1 Experimental Details

1.1 feature extractor

During the self-supervised training phase, we use the pre-trained ConvNeXt as the feature extractor to extract the differences between the augmented patches and the normal regions in the constructed samples. We use different sizes of ConvNeXt pre-trained models, and to ensure consistency between the source and target domains, we freeze the network parameters within the feature extractor at the start of training and subsequently fine-tune, with the default setting of 20 for the number of freezing epochs. Fig. 1 shows the start-up AUC and the after-trained AUC of the different ConvNeXt pre-trained models.

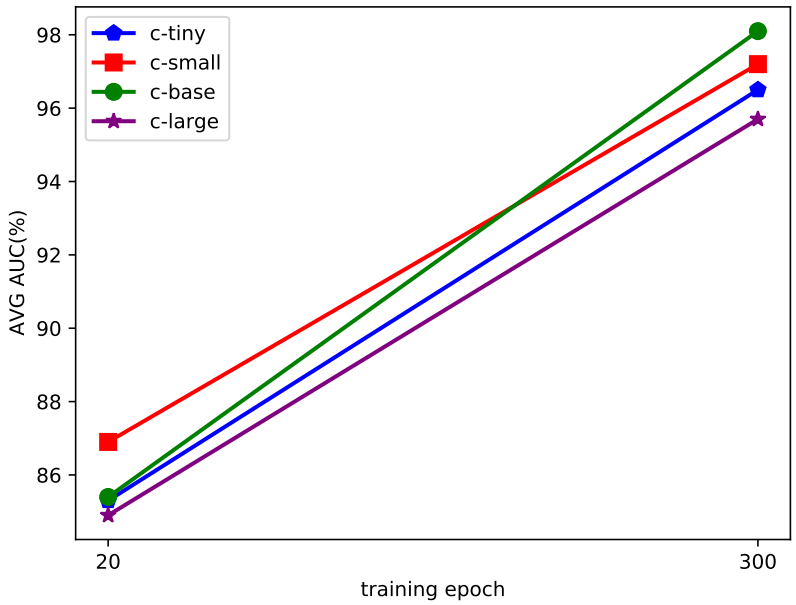

Fig. 1. Detection AUC after fine-tuning of different ConvNeXt pre-training models. Epoch 20 refer the end of freezing parameters.

1.2 data augmentation or not

When constructing anomaly samples, the augmented patches may come from other regions of the same image or from another image, and to provide better generality to the constructed anomaly patterns, we add data augmentation transforms to the augmented patches.

Fig. 2. Different data augmentation transforms are applied to the MVTEC, with the first row using a color transform and the second row using the Pepper Noise.

We apply color transformations and pepper noise on the augmented patches respectively. Color transformation includes brightness, contrast, saturation, hue. Fig. 2 shows some samples which are augmented by color transformation or Pepper Noise. The added noise increases the generalization of anomalies and provides support for the network to find differences between normal and abnormal patterns.

1.3 training samples

Table 1. Different proportions of positive and negative samples in the training phaseo on MVTEC AD.

Category	Proportion(negetive:positive)			
	1:0	**1:1**	**1:2**	**2:1**
object	85.1	**97.4**	96.5	95.3
texture	95.3	**99.6**	98.5	97.3
Overall AVG	88.5	**98.1**	97.2	96.0

During the self-supervised training phase, we mix the augmented samples with the same number of normal samples to participate in the training by default. We aim to explore whether different proportions of positive and negative samples during training will have an impact on the results and, at the extreme, whether augmented samples only will allow the feature extractor to learn more robust classification boundary. We conduct the following experiments.

As shown in the Table1, a certain number of positive samples need to be involved in the training phase, and if only the artificial negative samples used to

segment the abnormal patches, the network runs the risk of overfitting and thus misclassifying the normal regions as abnormal.

1.4 cutX transforms

As mentioned in paper, Fig. 3 show different data augmentation variants. Cutout masks an area of the image, CutPaste masks and fills it with content from other areas of the same image while CutMix fills it with other image blocks.

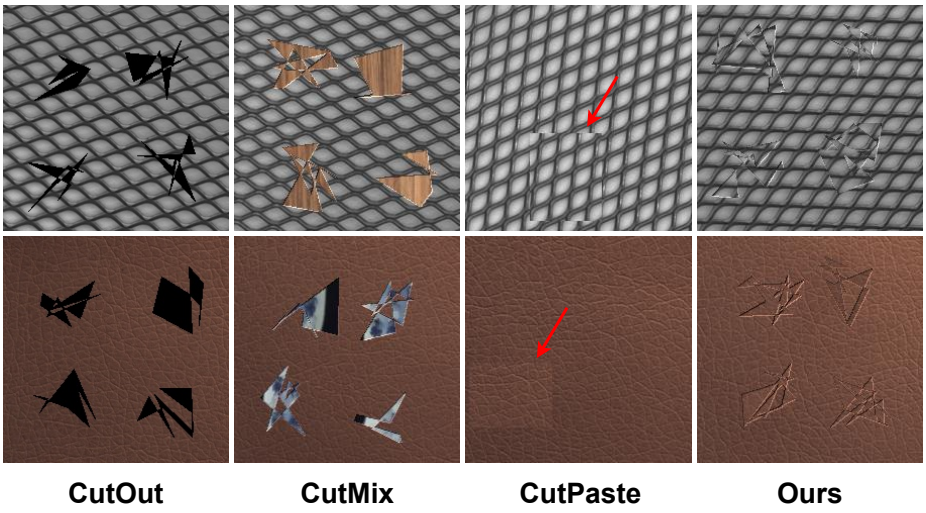

Fig. 3. Different data augmentation variants. From left to right are visualization of cutout, cutmix, cutpaste and our method operated on image.

2 Anomaly Segmentation Results

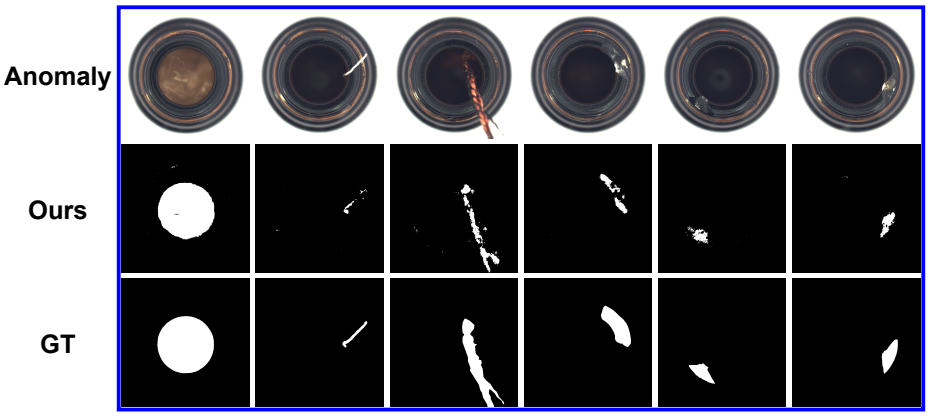

Fig. 4. Defect localization on bottle class of MVTec dataset.From top to bottom, input abnormal images, segmentation results generated by our method, and ground truth images

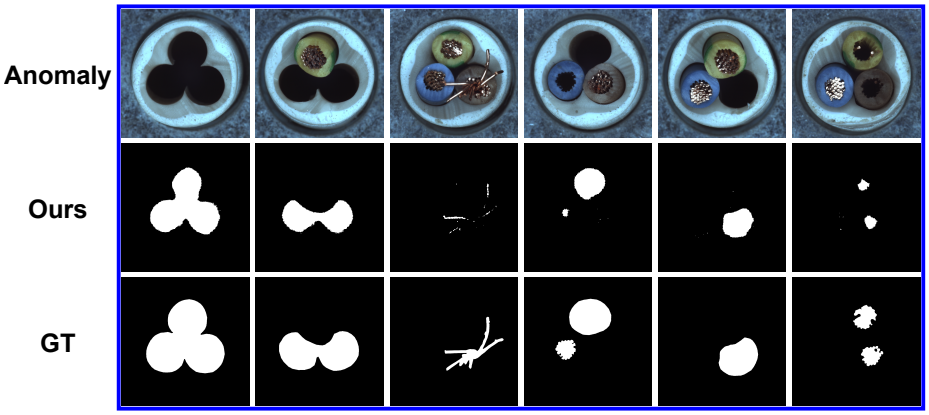

Fig. 5. Anomaly segmentation on cable class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

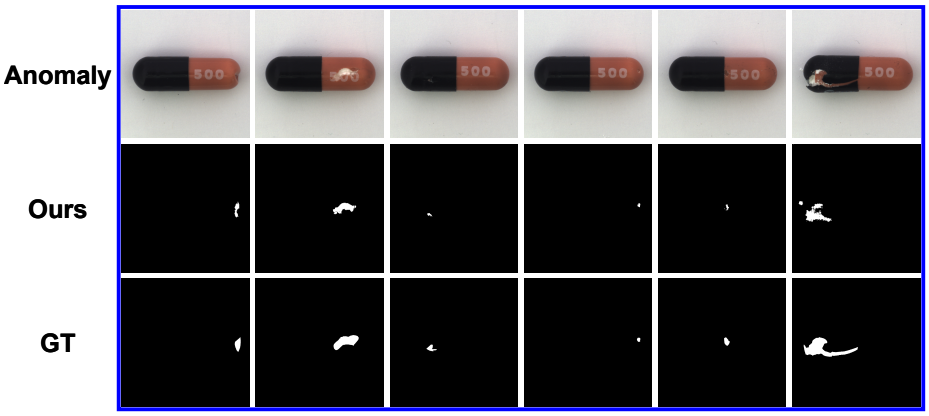

Fig. 6. Anomaly segmentation on capsule class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

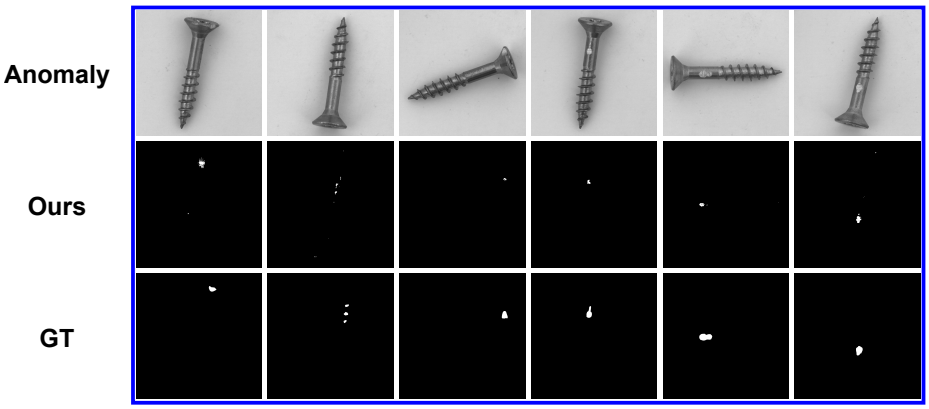

Fig. 7. Anomaly segmentation on screw class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

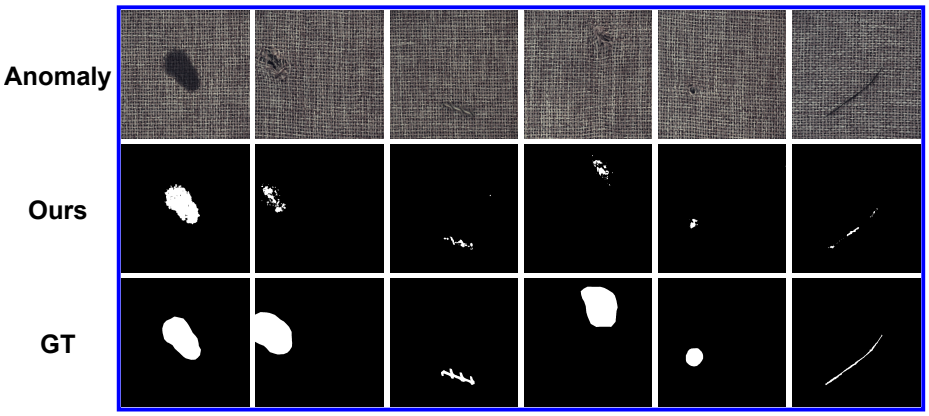

Fig. 8. Anomaly segmentation on carpet class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

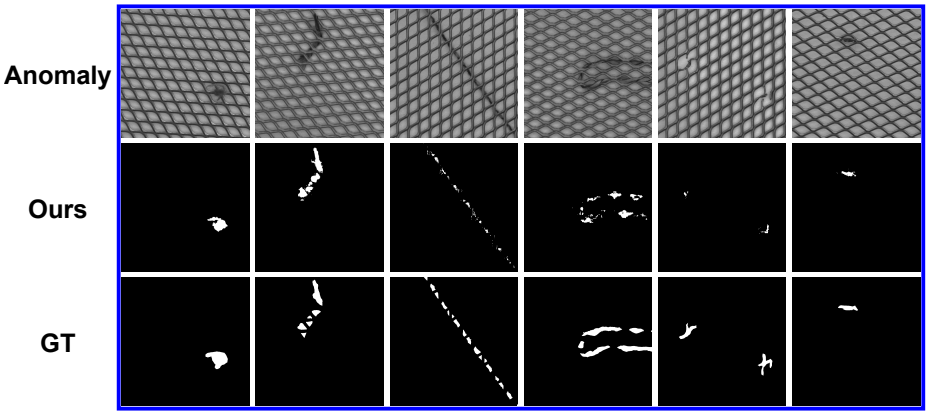

Fig. 9. Defect localization on bottle class of MVTec dataset.From top to bottom, input abnormal images, segmentation results generated by our method, and ground truth images

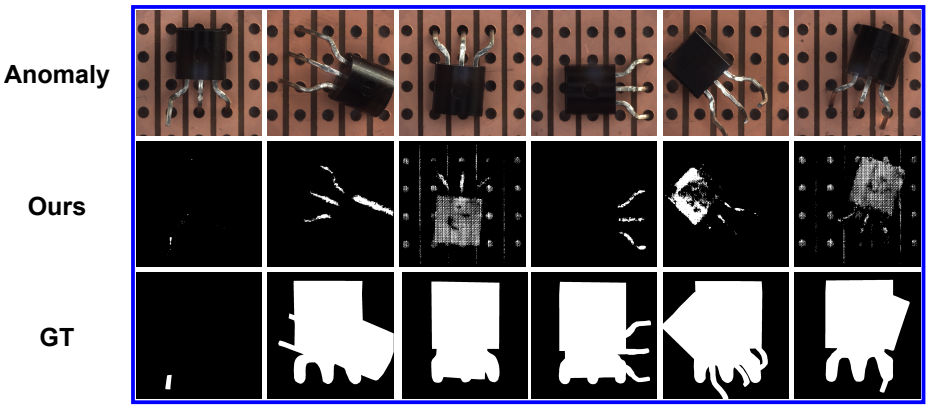

Fig. 10. Anomaly segmentation on transistor class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

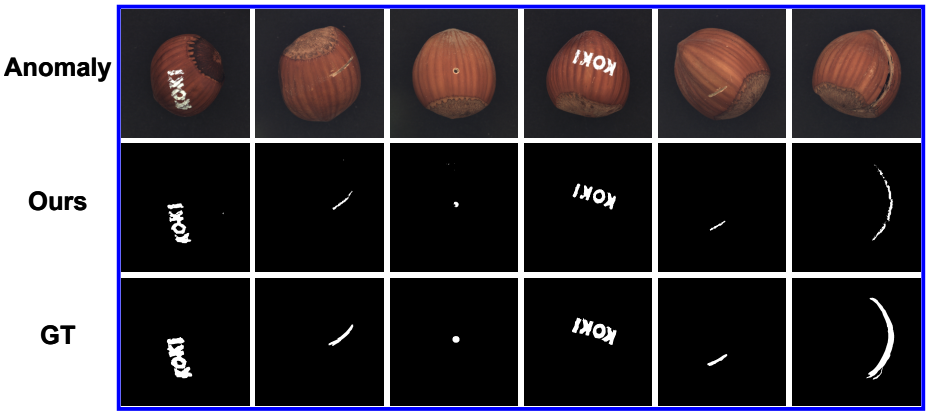

Fig. 11. Anomaly segmentation on hazelnut class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

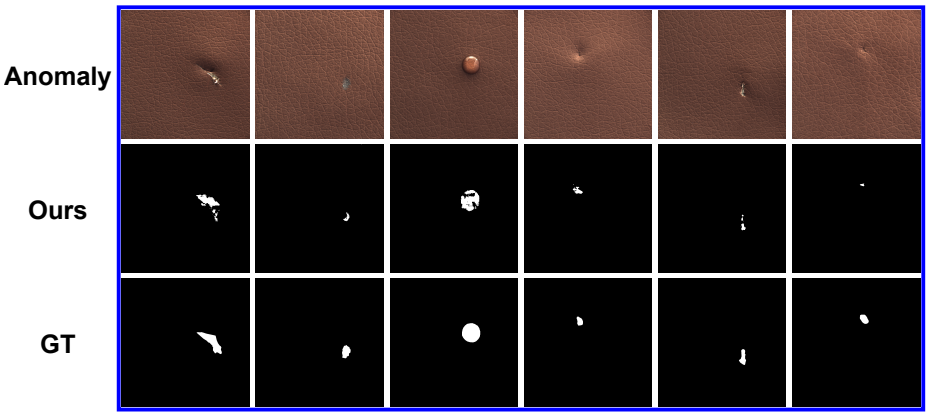

Fig. 12. Anomaly segmentation on leather class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

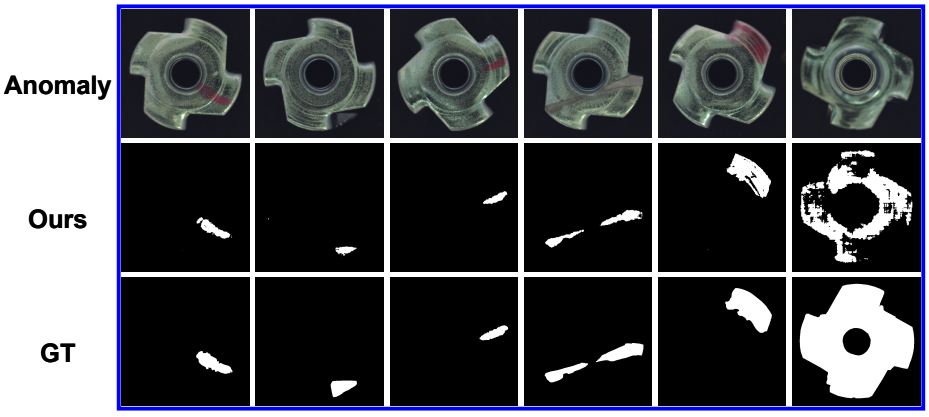

Fig. 13. Anomaly segmentation on metalnut class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

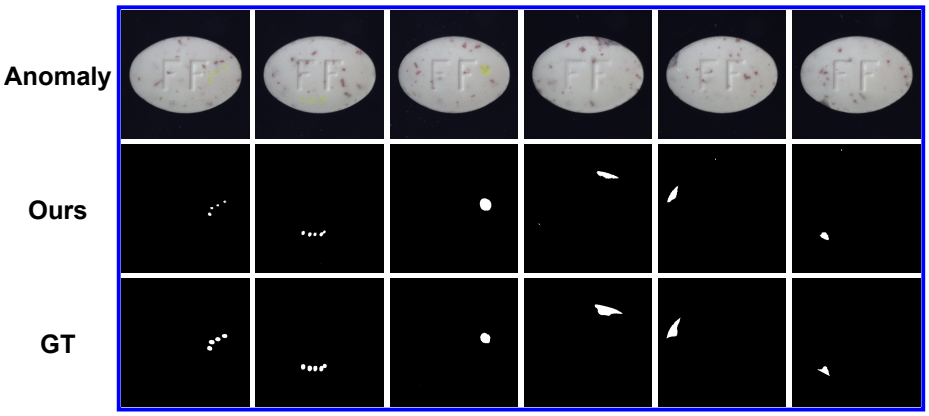

Fig. 14. Anomaly segmentation on cable class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

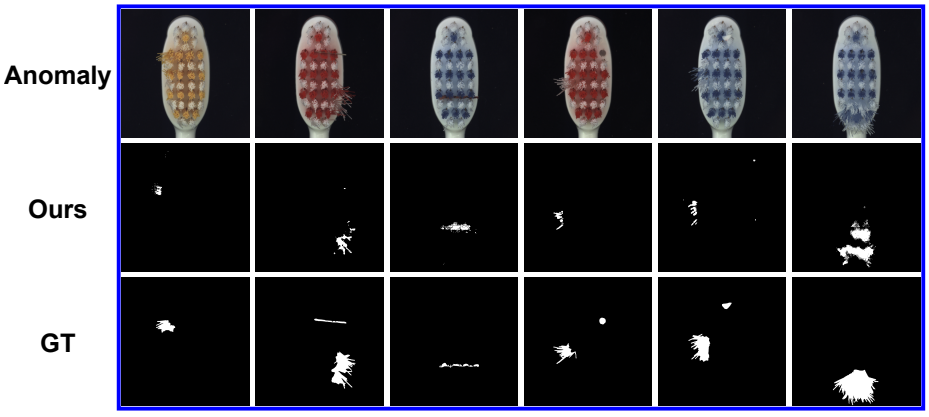

Fig. 15. Anomaly segmentation on toothbrush class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

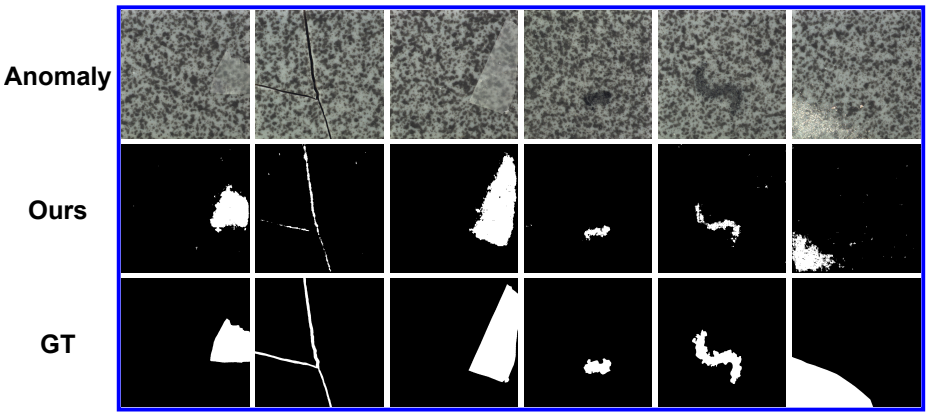

Fig. 16. Anomaly segmentation on tile class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

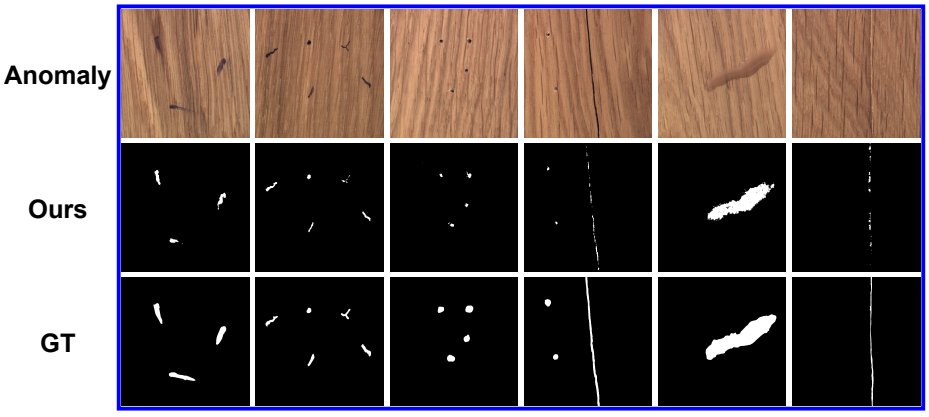

Fig. 17. Anomaly segmentation on wood class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images

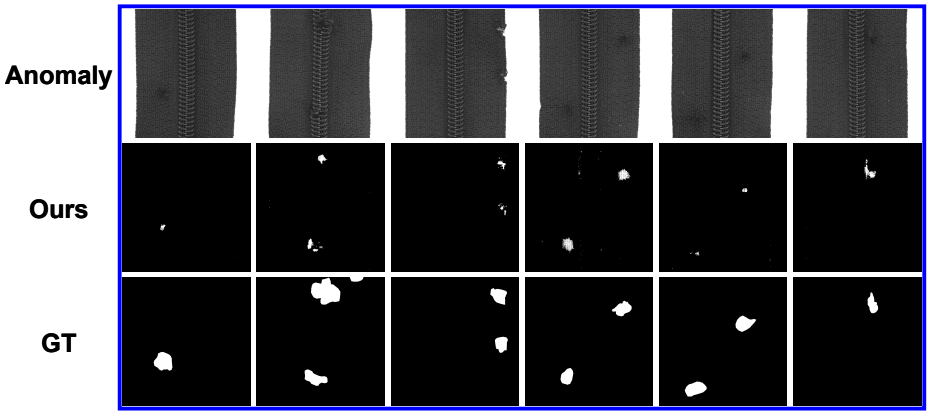

Fig. 18. Anomaly segmentation on zipper class of MVTec dataset.From top to bottom is, input abnormal images, segmentation results generated by our method, and ground truth images