Dinomaly: The *Less Is More* Philosophy in Multi-Class Unsupervised Anomaly Detection

Supplementary Material

A. Additional Related Work

Here, we discussed general methods for unsupervised anomlay detection. *Epistemic methods* are based on the assumption that the networks respond differently during inference between seen input and unseen input. Within this paradigm, *pixel reconstruction* methods assume that the networks trained on normal images can reconstruct anomaly-free regions well, but poorly for anomalous regions. Auto-encoder (AE) [2, 63], variational auto-encoder (VAE) [9, 32], or generative adversarial network (GAN) [1, 47] are used to restore normal pixels. However, *pixel reconstruction* models may also succeed in restoring unseen anomalous regions if they resemble normal regions in pixel values or the anomalies are barely noticeable [10]. Therefore, *feature reconstruction* is proposed to construct features of pre-trained encoders instead of raw pixels [10, 58, 60]. To prevent the whole network from converging to a trivial solution, the parameters of the encoders are frozen during training. In *feature distillation* [46, 55], the student network is trained from scratch to mimic the output features of the pre-trained teacher network with the same input of normal images, also based on the similar hypothesis that the student trained on normal samples only succeed in mimicking features of normal regions.

Pseudo-anomaly methods generate handcrafted defects on normal images to imitate anomalies, converting UAD to supervised classification [29] or segmentation tasks [62]. Specifically, CutPaste [29] simulates anomalous regions by randomly pasting cropped patches of normal images. DRAEM [62] constructs abnormal regions using Perlin noise as the mask and another image as the additive anomaly. DeTSeg [67] employs a similar anomaly generation strategy and combines it with feature reconstruction. SimpleNet [34] introduces anomaly by injecting Gaussian noise in the pre-trained feature space. These methods deeply rely on how well the pseudo anomalies match the real anomalies, which makes it hard to generalize to different datasets.

Feature statistics methods [8, 27, 45, 49] memorize all normal features (or their modeled distribution) extracted by networks pre-trained on large-scale datasets and match them with test samples during inference. Since these methods require memorizing, processing, and matching nearly all features from training samples, they are computationally expensive in both training and inference, especially when the training set is large.

Scope of Application. In this work, we focus on **sensory AD** that detects regional or structural anomalies (common in practical applications such as industrial inspection, medical disease screening, etc.), which is distinguished from **semantic AD**. In sensory AD, normal and anomalous samples are the same objects except for anomaly, e.g. good cable vs. spoiled cable. In semantic AD, the class of normal samples and anomalous samples are semantically different, e.g. animals vs. vehicles. Semantic AD methods usually utilize and compare the global representation of images, which generally do not suffer from the issues of multi-class setting discussed in this paper..

B. Full Implementation Details

ViT-Base/14 (patch size=14) pre-trained by DINOv2 with registers (DINOv2-R) [7] is utilized as the encoder by default. The discard rate of Dropout in Noisy Bottleneck is 0.2 by default, which is increased to 0.4 for the diverse Real-IAD. Loose constraint with 2 groups and $\mathcal{L}_{global-hm}$ loss are used by default. The input image is first resized to 448^2 and then center-cropped to 392^2, so that the feature map (28^2) is large enough for localization. As previously discussed, the middle 8 layers of 12-layer ViT-Base are used for reconstruction and feeding the bottleneck. ViT-Small also has 12 layers, which is the same. ViT-Large contains 24 layers; therefore, we use the [4,6,8,...18] layers (index start from 0). The decoder always contains 8 layer.

StableAdamW optimizer [56] with AMSGrad [41] is utilized with lr (learning rate)=2e-3, β=(0.9,0.999), wd (weight decay)=1e-4 and eps=1e-10. The network is trained for 10,000 iterations for MVTec-AD and VisA and 50,000 iterations for Real-IAD under MUAD setting. The network is trained for 5,000 iterations on each class under the class-separated UAD setting. The lr warms up from 0 to 2e-3 in the first 100 iterations and cosine anneals to 2e-4 throughout the training. The discarding rate in Equation 5 linearly rises from 0% to 90% in the first 1,000 iterations as warm-up (500 iters for class-separated setting). The anomaly map is obtained by upsampling the point-wise cosine distance between encoder and decoder feature maps (averaging if more than one pair or group). The mean of the top 1% pixels in an anomaly map is used as the image anomaly score. All experiments are conducted with random seed=1 with cuda deterministic for invariable weight initialization and batch order. Codes are implemented with Python 3.8 and PyTorch 1.12.0 cuda 11.3, and run on NVIDIA GeForce RTX3090 GPUs (24GB).

Most results of compared MUAD SoTAs are directly

Table A1. Comparison between pre-trained ViT foundations, conducted on MVTec-AD (%). All models are ViT-Base. The patch size of DINOv2 and DINOv2-R is 14^2; others are 16^2. R448^2-C392^2 represents first resizing images to 448×448, then center cropping to 392×392.

Pre-Train Backbone	Type	Image Size	Image-level			Pixel-level			
			AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
DeiT[50]	Supervised	R512^2-C448^2	98.19	99.24	97.64	97.93	68.98	67.91	91.45
MAE[19]	MIM	R512^2-C448^2	96.27	98.33	95.44	96.96	62.89	63.32	89.85
D-iGPT[44]	MIM	R512^2-C448^2	98.75	99.24	97.70	98.30	65.77	66.16	92.34
MOCOv3[6]	CL	R512^2-C448^2	98.47	99.42	97.36	98.52	70.99	69.41	92.83
DINO[4]	CL	R512^2-C448^2	98.97	99.58	98.14	98.52	70.89	69.02	93.48
iBOT[69]	CL+MIM	R512^2-C448^2	99.22	99.67	98.57	98.60	70.78	69.92	93.33
DINOv2[39]	CL+MIM	R448^2-C392^2	99.55	99.81	99.13	98.26	68.35	68.79	94.83
DINOv2-R[7]	CL+MIM	R448^2-C392^2	99.60	99.78	99.04	98.35	69.29	69.17	94.79
DeiT[50]	Supervised	R256^2-C224^2	97.65	99.05	97.40	97.80	62.58	63.39	89.98
MAE[19]	MIM	R256^2-C224^2	97.25	98.84	96.94	97.78	63.00	64.01	90.95
BEiTv2[40]	MIM	R256^2-C224^2	97.70	99.11	97.39	97.61	59.79	62.53	90.10
D-iGPT[44]	MIM	R256^2-C224^2	99.21	99.66	98.47	98.08	60.05	63.05	91.78
MOCOv3[6]	CL	R256^2-C224^2	98.74	99.56	98.33	98.05	63.36	64.38	91.13
DINO[4]	CL	R256^2-C224^2	99.20	99.72	98.77	98.16	64.16	65.07	92.02
iBOT[69]	CL+MIM	R256^2-C224^2	99.31	99.74	98.77	98.25	64.01	65.37	91.68
DINOv2[39]	CL+MIM	R256^2-C224^2	99.26	99.70	98.60	97.95	62.27	64.39	92.80
DINOv2-R[7]	CL+MIM	R256^2-C224^2	99.34	99.73	99.03	98.09	63.04	64.48	92.59

Table A2. Ablations of input size, conducted on MVTec-AD (%). R448^2-C392^2 represents first resizing images to 448×448, then center cropping to 392×392.

Image Size	MACs	Image-level			Pixel-level			
		AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
R512^2-C448^2	136.4G	**99.67**	99.81	99.12	98.33	69.24	**69.47**	94.76
R448^2	136.4G	99.59	99.77	99.19	**98.57**	68.09	68.58	**95.60**
R448^2-C392^2†	104.7G	99.60	99.78	99.04	98.35	**69.29**	**69.17**	94.79
R392^2	104.7G	99.48	99.74	99.04	98.47	67.02	67.86	95.34
R384^2-C336^2	77.1G	99.61	99.78	99.22	98.27	67.22	67.77	94.24
R336^2	77.1G	99.63	**99.84**	99.23	98.48	65.46	66.60	95.10
R320^2-C280^2	53.7G	99.62	99.81	99.07	98.21	65.21	66.34	93.57
R280^2	53.7G	99.46	99.75	**99.27**	98.40	63.28	64.79	94.47

drawn from a benchmark paper ADer [66]. We express great thanks for their wonderful work.

C. Additional Ablation and Experiment

Pre-Trained Foundations. The representation quality of the frozen backbone Transformer is of great significance to unsupervised anomaly detection. We conduct extensive experiments to probe the impact of different pre-training methods, including supervised learning and self-supervised learning. DeiT [50] is trained on ImageNet[11] in a supervised manner by distilling CNNs. MAE [19], BEiTv2 [40], and D-iGPT [44] are based on masked image modeling (MIM). Given input images with masked patches, MAE [19] is optimized to restore raw pixels; BEiTv2 [40] is trained to predict the token index of VQ-GAN and CLIP; D-iGPT [44] is trained to predict the features of CLIP model. MOCOv3 [6] is based on contrastive learning (CL), pulling the representations of the similar images and pushing those of different images. DINO [4] is based on positive-pair contrastive learning, which is also referred to as self-distillation. It trains the network to produce similar feature representations given two views (augmentations) of the same image. iBot [69] and DINOv2 [39] combine MIM and CL strategies, marking the SoTA of self-supervised foundation models. DINOv2-R [7] is a variation

Table A3. Scaling of ViT architectures on VisA and Real-IAD (%). †: default.

Dateset	Arch.	Image-level			Pixel-level			
		AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
VisA [70]	ViT-Small	97.94	98.09	95.33	98.57	51.19	55.10	93.71
	ViT-Base†	98.73	98.87	**96.18**	98.74	53.23	55.69	94.50
	ViT-Large	**98.85**	**99.09**	96.12	**99.10**	**55.68**	**57.33**	**94.76**
Real-IAD [54]	ViT-Small	89.10	86.91	79.87	98.69	41.88	46.74	94.08
	ViT-Base†	89.33	86.77	80.17	98.84	42.79	47.10	93.86
	ViT-Large	**90.07**	**87.57**	**80.90**	**99.02**	**44.29**	**48.36**	**94.37**

Table A4. Ablations of Dinomaly elements on VisA (%). NB: Noisy Bottleneck. LA: Linear Attention. LC: Loosen Constraint (2 groups). LL: Loosen Loss.

NB	LA	LC	LL	Image-level			Pixel-level			
				AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
				95.81	96.35	92.06	97.97	47.88	52.55	93.43
✓				97.38	97.74	94.07	97.84	50.42	54.57	93.71
	✓			95.74	96.23	91.87	98.01	47.89	52.58	93.34
		✓		96.39	97.01	92.54	97.37	46.80	51.66	92.75
			✓	96.93	97.26	93.32	98.37	49.52	53.59	94.11
✓	✓			97.52	97.75	94.33	98.06	51.49	55.09	93.75
✓		✓		98.06	98.37	95.18	98.21	51.43	54.89	93.94
✓		✓	✓	98.57	98.77	95.75	98.57	52.29	55.38	94.28
✓	✓	✓		98.22	98.43	95.27	98.51	53.11	55.48	94.24
✓	✓	✓	✓	**98.73**	**98.87**	**96.18**	**98.74**	**53.23**	**55.69**	**94.50**

of DINOv2 that employs 4 extra register tokens.

It is noted that most models are pre-trained with the image resolution of 224×224, except that DINOv2 [39] and DINOv2-R [7] have extra a high-resolution training phase with 518×518. Directly using the pre-trained weights on a different resolution for UAD without fine-tuning like other supervised tasks can cause generalization problems. Therefore, by default, we still keep the feature size of all compared models to 28×28, i.e., the input size is 392×392 for ViT-Base/14 and 448×448 for ViT-Base/16. Additionally, we train Dinomaly with the low-resolution input size of 224×224.

The results are presented in Table A1. Within Dinomaly, nearly all foundation models can produce SoTA-level results with image-level AUROC higher than 98%. Generally speaking, CL+MIM combined models outperform MIM and CL models. In addition, most foundations do not benefit from a higher resolution for image-level performance but suffer from it, indicating the lack of generalization on a input size different from pre-training; while as expected, DINOv2 and DINOv2-R pre-trained on larger inputs can better benefit from higher resolution in Dinomaly. Because some methods, i.e., D-iGPT, DINO, and iBOT, produce similar results to DINOv2 in 224×224, we expect that they also have the potential to be as powerful in Dinomaly if they are pre-trained in high-resolution. Employing MAE produces the worst results. MAE was also tested as the backbone of ViTAD[65], resulting in undesirable performances (I-AUROC=95.3), which was attributed to the weak semantic expression caused by the pretraining strategy. It is also noted that MAE is bad in other unsupervised tasks such as ImageNet kNN; therefore, MAE is considered to be less effective in tasks without finetuning.

Input Size. The patch size of ViTs (usually 14×14 or 16×16) is much larger than the stem layer's down-sampling rate of CNNs (usually 4×4), resulting in smaller feature map size. For dense prediction tasks like semantic segmentation, ViTs usually employ a large input image size [39]. This practice holds in anomaly localization as well. In Table A2, we present the results of Dinomaly with different input resolutions. Following PatchCore [45], by default, we adopt center-crop preprocessing to reduce the influence of background, which can also cause unreachable anomalies at the edge of images. Experimental results demonstrate our robustness to input size. While small image size is enough for image-level anomaly detection, larger inputs are beneficial to anomaly localization. All experiments evaluate localization performance in a unified size of 256×256 for fairness.

Scalability on VisA and Real-IAD. We demonstrate the performance of different ViT sizes on VisA and Real-IAD in Table A3.

Table A5. Ablations of Dropout rates in Noisy Bottleneck, conducted on MVTec-AD (%). †: default.

Dropout rate	Image-level			Pixel-level			
	AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
0 (noiseless)	99.19	99.55	98.51	97.55	63.11	64.39	93.33
0.1	99.54	99.75	98.90	**98.35**	**69.46**	**69.19**	94.53
0.2 †	99.60	99.78	99.04	**98.35**	69.29	69.17	**94.79**
0.3	**99.65**	**99.83**	99.16	98.34	68.46	68.81	94.63
0.4	99.64	99.80	**99.23**	98.22	67.95	68.33	94.57
0.5	99.56	99.81	99.14	98.15	67.43	67.82	94.64

Table A6. Ablations of reconstruction constraint, conduected on MVTec-AD (%). †: default.

Constraints	Image-level			Pixel-level			
	AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
layer-to-layer (dense, every 1)	99.39	99.68	98.73	98.12	68.55	68.63	94.28
layer-to-layer (sparse, every 2)	99.52	99.73	98.95	98.16	68.89	68.57	94.40
layer-to-layer (sparse, every 4)	99.54	99.77	99.05	98.04	66.69	67.17	94.07
layer-to-cat-layer (every 2)	99.48	99.71	99.26	97.83	62.29	62.91	93.16
group-to-group (1 group)	**99.64**	**99.80**	**99.36**	98.18	64.79	65.40	93.96
group-to-group (2 groups)†	99.60	99.78	99.04	**98.35**	**69.29**	**69.17**	**94.79**

Table A7. Comparison between Convolutional block, Softmax Attention, and Linear Attention as the spatial mixer of decoder, conducted on MVTec-AD (%).

Spatial Mixer	Image-level			Pixel-level			
	AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
ConvBlock 3×3	99.45	99.63	98.64	98.05	65.35	68.07	94.17
ConvBlock 5×5	99.41	99.62	98.86	97.99	66.64	67.47	94.24
ConvBlock 7×7	99.42	99.65	98.86	98.01	67.57	67.94	94.45
Softmax Attention	99.52	99.73	98.92	98.20	68.25	68.34	94.17
Softmax Attention w/ Neighbour-Mask $n = 1$	99.51	99.71	98.90	98.17	67.86	67.92	94.27
Softmax Attention w/ Neighbour-Mask $n = 3$	99.56	99.76	99.05	98.28	69.26	68.17	94.50
Linear Attention	**99.60**	99.78	99.04	98.35	69.29	69.17	**94.79**
Linear Attention w/ Neighbour-Mask $n = 1$	**99.60**	99.78	99.04	98.32	68.77	68.72	94.75
Linear Attention w/ Neighbour-Mask $n = 3$	**99.60**	**99.80**	**99.14**	**98.38**	**69.65**	**69.38**	94.70

Ablations on VisA. Similar to Table 3 that conduct ablation experiments on MVTec-AD, we additionally run them on VisA for further validations. As shown in Table A4, proposed components of Dinomaly contribute to the AD performances on VisA as on MVTec-AD.

Noisy Rates. We conduct ablations on the discarding rate of the Dropouts in MLP bottleneck, as shown in Table A5. Experimental results demonstrate that Dinomaly is robust to different levels of dropout rate.

Reconstruction Constraint. We quantitatively examine different reconstruction schemes presented in Figure 4. As shown in Table A6, group-to-group LC outperforms layer-to-layer supervision. On image-level metrics, 1-group LC with all layers added performs similarly to its 2-group counterpart that separates low-level and high-level layers; however, 1-group LC mixes low-level and high-level features which is harmful for anomaly localization. More ablations on scalability, input size, pre-trained foundations, etc., are presented in Appendix C.

Attention *vs.* Convolution. Previous works and this paper have proposed to leverage attentions instead of convolutions in UAD. Here, we conduct experiments substituting the attention in the decoder of Dinomaly by convolutions as the spatial mixers. Following MetaFormer [61], we employ Inverted Bottleneck block that consists of 1×1 conv, GELU activation, $N \times N$ deep-wise conv, and 1×1 conv, sequentially. The results are shown in Table A7, where Attentions outperform Convolutions, especially for pixel-level

Table A8. Dropout *vs.* feature jitter, conducted on MVTec-AD (%).

Noise type	Image-level			Pixel-level			
	AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
No Noise	99.19	99.55	98.51	97.55	63.11	64.39	93.33
Patch Masking p=0.1	99.27	99.60	98.80	97.92	67.15	66.90	94.18
Patch Masking p=0.2	99.17	99.56	98.59	97.75	66.55	66.32	94.11
Patch Masking p=0.3	99.11	99.54	98.37	97.53	65.48	65.96	93.84
Patch Masking p=0.4	99.20	99.59	98.53	97.71	65.58	66.36	94.15
Feature Jitter *scale*=1	99.23	99.54	98.48	97.58	63.22	64.31	93.55
Feature Jitter *scale*=5	99.24	99.57	98.55	97.84	65.28	65.81	93.75
Feature Jitter *scale*=10	99.46	99.73	99.12	98.19	67.59	67.80	94.19
Feature Jitter *scale*=20	99.59	99.79	99.04	98.23	67.93	68.21	94.40
Dropout p=0.1	99.54	99.75	98.90	**98.35**	**69.46**	**69.19**	94.53
Dropout p=0.2	99.60	99.78	99.04	**98.35**	69.29	69.17	**94.79**
Dropout p=0.3	**99.65**	**99.83**	99.16	98.34	68.46	68.81	94.63
Dropout p=0.4	99.64	99.80	**99.23**	98.22	67.95	68.33	94.57

Table A9. Integrating the essense of Noisy Bottleneck (NB) and Loose Loss (LL) on RD4AD, conducted on MVTec-AD (%). †: Reproduction in our framework; ReLU in ResNet decoder is replaced by GELU, StableAdamW optimizer is used.

Method	NB	LL	Image-level			Pixel-level			
			AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
RD4AD†			97.8	99.1	97.2	96.4	58.0	59.3	91.9
RD4AD	✓		98.4	99.4	97.9	97.2	58.6	60.4	92.9
RD4AD		✓	98.2	99.2	97.5	96.8	60.0	61.1	92.7
RD4AD	✓	✓	98.5	99.4	97.8	97.2	59.6	61.2	93.0

Table A10. Scaling properties of a previous ViT-based method, ViTAD[65] on MVTec-AD. †: their original setting.

Method	Pre-Train Backbone	Input Size	Image-level			Pixel-level			
			AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
ViTAD†	DINO	256^2	98.3	99.4	97.3	97.7	55.3	58.7	91.4
ViTAD	MAE	256^2	95.3	97.7	95.2	97.4	53.0	56.2	90.6
ViTAD	DINOv2	256^2	98.7	99.4	98.1	97.6	55.3	59.1	92.7
ViTAD	DINOv2-R	256^2	98.5	99.3	97.8	97.4	54.5	59.2	92.8
ViTAD†	DINO	256^2	98.3	99.4	97.3	97.7	55.3	58.7	91.4
ViTAD	DINO	320^2	98.3	99.2	97.1	97.6	61.3	63.3	92.4
ViTAD	DINO	384^2	97.8	98.9	96.3	97.5	62.5	63.7	92.4

anomaly localization. In addition, utilizing convolutions in the decoder can still yield SoTA results, demonstrating the universality of the proposed Dinomaly.

Neighbour-Masking. Prior method [60] proposed to mask the keys and values in an $n \times n$ square centered at each query, in order to alleviate identity mapping in Attention. This mechanism can also be applied to Linear Attention as well. As shown in Table A7, neighbor-masking can further improve Dinomaly with both Softmax Attention and Linear Attention moderately.

Noise Bottleneck. UniAD [60] proposed to perturb the encoder features by Feature Jitter, i.e. adding Gaussian noise with *scale* to control the noise magnitude. It is also easy to borrow the masking strategy of MAE [19] to randomly mask patch tokens before the decoder. We evaluate the effectiveness of feature jitter and patch-masking in Dinomaly by placing it at the beginning of Noisy Bottleneck. As shown in Table A8, both Dropout and Feature Jitter can

Table A11. Matching previous methods in computation consumption. Dinomaly can be easily scaled by model size and input size.

Method	Params	MACs	MVTec-AD [3]			VisA [70]		
			I-AUROC	P-AUROC	P-AUPRO	I-AUROC	P-AUROC	P-AUPRO
DiAD [18]	1331M	451.5G	97.2	96.8	90.7	86.8	96.0	75.2
ReContrast [14]	154.2M	67.4G	98.3	97.1	93.2	95.5	98.5	91.9
RD4AD [10]	126.7M	32.1G	94.6	96.1	91.1	92.4	98.1	91.8
ViTAD [65]	39.0M	9.7G	98.3	97.7	91.4	90.5	98.2	85.1
Dinomaly-Base-392^2	148M	104.7G	99.6	98.4	94.8	98.7	98.7	94.5
Dinomaly-Base-280^2	148M	53.7G	99.6	98.2	93.6	97.8	98.7	92.4
Dinomaly-Small-392^2	37.4M	26.2G	99.3	98.1	94.4	97.9	98.6	93.7
Dinomaly-Small-280^2	37.4M	14.5G	99.3	98.0	93.4	96.5	98.5	90.9

Table A12. Results of 5 random seeds on MVTec-AD (%).

Random Seed	Image-level			Pixel-level			
	AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
seed=1	99.60	99.78	99.04	98.35	69.29	69.17	94.79
seed=2	99.63	99.79	99.12	98.33	68.73	68.91	94.63
seed=3	99.63	99.79	99.16	98.31	68.70	68.93	94.60
seed=4	99.56	99.74	99.02	98.33	69.04	69.09	94.70
seed=5	99.59	99.77	99.02	98.32	68.64	68.47	94.51
mean±std	99.60±0.03	99.77±0.02	99.07±0.06	98.33±0.01	68.88±0.25	68.91±0.24	94.65±0.09

be a good noise injector in Noisy Bottleneck. Meanwhile, Dropout is more robust to the noisy scale hyperparameter, and more elegant without introducing new modules.

Adaptation on CNN Method. Some proposed elements (Linear Attention and Loose Constraint) are closely bounded to modern ViTs. Loose Loss (hard-mining) can be directly applied to previous CNN-based methods, e.g., RD4AD [10]. Noisy Bottleneck can be adapted to RD4AD with minor modifications (apply dropout before MFF layer). We apply these modules to RD4AD to validate the effectiveness of our contributions. The results are shown in Table A9, where these two elements boost the performance of RD4AD to a whole new level that can be compared with prior MUAD SoTAs.

Scaling of Compared Method. As previously discussed in the Experiment section, compared method cannot fully utilize the scaling of pre-trained method, model size, and input size. For example, RD4AD [10] found WideResNet50 better than WideResNet101 as the encoder backbone. ViTAD [65] found ViT-Small better than ViT-Base. Here, we also present the experiments on pre-training method and input size of ViTAD, as shown in Table A10. It is also noted that the paradigm of ViTAD is very similar to RD4AD (replacing CNN by ViT) as well as the starting point of Dinomaly (the first row in the ablation Table 3).

Computation Comparison. The computation costs of Dinomaly variants were previously presented in Table 4 and Table A2. Here, we compare the computation consumption of Dinomaly and prior works. As shown in Table A11, Dinomaly can be easily scaled by model size and input size to match different application scenarios.

Random Seeds. Due to limited computation resources, experiments in this paper are conducted for one run with random seed=1. Here, we conduct 5 runs with 5 random seeds on MVTec-AD. As shown in Table A12, Dinomaly is robust to randomness.

D. Additional Dataset

To demonstrate the generalization of our method, we conduct experiments on three more popular anomaly detection datasets under MUAD setting, including MPDD and BTAD and Uni-Medical. The MPDD [24] (Metal Parts Defect Detection Dataset) is a dataset aimed at benchmarking visual defect detection methods in industrial metal parts manufacturing. It consists of more than 1346 images across 6 categories with pixel-precise defect annotation masks. The BTAD [38] (beanTech Anomaly Detection) dataset is a real-world industrial anomaly dataset. The dataset contains a total of 2830 real-world images of 3 industrial products showcasing body and surface defects. It is noted that the training set of BTAD is noisy because it contains anomalous samples [25]. Uni-Medical [66] is a medical UAD dataset consisting of 2D image slices from 3D CT volumes. It con-

Table A13. Performance on MPDD and BTAD under **multi-class** UAD setting (%). †: method designed for MUAD.

Dateset	Method	Image-level			Pixel-level			
		AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
MPDD [24]	RD4AD [10]	90.3	92.8	90.5	98.3	39.6	40.6	95.2
	SimpleNet [34]	90.6	94.1	89.7	97.1	33.6	35.7	90.0
	DeSTSeg [67]	92.6	91.8	92.8	90.8	30.6	32.9	78.3
	UniAD [60]†	80.1	83.2	85.1	95.4	19.0	25.6	83.8
	DiAD [18]†	85.8	89.2	86.5	91.4	15.3	19.2	66.1
	ViTAD [65]†	87.4	90.8	87.0	97.8	44.1	46.4	95.3
	MambaAD [17]†	89.2	93.1	90.3	97.7	33.5	38.6	92.8
	Dinomaly (Ours)	**97.2**	**98.4**	**96.0**	**99.1**	**59.5**	**59.4**	**96.6**
BTAD [38]	RD4AD [10]	94.1	96.8	93.8	**98.0**	57.1	58.0	**79.9**
	SimpleNet [34]	94.0	97.9	93.9	96.2	41.0	43.7	69.6
	DeSTSeg [67]	93.5	96.7	93.8	94.8	39.1	38.5	72.9
	UniAD [60]†	94.5	**98.4**	94.9	97.4	52.4	55.5	78.9
	DiAD [18]†	90.2	88.3	92.6	91.9	20.5	27.0	70.3
	ViTAD [65]†	94.0	97.0	93.7	97.6	58.3	56.5	72.8
	MambaAD [17]†	92.9	96.2	93.0	97.6	51.2	55.1	77.3
	Dinomaly (Ours)	**95.4**	**98.4**	**95.6**	97.8	**70.1**	**68.0**	76.5
Uni-Medical [66]	RD4AD [10]	76.1	75.3	78.2	96.5	38.3	39.8	86.8
	SimpleNet [34]	77.5	77.7	76.7	94.3	34.4	36.0	77.0
	DeSTSeg [67]	78.5	77.0	78.2	65.7	41.7	34.0	35.3
	UniAD [60]†	79.0	76.1	77.1	96.6	39.3	41.1	86.0
	DiAD [18]†	78.8	77.2	77.7	95.8	34.2	35.5	84.3
	ViTAD [65]†	81.8	80.7	80.0	**97.1**	48.3	48.2	86.7
	MambaAD [17]†	83.9	80.8	**81.9**	96.8	45.8	47.5	**88.2**
	Dinomaly (Ours)	**84.9**	**84.1**	81.0	96.8	**51.7**	**52.1**	85.5

tains 13339 training images and 7013 test images across three objects, i.e., brain CT, liver CT, and retinal OCT. This dataset is not entirely suitable for evaluating 2D anomaly detection methods, as identifying lesions in medical images requires 3D contextual information. The training hyperparameters are the same to MVTec-AD, except the dropout rate for Uni-Medical is increased to 0.4. The performance of Dinomaly and previous SoTAs is presented in Table A13, where our method demonstrates superior results.

E. Results Per-Category

For future research, we report the per-class results of MVTec-AD [3], VisA [70], and Real-IAD [54]. The performance of compared methods is drawn from MambaAD [17]. Thanks for their exhaustive reproducing. The results of image-level anomaly detection and pixel-level anomaly localization on MVTec-AD are presented in Table A14 and Table A15, respectively. The results of image-level anomaly detection and pixel-level anomaly localization on VisA are presented in Table A16 and Table A17, respectively. The results of image-level anomaly detection and pixel-level anomaly localization on Real-IAD are presented in Table A18 and Table A19, respectively.

F. Qualitative Visualization

We visualize the output anomaly maps of Dinomaly on MVTec-AD, VisA, and Real-IAD, as shown in Figure A1, Figure A2, and Figure A3. It is noted that all visualized samples are randomly chosen without artificial selection.

G. Limitation

Vision Transformers are known for their high computation cost, which can be a barrier to low-computation scenarios that require inference speed. Future research can be conducted on the efficiency of Transformer-based methods, such as distillation, pruning, and hardware-friendly attention mechanism (such as FlashAttention).

As discussed in section A, Dinomaly is used for sensory AD that aims to detect regional anomalies in normal backgrounds. It is not suitable for semantic AD. Previous works have shown that methods designed for sensory AD usually fail to be competitive under semantic AD tasks [10, 60]. Conversely, methods designed for semantic AD do not perform well on sensory AD tasks [42, 43]. Future work can be conducted to unify these two tasks, but according to the "no free lunch" theorem, we believe that methods designed for specific anomaly assumption are likely to be more convincing.

Other special UAD settings, such as zero-shot UAD (vision-language model based) [23], few-shot UAD [22], UAD under noisy training set [25], are not included in this work.

Table A14. Per-class performance on **MVTec-AD** dataset for multi-class anomaly detection with AUROC/AP/F_1-max metrics.

	Method →	RD4AD [10]	UniAD [60]	SimpleNet [34]	DeSTSeg [67]	DiAD [18]	MambaAD [17]	Dinomaly
	Category ↓	CVPR'22	NeurIPS'22	CVPR'23	CVPR'23	AAAI'24	Arxiv'24	Ours
Objects	Bottle	99.6/99.9/98.4	99.7/**100./100.**	**100./100./100.**	98.7/99.6/96.8	99.7/96.5/91.8	**100./100./100.**	**100./100./100.**
	Cable	84.1/89.5/82.5	95.2/95.9/88.0	97.5/98.5/94.7	89.5/94.6/85.9	94.8/98.8/95.2	98.8/99.2/95.7	**100./100./100.**
	Capsule	94.1/96.9/96.9	86.9/97.8/94.4	90.7/97.9/93.5	82.8/95.9/92.6	89.0/97.5/95.5	94.4/98.7/94.9	**97.9/99.5/97.7**
	Hazelnut	60.8/69.8/86.4	99.8/**100.**/99.3	99.9/99.9/99.3	98.8/99.2/98.6	99.5/99.7/97.3	**100./100./100.**	**100./100./100.**
	Metal Nut	**100./100.**/99.5	99.2/99.9/99.5	96.9/99.3/96.1	92.9/98.4/92.2	99.1/96.0/91.6	99.9/**100.**/99.5	**100./100./100.**
	Pill	97.5/99.6/96.8	93.7/98.7/95.7	88.2/97.7/92.5	77.1/94.4/91.7	95.7/98.5/94.5	97.0/99.5/96.2	**99.1/99.9/98.3**
	Screw	97.7/99.3/95.8	87.5/96.5/89.0	76.7/90.6/87.7	69.9/88.4/85.4	90.7/**99.7/97.9**	94.7/97.9/94.0	**98.4**/99.5/96.1
	Toothbrush	97.2/99.0/94.7	94.2/97.4/95.2	89.7/95.7/92.3	71.7/89.3/84.5	99.7/99.9/99.2	98.3/99.3/98.4	**100./100./100.**
	Transistor	94.2/95.2/90.0	99.8/98.0/93.8	99.2/98.7/97.6	78.2/79.5/68.8	99.8/99.6/97.4	**100./100./100.**	99.0/98.0/96.4
	Zipper	99.5/99.9/99.2	95.8/99.5/97.1	99.0/99.7/98.3	88.4/96.3/93.1	95.1/99.1/94.4	99.3/99.8/97.5	**100./100./100.**
Textures	Carpet	98.5/99.6/97.2	**99.8**/99.9/**99.4**	95.7/98.7/93.2	95.9/98.8/94.9	99.4/99.9/98.3	**99.8**/99.9/**99.4**	**99.8/100.**/98.9
	Grid	98.0/99.4/96.5	98.2/99.5/97.3	97.6/99.2/96.4	97.9/99.2/96.6	98.5/99.8/97.7	**100./100./100.**	99.9/**100.**/99.1
	Leather	**100./100./100.**	**100./100./100.**	**100./100./100.**	99.2/99.8/98.9	99.8/99.7/97.6	**100./100./100.**	**100./100./100.**
	Tile	98.3/99.3/96.4	99.3/99.8/98.2	99.3/99.8/98.8	97.0/98.9/95.3	96.8/99.9/98.4	98.2/99.3/95.4	**100./100./100.**
	Wood	99.2/99.8/98.3	98.6/99.6/96.6	98.4/99.5/96.7	99.9/**100.**/99.2	99.7/**100./100.**	98.8/99.6/96.6	99.8/99.9/99.2
	Mean	94.6/96.5/95.2	96.5/98.8/96.2	95.3/98.4/95.8	89.2/95.5/91.6	97.2/99.0/96.5	98.6/99.6/97.8	**99.6/99.8/99.0**

Table A15. Per-class performance on **MVTec-AD** dataset for multi-class anomaly localization with AUROC/AP/F_1-max/AUPRO metrics.

	Method →	RD4AD [10]	UniAD [60]	SimpleNet [34]	DeSTSeg [67]	DiAD [18]	MambaAD [17]	Dinomaly
	Category ↓	CVPR'22	NeurIPS'22	CVPR'23	CVPR'23	AAAI'24	Arxiv'24	Ours
Objects	Bottle	97.8/68.2/67.6/94.0	98.1/66.0/69.2/93.1	97.2/53.8/62.4/89.0	93.3/61.7/56.0/67.5	98.4/52.2/54.8/86.6	98.8/79.7/76.7/95.2	**99.2/88.6/84.2/96.6**
	Cable	85.1/26.3/33.6/75.1	97.3/39.9/45.2/86.1	96.7/42.4/51.2/85.4	89.3/37.5/40.5/49.4	96.8/50.1/57.8/80.5	95.8/42.2/48.1/90.3	**98.6/72.0/74.3/94.2**
	Capsule	**98.8**/43.4/50.0/94.8	98.5/42.7/46.5/92.1	98.5/35.4/44.3/84.5	95.8/47.9/48.9/62.1	97.1/42.0/45.3/87.2	98.4/43.9/47.7/92.6	98.7/**61.4/60.3/97.2**
	Hazelnut	97.9/36.2/51.6/92.7	98.1/55.2/56.8/94.1	98.4/44.6/51.4/87.4	98.2/65.8/61.6/84.5	98.3/79.2/**80.4**/91.5	99.0/63.6/64.4/95.7	**99.4/82.2**/76.4/**97.0**
	Metal Nut	94.8/55.5/66.4/91.9	62.7/14.6/29.2/81.8	**98.0/83.1**/79.4/85.2	84.2/42.0/22.8/53.0	97.3/30.0/38.3/90.6	96.7/74.5/79.1/93.7	96.9/78.6/**86.7/94.9**
	Pill	97.5/63.4/65.2/95.8	95.0/44.0/53.9/95.3	96.5/72.4/67.7/81.9	96.2/61.7/41.8/27.9	95.7/46.0/51.4/89.0	97.4/64.0/66.5/95.7	**97.8/76.4/71.6/97.3**
	Screw	99.4/40.2/44.6/96.8	98.3/28.7/37.6/95.2	96.5/15.9/23.2/84.0	93.8/19.9/25.3/47.3	97.9/**60.6/59.6**/95.0	99.5/49.8/50.9/97.1	**99.6**/60.2/**59.6/98.3**
	Toothbrush	**99.0**/53.6/58.8/92.0	98.4/34.9/45.7/87.9	98.4/46.9/52.5/87.4	96.2/52.9/58.8/30.9	**99.0**/78.7/72.8/95.0	**99.0**/48.5/59.2/91.7	98.9/**51.5/62.6/95.3**
	Transistor	85.9/42.3/45.2/74.7	**97.9**/59.5/**64.6/93.5**	95.8/58.2/56.0/83.2	73.6/38.4/39.2/43.9	95.1/15.6/31.7/90.0	96.5/69.4/67.1/87.0	93.2/**59.9**/58.5/77.0
	Zipper	98.5/53.9/60.3/94.1	96.8/40.1/49.9/92.6	97.9/53.4/54.6/90.7	97.3/64.7/59.2/66.9	96.2/60.7/60.0/91.6	98.4/60.4/61.7/94.3	**99.2/79.5/75.4/97.2**
Textures	Carpet	99.0/58.5/60.4/95.1	98.5/49.9/51.1/94.4	97.4/38.7/43.2/90.6	93.6/59.9/58.9/89.3	98.6/42.2/46.4/90.6	99.2/60.0/63.3/96.7	**99.3/68.7/71.1/97.6**
	Grid	96.5/23.0/28.4/97.0	63.1/10.7/11.9/92.9	96.8/20.5/27.6/88.6/	97.0/42.1/46.9/86.8	96.6/66.0/64.1/94.0	99.2/47.4/47.7/97.0	**99.4/55.3/57.7/97.2**
	Leather	99.3/38.0/45.1/97.4	98.8/32.9/34.4/96.8	98.7/28.5/32.9/92.7	**99.5/71.5/66.5**/91.1	98.8/56.1/62.3/91.3	99.4/50.3/53.3/98.7	99.4/52.2/55.0/**97.6**
	Tile	95.3/48.5/60.5/85.8	91.8/42.1/50.6/78.4	95.7/60.5/59.9/90.6	93.0/71.0/66.2/87.1	92.4/65.7/64.1/**90.7**	93.8/45.1/54.8/80.0	**98.1/80.1/75.7**/90.5
	Wood	95.3/47.8/51.0/90.0	93.2/37.2/41.5/86.7	91.4/34.8/39.7/76.3	95.9/**77.3/71.3**/83.4	93.3/43.3/43.5/**97.5**	94.4/46.2/48.2/91.2	**97.6**/72.8/68.4/94.0
	Mean	96.1/48.6/53.8/91.1	96.8/43.4/49.5/90.7	96.9/45.9/49.7/86.5	93.1/54.3/50.9/64.8	96.8/52.6/55.5/90.7	97.7/56.3/59.2/93.1	**98.4/69.3/69.2/94.8**

Table A16. Per-class performance on **VisA** dataset for multi-class anomaly detection with AUROC/AP/F1-max metrics.

Method →	RD4AD [10]	UniAD [60]	SimpleNet [34]	DeSTSeg [67]	DiAD [18]	MambaAD	Dinomaly
Category ↓	CVPR'22	NeurIPS'22	CVPR'23	CVPR'23	AAAI'24	Arxiv'24	Ours
pcb1	96.2/95.5/91.9	92.8/92.7/87.8	91.6/91.9/86.0	87.6/83.1/83.7	88.1/88.7/80.7	95.4/93.0/91.6	**99.1/99.1/96.6**
pcb2	97.8/97.8/94.2	87.8/87.7/83.1	92.4/93.3/84.5	86.5/85.8/82.6	91.4/91.4/84.7	94.2/93.7/89.3	**99.3/99.2/97.0**
pcb3	96.4/96.2/91.0	78.6/78.6/76.1	89.1/91.1/82.6	93.7/95.1/87.0	86.2/87.6/77.6	93.7/94.1/86.7	**98.9/98.9/96.1**
pcb4	99.9/99.9/99.0	98.8/98.8/94.3	97.0/97.0/93.5	97.8/97.8/92.7	99.6/99.5/97.0	**99.9/99.9/98.5**	99.8/99.8/98.0
macaroni1	75.9/ 1.5/76.8	79.9/79.8/72.7	85.9/82.5/73.1	76.6/69.0/71.0	85.7/85.2/78.8	91.6/89.8/81.6	**98.0/97.6/94.2**
macaroni2	88.3/84.5/83.8	71.6/71.6/69.9	68.3/54.3/59.7	68.9/62.1/67.7	62.5/57.4/69.6	81.6/78.0/73.8	**95.9/95.7/90.7**
capsules	82.2/90.4/81.3	55.6/55.6/76.9	74.1/82.8/74.6	87.1/93.0/84.2	58.2/69.0/78.5	91.8/95.0/88.8	**98.6/99.0/97.1**
candle	92.3/92.9/86.0	94.1/94.0/86.1	84.1/73.3/76.6	94.9/94.8/89.2	92.8/92.0/87.6	96.8/96.9/90.1	**98.7/98.8/95.1**
cashew	92.0/95.8/90.7	92.8/92.8/91.4	88.0/91.3/84.7	92.0/96.1/88.1	91.5/95.7/89.7	94.5/97.3/91.1	**98.7/99.4/97.0**
chewinggum	94.9/97.5/92.1	96.3/96.2/95.2	96.4/98.2/93.8	95.8/98.3/94.7	99.1/99.5/95.9	97.7/98.9/94.2	**99.8/99.9/99.0**
fryum	95.3/97.9/91.5	83.0/83.0/85.0	88.4/93.0/83.3	92.1/96.1/89.5	89.8/95.0/87.2	95.2/97.7/90.5	**98.8/99.4/96.5**
pipe_fryum	97.9/98.9/96.5	94.7/94.7/93.9	90.8/95.5/88.6	94.1/97.1/91.9	96.2/98.1/93.7	98.7/99.3/97.0	**99.2/99.7/97.0**
Mean	92.4/92.4/89.6	85.5/85.5/84.4	87.2/87.0/81.8	88.9/89.0/85.2	86.8/88.3/85.1	94.3/94.5/89.4	**98.7/98.9/96.2**

Table A17. Per-class performance on **VisA** dataset for multi-class anomaly localization with AUROC/AP/F_1-max/AUPRO metrics.

Method → / Category ↓	RD4AD [10] CVPR'22	UniAD [60] NeurIPS'22	SimpleNet [34] CVPR'23	DeSTSeg [67] CVPR'23	DiAD [18] AAAI'24	MambaAD Arxiv'24	Dinomaly Ours
pcb1	99.4/66.2/62.4/**95.8**	93.3/ 3.9/ 8.3/64.1	99.2/86.1/78.8/83.6	95.8/46.4/49.0/83.2	98.7/49.6/52.8/80.2	**99.8**/77.1/72.4/92.8	99.5/**87.9/80.5**/95.1
pcb2	98.0/22.3/30.0/90.8	93.9/ 4.2/ 9.2/66.9	96.6/ 8.9/18.6/85.7	97.3/14.6/28.2/79.9	95.2/ 7.5/16.7/67.0	**98.9**/13.3/23.4/89.6	98.0/**47.0/49.8/91.3**
pcb3	97.9/26.2/35.2/93.9	97.3/13.8/21.9/70.6	97.2/31.0/36.1/85.1	97.7/28.1/33.4/62.4	96.7/ 8.0/18.8/68.9	**99.1**/18.3/27.4/89.1	98.4/**41.7/45.3/94.6**
pcb4	97.8/31.4/37.0/88.7	94.9/14.7/22.9/72.3	93.9/23.9/32.9/61.1	95.8/**53.0/53.2**/76.9	97.0/17.6/27.2/85.0	98.6/47.0/46.9/87.6	**98.7**/50.5/53.1/**94.4**
macaroni1	99.4/ 2.9/6.9/95.3	97.4/ 3.7/ 9.7/84.0	98.9/ 3.5/8.4/92.0	99.1/ 5.8/13.4/62.4	94.1/10.2/16.7/68.5	99.5/17.5/27.6/95.2	**99.6/33.5/40.6/96.4**
macaroni2	99.7/13.2/21.8/97.4	95.2/ 0.9/ 4.3/76.6	93.2/ 0.6/ 3.9/77.8	98.5/ 6.3/14.4/70.0	93.6/ 0.9/ 2.8/73.1	99.5/ 9.2/16.1/96.2	**99.7/24.7/36.1/98.7**
capsules	99.4/60.4/60.8/93.1	88.7/ 3.0/ 7.4/43.7	97.1/52.9/53.3/73.7	96.9/33.2/ 9.1/76.7	97.3/10.0/21.0/77.9	99.1/61.3/59.8/91.8	**99.6/65.0/66.6/97.4**
candle	99.1/25.3/35.8/94.9	98.5/17.6/27.9/91.6	97.6/ 8.4/16.5/87.6	98.7/39.9/45.8/69.0	97.3/12.8/22.8/89.4	99.0/23.2/32.4/**95.5**	**99.4/43.0/47.9**/95.4
cashew	91.7/44.2/49.7/86.2	98.6/51.7/58.3/87.9	**98.9/68.9/66.0**/84.1	87.9/47.6/52.1/66.3	90.9/53.1/60.9/61.8	94.3/46.8/51.4/87.8	97.1/64.5/62.4/**94.0**
chewinggum	98.7/59.9/61.7/76.9	98.8/54.9/56.1/81.3	97.9/26.8/29.8/78.3	98.8/**86.9/81.0**/68.3	94.7/11.9/25.8/59.5	98.1/57.5/59.9/79.7	**99.1**/65.0/67.7/**88.1**
fryum	97.0/47.6/51.5/93.4	95.9/34.0/40.6/76.2	93.0/39.1/45.4/85.1	88.1/35.2/38.5/47.7	**97.6/58.6/60.1**/81.3	96.9/47.8/51.9/91.6	96.6/51.6/53.4/**93.5**
pipe_fryum	99.1/56.8/58.8/**95.4**	98.9/50.2/57.7/91.5	98.5/65.6/63.4/83.0	98.9/78.8/72.7/45.9	**99.4/72.7/69.9**/89.9	99.1/53.5/58.5/95.1	99.2/64.3/65.1/95.2
Mean	98.1/38.0/42.6/91.8	95.9/21.0/27.0/75.6	96.8/34.7/37.8/81.4	96.1/39.6/43.4/67.4	96.0/26.1/33.0/75.2	98.5/39.4/44.0/91.0	**98.7/53.2/55.7/94.5**

Table A18. Per-class performance on **Real-IAD** dataset for multi-class anomaly detection with AUROC/AP/F_1-max metrics.

Method → / Category ↓	RD4AD [10] CVPR'22	UniAD [60] NeurIPS'22	SimpleNet [34] CVPR'23	DeSTSeg [67] CVPR'23	DiAD [18] AAAI'24	MambaAD Arxiv'24	Dinomaly Ours
audiojack	76.2/63.2/60.8	81.4/76.6/64.9	58.4/44.2/50.9	81.1/72.6/64.5	76.5/54.3/65.7	84.2/76.5/67.4	**86.8/82.4/72.2**
bottle cap	89.5/86.3/81.0	92.5/91.7/81.7	54.1/47.6/60.3	78.1/74.6/68.1	91.6/**94.0/87.9**	**92.8**/92.0/82.1	89.9/86.7/81.2
button battery	73.3/78.9/76.1	75.9/81.6/76.3	52.5/60.5/72.4	86.7/89.2/83.5	80.5/71.3/70.6	79.8/85.3/77.8	86.6/88.9/82.1
end cap	79.8/84.0/77.8	80.9/86.1/78.0	51.6/60.8/72.9	77.9/81.1/77.1	85.1/83.4/**84.8**	78.0/82.8/77.2	**87.0/87.5**/83.4
eraser	90.0/88.7/79.7	**90.3/89.2/80.2**	46.4/39.1/55.8	84.6/82.9/71.8	80.0/80.0/77.3	87.5/86.2/76.1	**90.3**/87.6/78.6
fire hood	78.3/70.1/64.5	80.6/74.8/66.4	58.1/41.9/54.4	81.7/72.4/67.7	83.3/**81.7/80.5**	79.3/72.5/64.8	**83.8**/76.2/69.5
mint	65.8/63.1/64.8	67.0/66.6/64.6	52.4/50.3/63.7	58.4/55.8/63.7	**76.7/76.7/76.0**	70.1/70.8/65.5	73.1/72.0/67.7
mounts	88.6/79.9/74.8	87.6/77.3/77.2	58.7/48.1/52.4	74.7/56.5/63.1	75.3/74.5/**82.5**	86.8/78.0/73.5	**90.4/84.2**/78.0
pcb	79.5/85.8/79.7	81.0/88.2/79.1	54.5/66.0/75.5	82.0/88.7/79.6	86.0/85.1/85.4	89.1/93.7/84.0	**92.0/95.3/87.0**
phone battery	87.5/83.3/77.1	83.6/80.0/71.6	51.6/43.8/58.0	83.3/81.8/72.1	82.3/77.7/75.9	90.2/88.9/80.5	**92.9/91.6/82.5**
plastic nut	80.3/68.0/64.4	80.0/69.2/63.7	59.2/40.3/51.8	83.1/75.4/66.5	71.9/58.2/65.6	87.1/80.7/70.7	**88.3/81.8/74.7**
plastic plug	81.9/74.3/68.8	81.4/75.9/67.6	48.2/38.4/54.6	71.7/63.1/60.0	88.7/**89.2/90.9**	85.7/82.2/72.6	**90.5**/86.4/78.6
porcelain doll	86.3/76.3/71.5	85.1/75.2/69.3	66.3/54.5/52.1	78.7/66.2/64.3	72.6/66.8/65.2	**88.0/82.2/74.1**	85.1/73.3/69.6
regulator	66.9/48.8/47.7	56.9/41.5/44.5	50.5/29.0/43.9	79.2/63.5/56.9	72.1/71.4/**78.2**	69.7/58.7/50.4	**85.2/78.9**/69.8
rolled strip base	97.5/98.7/94.7	98.7/99.3/96.5	59.0/75.7/79.8	96.5/98.2/93.0	68.4/55.9/56.8	98.0/99.0/95.0	**99.2/99.6/97.1**
sim card set	91.6/91.8/84.8	89.7/90.3/83.2	63.1/69.7/70.8	95.5/96.2/**89.2**	72.6/53.7/61.5	94.4/95.1/87.2	**95.8/96.3**/88.8
switch	84.3/87.2/77.9	85.5/88.6/78.4	62.2/66.8/68.6	90.1/92.8/83.1	73.4/49.4/61.2	91.7/94.0/85.4	**97.8/98.1/93.3**
tape	96.0/95.1/87.6	**97.2/96.2/89.4**	49.9/41.1/54.5	94.5/93.4/85.9	73.9/57.8/66.1	96.8/95.9/89.3	96.9/95.0/88.8
terminalblock	89.4/89.7/83.1	87.5/89.1/81.0	59.8/64.7/68.8	83.1/86.2/76.6	62.1/36.4/47.8	96.1/96.8/90.0	**96.7/97.4/91.1**
toothbrush	82.0/83.8/77.2	78.4/80.1/75.6	65.9/70.0/70.1	83.7/85.3/79.0	**91.2/93.7/90.9**	85.1/86.2/80.3	90.4/91.9/83.4
toy	69.4/74.2/75.9	68.4/75.1/74.8	57.8/64.4/73.4	70.3/74.8/75.4	66.2/57.3/59.8	83.0/87.5/79.6	85.6/89.1/81.9
toy brick	63.6/56.1/59.0	**77.0/71.1**/66.2	58.3/49.7/58.2	73.2/68.7/**63.3**	68.4/45.3/55.9	70.5/63.7/61.6	72.3/65.1/63.4
transistor1	91.0/94.0/85.1	93.7/95.9/88.9	62.2/69.2/72.1	90.2/92.1/84.6	73.1/63.1/62.7	94.4/96.0/89.0	**97.4/98.2/93.1**
u block	89.5/85.0/74.2	88.8/84.2/75.5	62.4/48.4/51.8	80.1/73.9/64.3	75.2/68.4/67.9	89.7/**85.7/75.3**	**89.9**/84.0/75.2
usb	84.9/84.3/75.1	78.7/79.4/69.1	57.0/55.3/62.9	87.8/88.0/78.3	58.9/37.4/45.7	**92.0/92.2/84.5**	**92.0**/91.6/83.3
usb adaptor	71.1/61.4/62.2	76.8/71.3/64.9	47.5/38.4/56.5	80.1/**74.9**/67.4	76.9/60.2/67.2	79.4/76.0/66.3	**81.5**/74.5/**69.4**
vcpill	85.1/80.3/72.4	87.1/84.0/74.7	59.0/48.7/56.4	83.8/81.5/69.9	64.1/40.4/56.2	88.3/87.7/77.4	**92.0/91.2/82.0**
wooden beads	81.2/78.9/70.9	78.4/77.2/67.8	55.1/52.0/60.2	82.4/78.5/73.0	62.1/56.4/65.9	82.5/81.7/71.8	**87.3/85.8/77.4**
woodstick	76.9/61.2/58.1	80.8/72.6/63.6	58.2/35.6/45.2	80.4/69.2/60.3	74.1/66.0/62.1	80.4/69.0/63.4	**84.0/73.3/65.6**
zipper	95.3/97.2/91.2	98.2/98.9/95.3	77.2/86.7/77.6	96.9/98.1/93.5	86.0/87.0/84.0	**99.2/99.6/96.9**	99.1/99.5/96.5
Mean	82.4/79.0/73.9	83.0/80.9/74.3	57.2/53.4/61.5	82.3/79.2/73.2	75.6/66.4/69.9	86.3/84.6/77.0	**89.3/86.8/80.2**

Table A19. Per-class performance on **Real-IAD** dataset for multi-class anomaly localization with AUROC/AP/F_1-max/AUPRO metrics.

Method →	RD4AD [10]	UniAD [60]	SimpleNet [34]	DeSTSeg [67]	DiAD [18]	MambaAD [17]	Dinomaly
Category ↓	CVPR'22	NeurIPS'22	CVPR'23	CVPR'23	AAAI'24	Arxiv'24	Ours
audiojack	96.6/12.8/22.1/79.6	97.6/20.0/31.0/83.7	74.4/ 0.9/ 4.8/38.0	95.5/25.4/31.9/52.6	91.6/ 1.0/ 3.9/63.3	97.7/21.6/29.5/83.9	**98.7/48.1/54.5/91.7**
bottle cap	99.5/18.9/29.9/95.7	99.5/19.4/29.6/96.0	85.3/ 2.3/ 5.7/45.1	94.5/25.3/31.1/25.3	94.6/ 4.9/11.4/73.0	**99.7**/30.6/34.6/97.2	**99.7/32.4/36.7/98.1**
button battery	97.6/33.8/37.8/86.5	96.7/28.5/34.4/77.5	75.9/ 3.2/ 6.6/40.5	98.3/**63.9/60.4**/36.9	84.1/ 1.4/ 5.3/66.9	98.1/46.7/49.5/86.2	**99.1**/46.9/56.7/**92.9**
end cap	96.7/12.5/22.5/89.2	95.8/ 8.8/17.4/85.4	63.1/ 0.5/ 2.8/25.7	89.6/14.4/22.7/29.5	81.3/ 2.0/ 6.9/38.2	97.0/12.0/19.6/89.4	**99.1/26.2/32.9/96.0**
eraser	**99.5**/30.8/36.7/96.0	99.3/24.4/30.9/94.1	80.6/ 2.7/ 7.1/42.8	95.8/52.7/53.9/46.7	91.1/ 7.7/15.4/67.5	99.2/30.2/38.3/93.7	**99.5/39.6/43.3/96.4**
fire hood	98.9/27.7/35.2/87.9	98.6/23.4/32.2/85.3	70.5/ 0.3/ 2.2/25.3	97.3/27.1/35.3/34.7	91.8/ 3.2/ 9.2/66.7	98.7/25.1/31.3/86.3	**99.3/38.4/42.7/93.0**
mint	95.0/11.7/23.0/72.3	94.4/ 7.7/18.1/62.3	79.9/ 0.9/ 3.6/43.3	84.1/10.3/22.4/ 9.9	91.1/ 5.7/11.6/64.2	96.5/15.9/27.0/72.6	**96.9/22.0/32.5/77.6**
mounts	99.3/30.6/37.1/94.9	**99.4**/28.0/32.8/95.2	80.5/ 2.2/ 6.8/46.1	94.2/30.0/41.3/43.3	84.3/ 0.4/ 1.1/48.8	99.2/31.4/35.4/93.5	**99.4/39.9/44.3/95.6**
pcb	97.5/15.8/24.3/88.3	97.0/18.5/28.1/81.6	78.0/ 1.4/ 4.3/41.3	97.2/37.1/40.4/48.8	92.0/ 3.7/ 7.4/66.5	99.2/46.3/50.4/93.1	**99.3/55.0/56.3/95.7**
phone battery	77.3/22.6/31.7/94.5	85.5/11.2/21.6/88.5	43.4/ 0.1/ 0.9/11.8	79.5/25.6/33.8/39.5	96.8/ 5.3/11.4/85.4	99.4/36.3/41.3/95.3	**99.7/51.6/54.2/96.8**
phone battery	77.3/22.6/31.7/94.5	85.5/11.2/21.6/88.5	43.4/ 0.1/ 0.9/11.8	79.5/25.6/33.8/39.5	96.8/5.3/11.4/85.4	99.4/36.3/41.3/95.3	**99.7/51.6/54.2/96.8**
plastic nut	98.8/21.1/29.6/91.0	98.4/20.6/27.1/88.9	77.4/ 0.6/ 3.6/41.5	96.5/44.8/45.7/38.4	81.1/ 0.4/ 3.4/38.6	99.4/33.1/37.3/96.1	**99.7/41.0/45.0/97.4**
plastic plug	99.1/20.5/28.4/94.9	98.6/17.4/26.1/90.3	78.6/ 0.7/ 1.9/38.8	91.9/20.1/27.3/21.0	92.9/ 8.7/15.0/66.1	99.0/24.2/31.7/91.5	**99.4/31.7/37.2/96.4**
porcelain doll	99.2/24.8/34.6/95.7	98.7/14.1/24.5/93.2	81.8/ 2.0/ 6.4/47.0	93.1/35.9/40.3/24.8	93.1/ 1.4/ 4.8/70.4	99.2/**31.3/36.6**/95.4	**99.3**/27.9/33.9/**96.0**
regulator	98.0/7.8/16.1/88.6	95.5/9.1/17.4/76.1	76.6/0.1/0.6/38.1	88.8/18.9/23.6/17.5	84.2/0.4/1.5/44.4	97.6/20.6/29.8/87.0	**99.3/42.2/48.9/95.6**
rolled strip base	**99.7**/31.4/39.9/98.4	99.6/20.7/32.2/97.8	80.5/ 1.7/ 5.1/52.1	99.2/**48.7/50.1**/55.5	87.7/ 0.6/ 3.2/63.4	99.7/37.4/42.5/98.8	**99.7**/41.6/45.5/**98.5**
sim card set	98.5/40.2/44.2/89.5	97.9/31.6/39.8/85.0	71.0/ 6.8/14.3/30.8	**99.1/65.5/62.1**/73.9	89.9/ 1.7/ 5.8/60.4	98.8/51.1/50.6/89.4	99.0/52.1/52.9/**90.9**
switch	94.4/18.9/26.6/90.9	98.1/33.8/40.6/90.7	71.7/ 3.7/ 9.3/44.2	97.4/57.6/55.6/44.7	90.5/ 1.4/ 5.3/64.2	**98.2**/39.9/45.4/92.9	96.7/**62.3/63.6/95.9**
tape	99.7/42.4/47.8/98.4	99.7/29.2/36.9/97.5	77.5/ 1.2/ 3.9/41.4	99.0/61.7/57.6/48.2	81.7/ 0.4/ 2.7/47.3	**99.8**/47.1/48.2/98.0	**99.8/54.0/55.8**/98.8
terminalblock	99.5/27.4/35.8/97.6	99.2/23.1/30.5/94.4	87.0/ 0.8/ 3.6/54.8	96.6/40.6/44.1/34.8	75.5/ 0.1/ 1.1/38.5	**99.8**/35.3/39.7/98.2	**99.8/48.0/50.7/98.8**
toothbrush	96.9/26.1/34.2/88.7	95.7/16.4/25.3/84.3	84.7/ 7.2/14.8/52.6	94.3/30.0/37.3/42.8	82.0/ 1.9/ 6.6/54.5	**97.5**/27.8/36.7/**91.4**	96.9/**38.3/43.9**/90.4
toy	95.2/ 5.1/12.8/82.3	93.4/ 4.6/12.4/70.5	67.7/ 0.1/ 0.4/25.0	86.3/ 8.1/15.9/16.4	82.1/ 1.1/ 4.2/50.3	**96.0**/16.4/25.8/86.3	94.9/**22.5/32.1/91.0**
toy brick	96.4/16.0/24.6/75.3	**97.4**/17.1/27.6/**81.3**	86.5/ 5.2/11.1/56.3	94.7/24.6/30.8/45.5	93.5/ 3.1/ 8.1/66.4	96.6/18.0/25.8/74.7	96.8/**27.9/34.0**/76.6
transistor1	99.1/29.6/35.5/95.1	98.9/25.6/33.2/94.3	71.7/ 5.1/11.3/35.3	97.3/43.8/44.5/45.4	88.6/ 7.2/15.3/58.1	99.4/39.4/40.0/96.5	**99.6/53.5/53.3/97.8**
u block	99.6/40.5/45.2/96.9	99.3/22.3/29.6/94.3	76.2/ 4.8/12.2/34.0	96.9/**57.1/55.7**/38.5	88.8/ 1.6/ 5.4/54.2	**99.5**/37.8/46.1/95.4	**99.5**/41.8/45.6/**96.8**
usb	98.1/26.4/35.2/91.0	97.9/20.6/31.7/85.3	81.1/ 1.5/ 4.9/52.4	98.4/42.2/47.7/57.1	78.0/ 1.0/ 3.1/28.0	**99.2**/39.1/44.4/95.2	**99.2/45.0/48.7/97.5**
usb adaptor	94.5/ 9.8/17.9/73.1	96.6/10.5/19.0/78.4	67.9/ 0.2/ 1.3/28.9	94.9/**25.5/34.9**/36.4	94.0/ 2.3/ 6.6/75.5	97.3/15.3/22.6/82.5	**98.7**/23.7/32.7/**91.0**
vcpill	98.3/43.1/48.6/88.7	99.1/40.7/43.0/91.3	68.2/ 1.1/ 3.3/22.0	97.1/64.7/62.3/42.3	90.2/ 1.3/ 5.2/60.8	98.7/50.2/54.5/89.3	**99.1/66.4/66.7/93.7**
wooden beads	98.0/27.1/34.7/85.7	97.6/16.5/23.6/84.6	68.1/ 2.4/ 6.0/28.3	94.7/38.9/42.9/39.4	85.0/ 1.1/ 4.7/45.6	98.0/32.6/39.8/84.5	**99.1/45.8/50.1/90.5**
woodstick	97.8/30.7/38.4/85.0	94.0/36.2/44.3/77.2	76.1/ 1.4/ 6.0/32.0	97.9/**60.3/60.0**/51.0	90.9/ 2.6/ 8.0/60.7	97.7/40.1/44.9/82.7	**99.0**/50.9/52.1/**90.4**
zipper	99.1/44.7/50.2/96.3	98.4/32.5/36.1/95.1	89.9/23.3/31.2/55.5	98.2/35.3/39.0/78.5	90.2/12.5/18.8/53.5	99.3/58.2/61.3/97.6	**99.3/67.2/66.5/97.8**
Mean	97.3/25.0/32.7/89.6	97.3/21.1/29.2/86.7	75.7/ 2.8/ 6.5/39.0	94.6/37.9/41.7/40.6	88.0/ 2.9/ 7.1/58.1	98.5/33.0/38.7/90.5	**98.8/42.8/47.1/93.9**

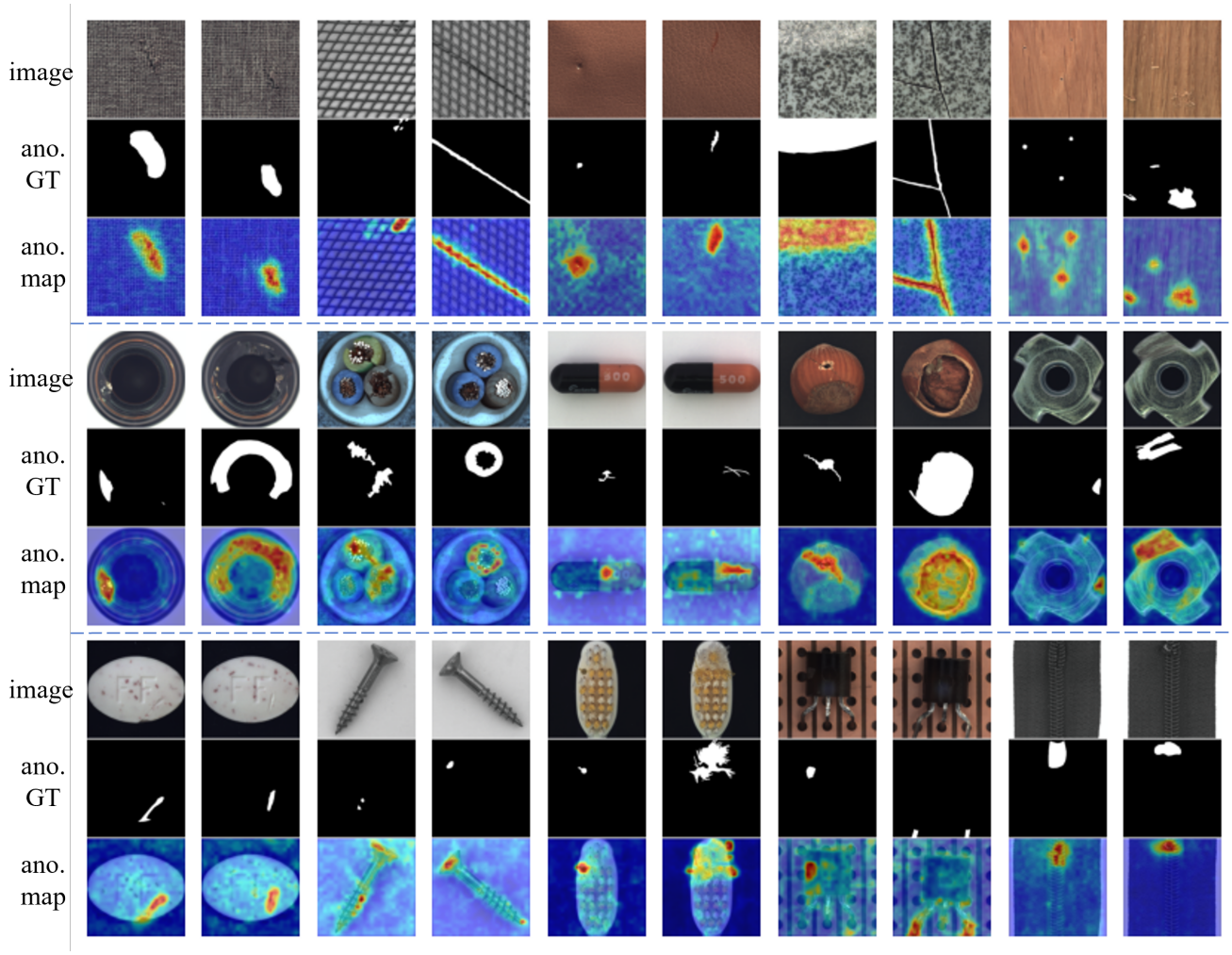

Figure A1. Anomaly maps visualization on MVTec-AD. All samples are randomly chosen.

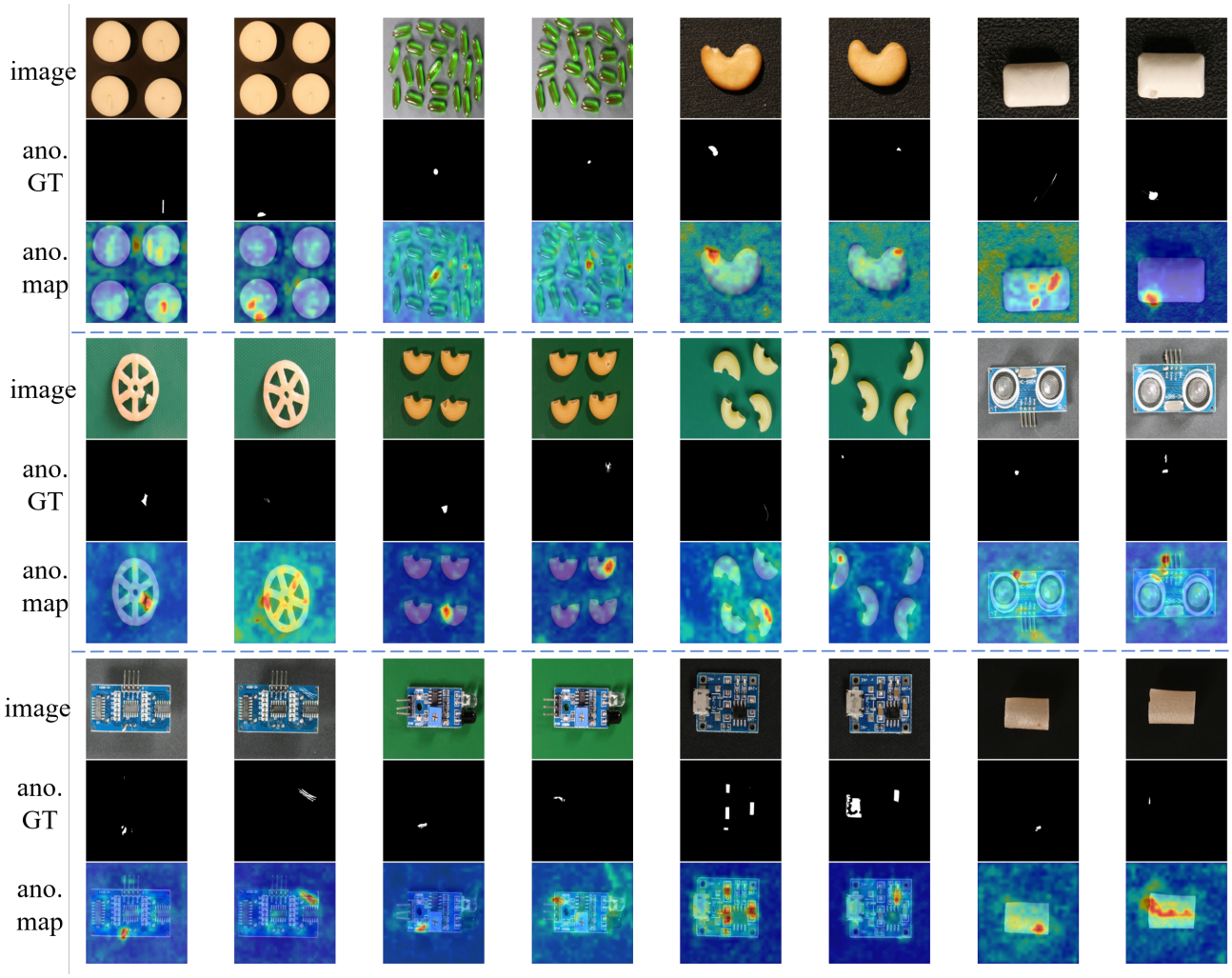

Figure A2. Anomaly maps visualization on VisA. All samples are randomly chosen.

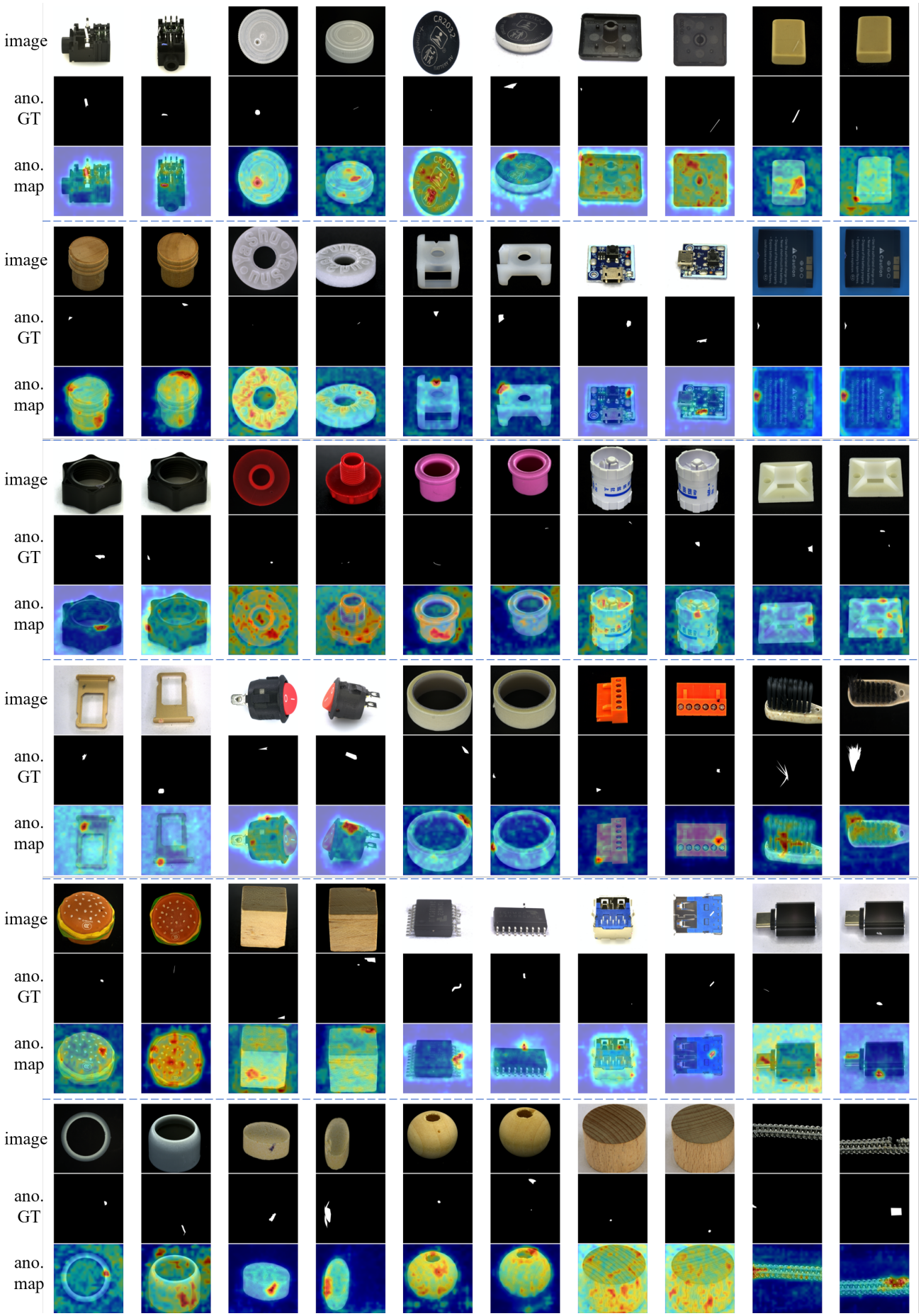

Figure A3. Anomaly maps visualization on Real-IAD. All samples are randomly chosen.

Acknowledgments

The authors acknowledge supports from National Natural Science Foundation of China (U22A2051, 82027807), National Key Research and Development Program of China (2022YFC2405200), Tsinghua-Foshan Innovation Special Fund (2021THFS0104), and Institute for Intelligent Healthcare, Tsinghua University (2022ZLB001).

References

[1] Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III 14*, pages 622–637. Springer, 2019. 1

[2] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. *arXiv preprint arXiv:1807.02011*, 2018. 1

[3] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9592–9600, 2019. 1, 2, 6, 7

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 3, 8, 2

[5] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15750–15758, 2021. 3

[6] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9640–9649, 2021. 3, 8, 2

[7] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. *arXiv preprint arXiv:2309.16588*, 2023. 3, 6, 8, 1, 2

[8] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition*, pages 475–489. Springer, 2021. 1

[9] David Dehaene and Pierre Eline. Anomaly localization by modeling perceptual features. *arXiv preprint arXiv:2008.05369*, 2020. 1

[10] Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9737–9746, 2022. 5, 6, 7, 1, 8, 9, 10

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 2

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2, 3

[13] Jia Guo, Shuai Lu, Lize Jia, Weihang Zhang, and Huiqi Li. Encoder-decoder contrast for unsupervised anomaly detection in medical images. *IEEE Transactions on Medical Imaging*, 2023. 1

[14] Jia Guo, Shuai Lu, Lize Jia, Weihang Zhang, and Huiqi Li. Recontrast: Domain-specific anomaly detection via contrastive reconstruction. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 10721–10740, 2023. 2, 3, 5, 6

[15] Dongchen Han, Xuran Pan, Yizeng Han, Shiji Song, and Gao Huang. Flatten transformer: Vision transformer using focused linear attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5961–5971, 2023. 5

[16] Haoyang He, Jiangning Zhang, Hongxu Chen, Xuhai Chen, Zhishan Li, Xu Chen, Yabiao Wang, Chengjie Wang, and Lei Xie. Diad: A diffusion-based framework for multi-class anomaly detection. *arXiv preprint arXiv:2312.06607*, 2023. 1, 3

[17] Haoyang He, Yuhu Bai, Jiangning Zhang, Qingdong He, Hongxu Chen, Zhenye Gan, Chengjie Wang, Xiangtai Li, Guanzhong Tian, and Lei Xie. Mambaad: Exploring state space models for multi-class unsupervised anomaly detection. *arXiv preprint arXiv:2404.06564*, 2024. 1, 3, 6, 7, 8, 10

[18] Haoyang He, Jiangning Zhang, Hongxu Chen, Xuhai Chen, Zhishan Li, Xu Chen, Yabiao Wang, Chengjie Wang, and Lei Xie. A diffusion-based framework for multi-class anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8472–8480, 2024. 2, 6, 7, 8, 9, 10

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 3, 8, 2, 5

[20] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 5

[21] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012. 4

[22] Chaoqin Huang, Haoyan Guan, Aofan Jiang, Ya Zhang, Michael Spratling, and Yan-Feng Wang. Registration based few-shot anomaly detection. In *European Conference on Computer Vision*, pages 303–319. Springer, 2022. 7

[23] Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. Winclip: Zero-/few-shot anomaly classification and segmentation. *arXiv preprint arXiv:2303.14814*, 2023. 7

[24] Stepan Jezek, Martin Jonak, Radim Burget, Pavel Dvorak, and Milos Skotak. Deep learning-based defect detection of

metal parts: evaluating current methods in complex conditions. In *2021 13th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pages 66–71. IEEE, 2021. 7, 6

[25] Xi Jiang, Jianlin Liu, Jinbao Wang, Qiang Nie, Kai Wu, Yong Liu, Chengjie Wang, and Feng Zheng. Softpatch: Unsupervised anomaly detection with noisy data. *Advances in Neural Information Processing Systems*, 35:15433–15445, 2022. 6, 7

[26] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pages 5156–5165. PMLR, 2020. 5

[27] Sungwook Lee, Seunghyun Lee, and Byung Cheol Song. Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. *IEEE Access*, 10: 78446–78454, 2022. 1

[28] Yujin Lee, Harin Lim, and Hyunsoo Yoon. Selformaly: Towards task-agnostic unified anomaly detection. *arXiv preprint arXiv:2307.12540*, 2023. 3

[29] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9664–9674, 2021. 1

[30] Chen Liang, Jiahui Yu, Ming-Hsuan Yang, Matthew Brown, Yin Cui, Tuo Zhao, Boqing Gong, and Tianyi Zhou. Module-wise adaptive distillation for multimodality foundation models. *Advances in Neural Information Processing Systems*, 36, 2024. 5

[31] Jiangqi Liu and Feng Wang. mixed attention auto encoder for multi-class industrial anomaly detection. *arXiv preprint arXiv:2309.12700*, 2023. 2

[32] Wenqian Liu, Runze Li, Meng Zheng, Srikrishna Karanam, Ziyan Wu, Bir Bhanu, Richard J Radke, and Octavia Camps. Towards visually explaining variational autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8642–8651, 2020. 1

[33] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 3

[34] Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. Simplenet: A simple network for image anomaly detection and localization. *arXiv preprint arXiv:2303.15140*, 2023. 6, 7, 1, 8, 9, 10

[35] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[36] Ruiying Lu, YuJie Wu, Long Tian, Dongsheng Wang, Bo Chen, Xiyang Liu, and Ruimin Hu. Hierarchical vector quantized transformer for multi-class unsupervised anomaly detection. *arXiv preprint arXiv:2310.14228*, 2023. 1, 2, 4

[37] Amira Ben Mabrouk and Ezzeddine Zagrouba. Abnormal behavior recognition for intelligent video surveillance systems: A review. *Expert Systems with Applications*, 91:480–491, 2018. 1

[38] Pankaj Mishra, Riccardo Verk, Daniele Fornasier, Claudio Piciarelli, and Gian Luca Foresti. Vt-adl: A vision transformer network for image anomaly detection and localization. In *2021 IEEE 30th International Symposium on Industrial Electronics (ISIE)*, pages 01–06. IEEE, 2021. 7, 6

[39] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3, 8, 2

[40] Z Peng, L Dong, H Bao, Q Ye, and F Wei. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*, 2022. 3, 2

[41] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, 2019. 6, 1

[42] Tal Reiss, Niv Cohen, Liron Bergman, and Yedid Hoshen. Panda: Adapting pretrained features for anomaly detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2806–2814, 2021. 7

[43] Tal Reiss, Niv Cohen, Eliahu Horwitz, Ron Abutbul, and Yedid Hoshen. Anomaly detection requires better representations. In *European Conference on Computer Vision*, pages 56–68. Springer, 2022. 3, 7

[44] Sucheng Ren, Zeyu Wang, Hongru Zhu, Junfei Xiao, Alan Yuille, and Cihang Xie. Rejuvenating image-gpt as strong visual representation learners. *arXiv preprint arXiv:2312.02147*, 2023. 8, 2

[45] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14318–14328, 2022. 7, 1, 3

[46] Mohammadreza Salehi, Niousha Sadjadi, Soroosh Baselizadeh, Mohammad H Rohban, and Hamid R Rabiee. Multiresolution knowledge distillation for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14902–14912, 2021. 5, 1

[47] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. *Medical image analysis*, 54:30–44, 2019. 1

[48] Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3531–3539, 2021. 5

[49] Shelly Sheynin, Sagie Benaim, and Lior Wolf. A hierarchical transformation-discriminating generative model for few shot anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8495–8504, 2021. 1

[50] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herv'e J'egou. Training data-efficient image transformers & distillation through attention. *arXiv preprint arXiv:2012.12877*, 2021. 8, 2

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 2

[52] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008. 4

[53] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. In *Journal of Machine Learning Research*, pages 3371–3408, 2010. 4

[54] Chengjie Wang, Wenbing Zhu, Bin-Bin Gao, Zhenye Gan, Jianning Zhang, Zhihao Gu, Shuguang Qian, Mingang Chen, and Lizhuang Ma. Real-iad: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. *arXiv preprint arXiv:2403.12580*, 2024. 2, 6, 7, 3

[55] Guodong Wang, Shumin Han, Errui Ding, and Di Huang. Student-teacher feature pyramid matching for anomaly detection. In *The British Machine Vision Conference (BMVC)*, 2021. 1

[56] Mitchell Wortsman, Tim Dettmers, Luke Zettlemoyer, Ari Morcos, Ali Farhadi, and Ludwig Schmidt. Stable and low-precision training for large-scale vision-language models. *Advances in Neural Information Processing Systems*, 36: 10271–10298, 2023. 6, 1

[57] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9653–9663, 2022. 3

[58] Jie Yang, Yong Shi, and Zhiquan Qi. Dfr: Deep feature reconstruction for unsupervised anomaly segmentation. *arXiv preprint arXiv:2012.07122*, 2020. 5, 1

[59] Haonan Yin, Guanlong Jiao, Qianhui Wu, Borje F Karlsson, Biqing Huang, and Chin Yew Lin. Lafite: Latent diffusion model with feature editing for unsupervised multi-class anomaly detection. *arXiv preprint arXiv:2307.08059*, 2023. 1, 2, 3, 4

[60] Zhiyuan You, Lei Cui, Yujun Shen, Kai Yang, Xin Lu, Yu Zheng, and Xinyi Le. A unified model for multi-class anomaly detection. *arXiv preprint arXiv:2206.03687*, 2022. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

[61] Weihao Yu, Chenyang Si, Pan Zhou, Mi Luo, Yichen Zhou, Jiashi Feng, Shuicheng Yan, and Xinchao Wang. Metaformer baselines for vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 4

[62] Vitjan Zavrtanik, Matej Kristan, and Danijel Skoč aj. Draema discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8330–8339, 2021. 4, 1

[63] Vitjan Zavrtanik, Matej Kristan, and Danijel Skocaj. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112:107706, 2021. 1

[64] Dingwen Zhang, Guohai Huang, Qiang Zhang, Jungong Han, Junwei Han, Yizhou Wang, and Yizhou Yu. Exploring task structure for brain tumor segmentation from multimodality mr images. *IEEE Transactions on Image Processing*, 29:9032–9043, 2020. 7

[65] Jiangning Zhang, Xuhai Chen, Yabiao Wang, Chengjie Wang, Yong Liu, Xiangtai Li, Ming-Hsuan Yang, and Dacheng Tao. Exploring plain vit reconstruction for multi-class unsupervised anomaly detection. *arXiv preprint arXiv:2312.07495*, 2023. 3, 5, 6, 7

[66] Jiangning Zhang, Haoyang He, Zhenye Gan, Qingdong He, Yuxuan Cai, Zhucun Xue, Yabiao Wang, Chengjie Wang, Lei Xie, and Yong Liu. Ader: A comprehensive benchmark for multi-class visual anomaly detection. *arXiv preprint arXiv:2406.03262*, 2024. 7, 2, 6

[67] Xuan Zhang, Shiyu Li, Xi Li, Ping Huang, Jiulong Shan, and Ting Chen. Destseg: Segmentation guided denoising student-teacher for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3914–3923, 2023. 4, 6, 1, 7, 8, 9, 10

[68] Ying Zhao. Omnial: A unified cnn framework for unsupervised anomaly localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3924–3933, 2023. 1, 3

[69] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021. 3, 8, 2

[70] Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *European Conference on Computer Vision*, pages 392–408. Springer, 2022. 2, 6, 7, 3