cable" and "capsule" class in MVTec dataset. The first row shows the defect image. Row 2 and 3 shows the original annotation and our improved annotation. **Best viewed in color.**

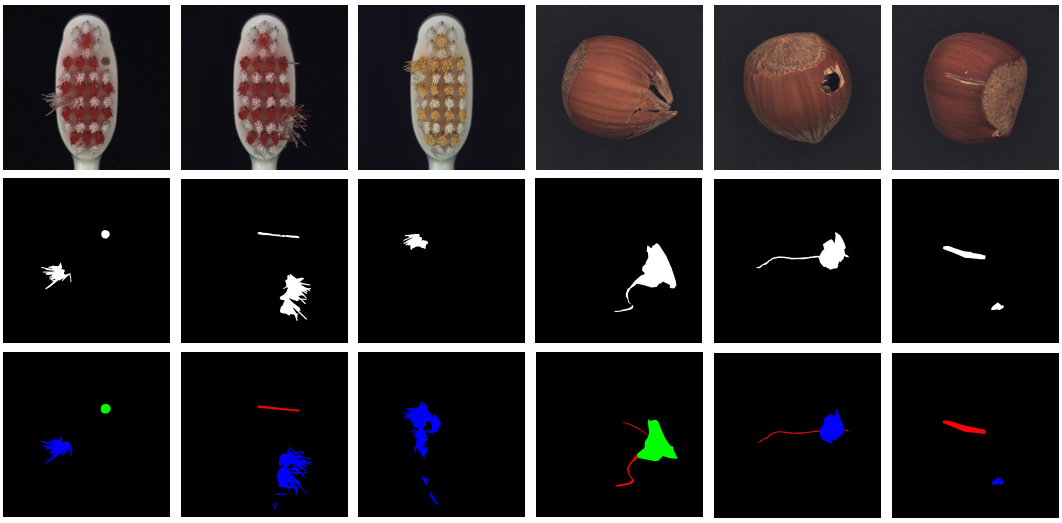

Figure 5: The annotation comparison of the "toothbrush" and "hazelnut" class in MVTec dataset. The first row shows the defect image. Row 2 and 3 shows the original annotation and our improved annotation. **Best viewed in color.**

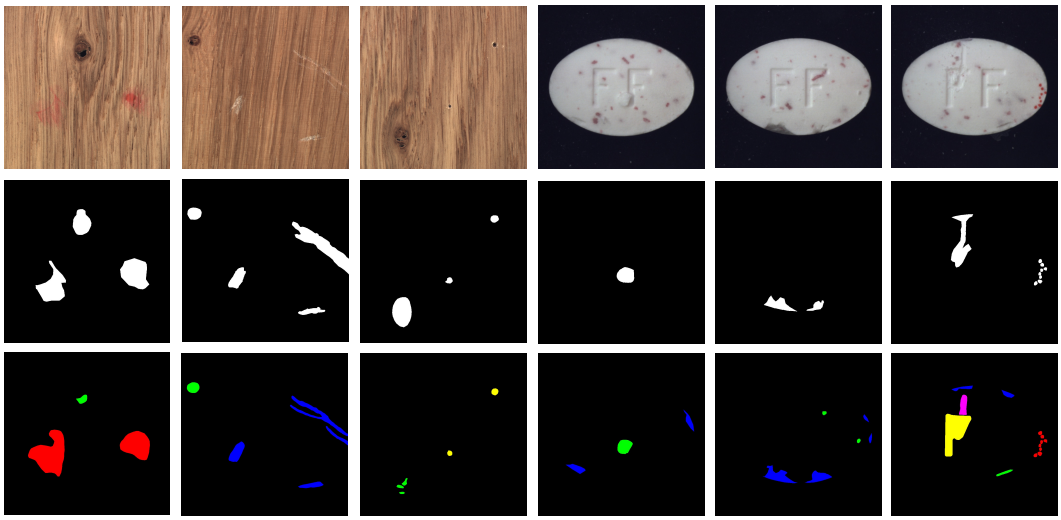

Figure 6: The annotation comparison of the "wood" and "pill" class in MVTec dataset. The first row shows the defect image. Row 2 and 3 shows the original annotation and our improved annotation. **Best viewed in color.**

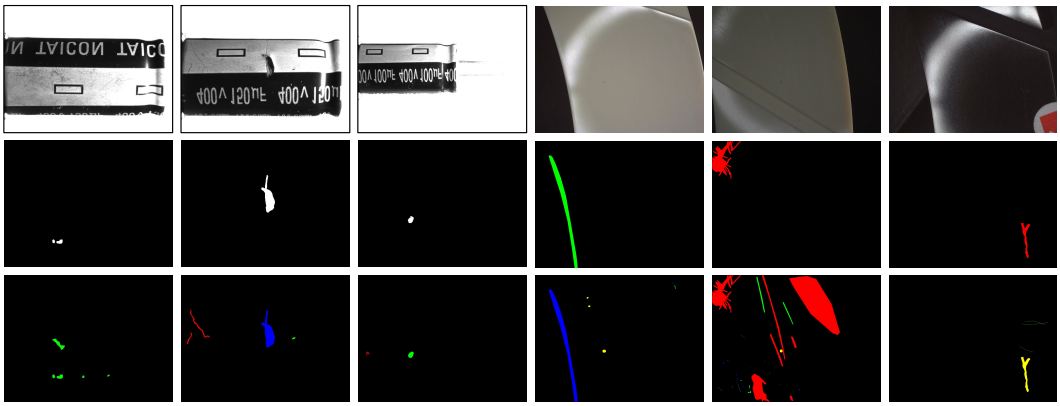

Figure 7: The annotation comparison of the "capacitor" and "ring" class in VISION dataset. The first row shows the defect image. Row 2 and 3 shows the original annotation and our improved annotation. **Best viewed in color.**

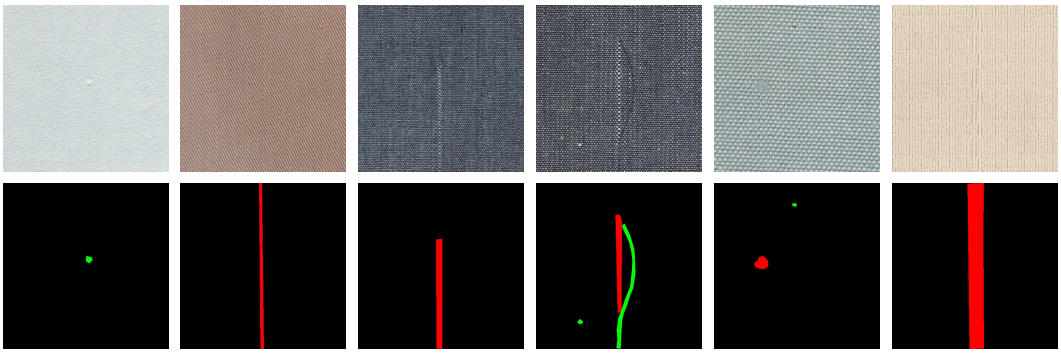

Figure 8: The annotation comparison of the "cotton fabric" class in the COTTON dataset. The first row shows the defect image. Row 2 shows our improved annotation. **Best viewed in color.**

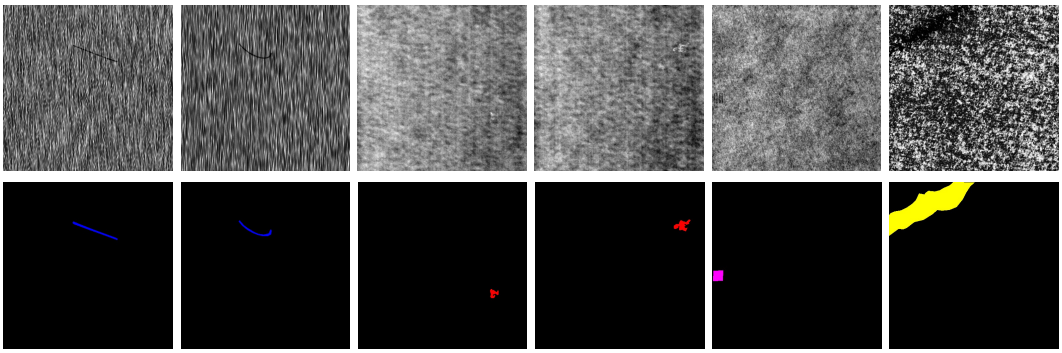

Figure 9: The annotation comparison of the "texture surface" in DAGM dataset. The first row shows the defect image. Row 2 shows our improved annotation. **Best viewed in color.**

H DEFECT GENERATION

H.0.1 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 training set. Table 4 to 7 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 4: 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 5: 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 hyperparameters: 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

Table 6: 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 7: 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

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 8 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 8: 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

H.0.2 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. 9. 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 9: 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

H.0.3 DEMONSTRATION OF PATCH-LEVEL MODELING

Figure 10 demonstrates the effectiveness of this strategy. Overall, the generated sample is different from the training samples, while at the patch level, we can find some connections in between.

I VISUAL GENERATION RESULTS

We have included more defect generation results along with their masks as shown in Figure 11 to 16 below.

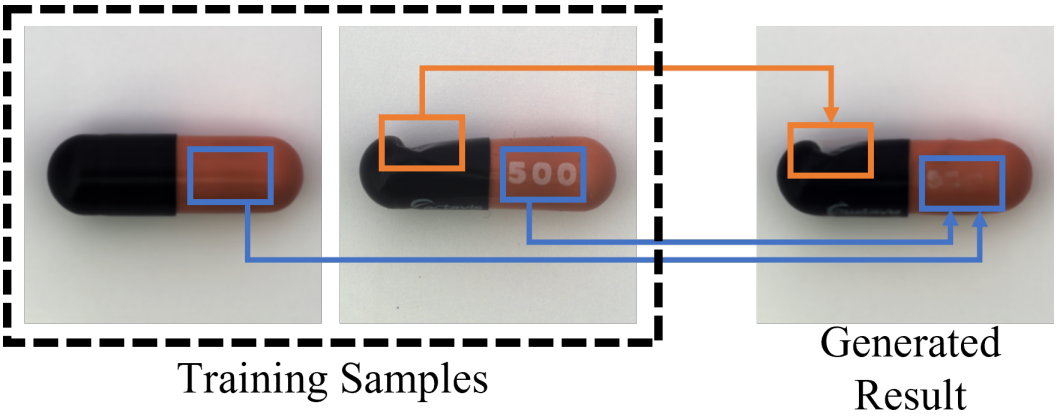

Figure 10: The property of patch-level modeling. The right image is generated from the small-receptive-field model, and the two left images are the two most similar images from the training set.

Figure 11: The generated images and masks of the "bottle" and "capsule" class. **Best viewed in color.**

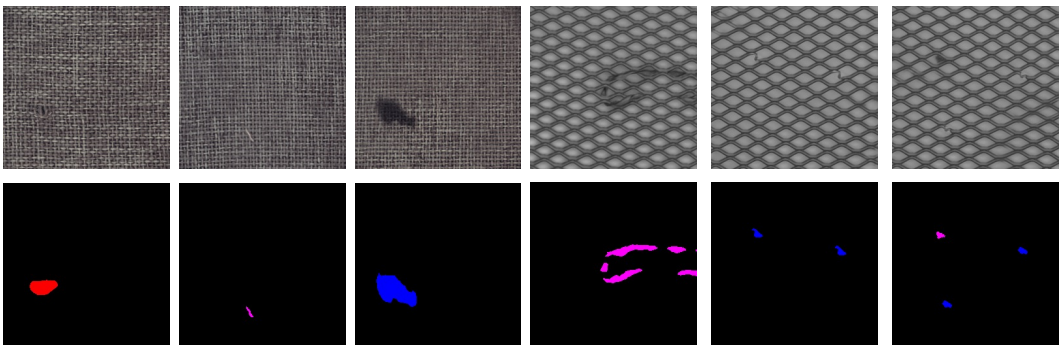

Figure 12: The generated images and masks of the "carpet" and "grid" class. **Best viewed in color.**

Figure 13: The generated images and masks of the “pill” and “ring” class. **Best viewed in color.**

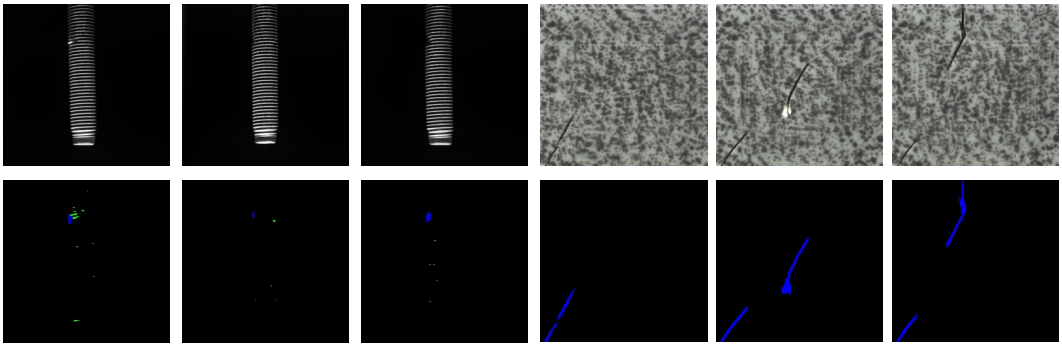

Figure 14: The generated images and masks of the “screw” and “tile” class. **Best viewed in color.**

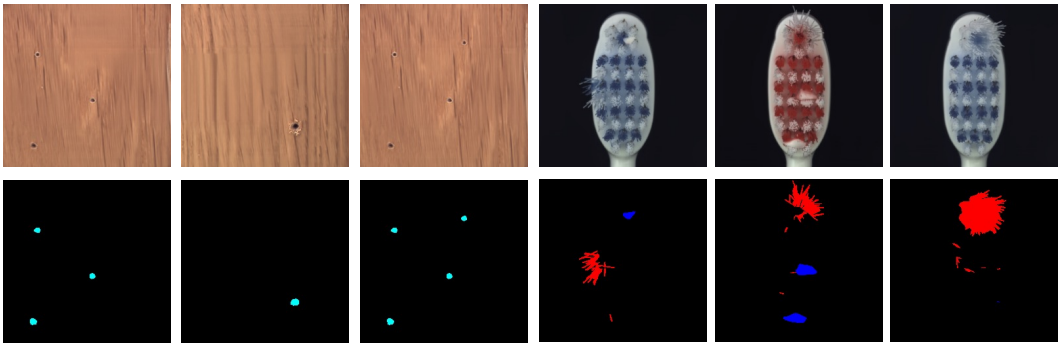

Figure 15: The generated images and masks of the “wood” and “toothbrush” class. **Best viewed in color.**

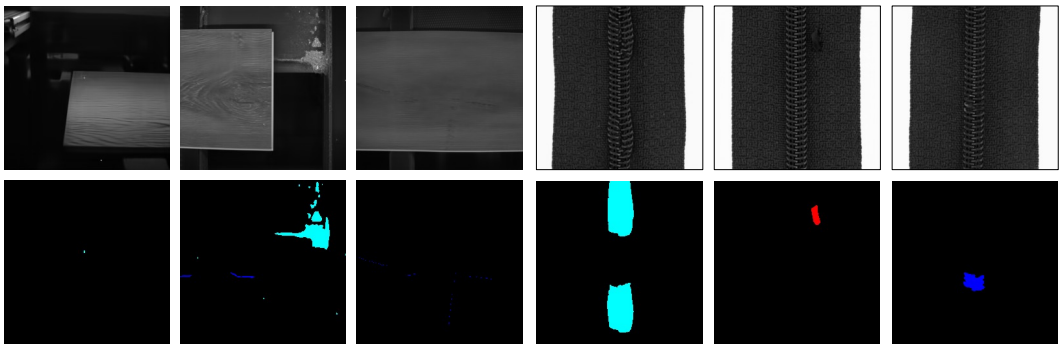

Figure 16: The generated images and masks of the “wood-surface” and “zipper” class. **Best viewed in color.**