Appendix

Overview

The supplementary material presents the following sections to strengthen the main manuscript:

- **Sec. A** shows more implementation details.
- **Sec. B** shows more details about comparison methods.
- **Sec. C** shows the complexity comparisons.
- **Sec. D** shows the influence of input resolution.
- **Sec. E** shows the influence of ViT architecture.
- **Sec. F** shows the influence of the weight of loss functions.
- **Sec. G** shows the super-multi-class anomaly detection.
- **Sec. H** shows per-class multi-class anomaly detection results.
- **Sec. I** shows more few-shot anomaly detection results.
- **Sec. J** shows per-class single-class anomaly detection results.
- **Sec. K** shows more zero-shot anomaly detection results.
- **Sec. L** shows more visualized anomaly localization results
- **Sec. M** shows a more detailed analysis of the limitations
- **Sec. N** shows a comparison of INP with handcrafted aggregated prototypes
- **Sec. O** shows a comparison of INP with MuSc
- **Sec. P** shows more INP visualization results

A. More implementation details

Building on previous work [7], we adopt a group-to-group supervision approach by summing the features of the layers of interest to form distinct groups. In our study, we define two groups: the features from layers 3 to 6 of ViT-Base [6] constitute one group, while those from layers 7 to 10 form another. We construct the anomaly detection map using the regional cosine distance [5] between the feature groups of the encoder and decoder, computing the average of the top 1% of this map as the image-level anomaly score. In the few-shot setting, we employ data augmentation techniques similar to RegAD [10]. Additionally, it is worth noting that in our few-shot experiments on the Real-IAD [16] dataset, the term "shot" refers to the number of images rather than the number of views. The experimental code is implemented in Python 3.8 and PyTorch 2.0.0 (CUDA 11.8) and runs on an NVIDIA GeForce RTX 4090 GPU (24GB).

Table S1. **Comparison of computational efficiency among SOTA methods**. mAD represents the average value of seven metrics on the Real-IAD [16] dataset. The **INP-Former-S** denotes a model variant based on the ViT-Small architecture, while **INP-Former-S*** refers to a model variant using the ViT-Small architexture with an image size of R256^2-C224^2.

Method	Params(M)	FLOPs(G)	mAD
RD4AD [5]	150.6	38.9	68.6
UniAD [17]	**24.5**	**3.6**	67.5
SimpleNet [14]	72.8	16.1	42.3
DeSTSeg [18]	35.2	122.7	64.2
DiAD [9]	1331.3	451.5	52.6
MambaAD [8]	<u>25.7</u>	8.3	72.7
Dinomaly [7]	132.8	104.7	77.0
INP-Former	139.8	98.0	**78.8**
INP-Former-S	35.1	24.6	<u>78.4</u>
INP-Former-S*	35.1	<u>8.1</u>	73.8

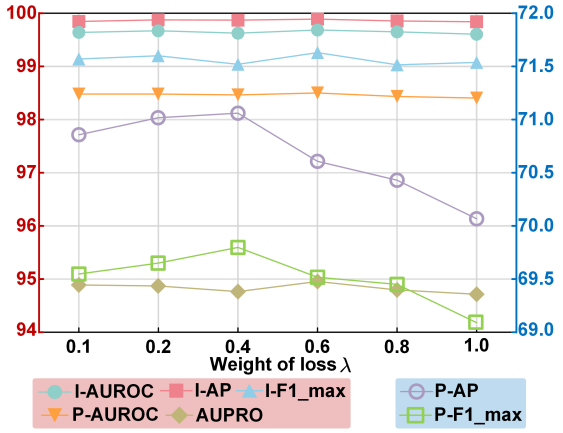

Figure S1. **Influence of the weight of loss function λ on model performance for the MVTec-AD [2] dataset**. Pixel- level AP and F1_max use the right vertical axis, while the other metrics share the left vertical axis.

B. More details about comparison methods

The detailed information of the other compared methods in the experiment are as follows. Unless otherwise indicated, we utilize the performance metrics as reported in the original paper. In the few-shot setting on the Real-IAD [16] dataset, SPADE [3] [1], PaDiM [4] [2], PatchCore [15] [3], WinCLIP [11] [4], and PromptAD [13] [5] are run with the publicly

[1] https://github.com/byungjae89/SPADE-pytorch
[2] https://github.com/xiahaifeng1995/PaDiM-Anomaly-Detection-Localization-master
[3] https://github.com/hcw-00/PatchCore_anomaly_detection
[4] https://github.com/zqhang/Accurate-WinCLIP-pytorch
[5] https://github.com/FuNz-0/PromptAD

Table S2. Influence of the **Image Size** on model performance for the MVTec-AD [2] dataset. $R256^2$-$C224^2$ denotes resizing the image to 256×256, followed by a center crop to 224×224.

Metric →	Image-level			Pixel-level				Efficiency	
Image Size ↓	AUROC	AP	F1_max	AUROC	AP	F1_max	AUPRO	Params(M)	FLOPs(G)
$R224^2$	99.3	99.8	98.8	98.2	60.8	61.9	93.6	139.8	**32.3**
$R256^2$-$C224^2$	99.3	99.8	99.0	98.1	64.2	64.4	92.7	139.8	**32.3**
$R280^2$	99.5	**99.9**	**99.2**	98.4	64.9	64.8	94.6	139.8	50.2
$R320^2$-$C280^2$	99.6	**99.9**	99.1	98.3	67.5	67.1	93.9	139.8	50.2
$R392^2$	99.6	99.8	99.1	**98.6**	69.1	68.5	**95.6**	139.8	98.0
$R448^2$-$C392^2$	**99.7**	**99.9**	**99.2**	98.5	**71.0**	**69.7**	94.9	139.8	98.0

Table S3. Influence of the **Image Size** on the performance of other methods on MVTec-AD [2] dataset.

Method	Input Size	Image-level	Pixel-level
RD4AD [5]	$R256^2$	94.6/96.5/96.1	96.1/48.6/53.8/91.1
	$R384^2$	91.9/96.2/95.0	94.0/47.8/50.9/88.6
	△	-2.7/0.3/1.1	-2.1/0.8/2.9/2.5
SimpleNet [14]	$R256^2$	95.3/98.4/95.8	96.9/45.9/49.7/86.5
	$R384^2$	86.1/93.6/90.9	89.5/36.0/40.5/76.4
	△	-9.2/4.8/4.9	-7.4/9.9/9.2/10.1
PatchCore [15]	$R256^2$	97.2/99.1/97.2	97.9/53.8/56.3/91.3
	$R384^2$	98.9/99.6/98.3	98.0/58.4/59.8/93.2
	△	+1.7/0.5/1.1	+0.1/4.6/3.5/1.9

available implementations.

RD4AD [5]: RD4AD is a robust baseline model for anomaly detection methods based on knowledge distillation and has been widely adopted by subsequent researchers.

UniAD [17]: UniAD is a baseline model for multi-class anomaly detection, which employs a Transformer-based non-identical mapping reconstruction model to enable complex multi-class semantic learning. Similarly,

SimpleNet [14]: SimpleNet is an efficient and user-friendly network for anomaly detection and localization, which relies on a binary discriminator of adapted features to distinguish between anomalies and normal samples.

DeSTSeg [18]: DeSTSeg is an improved student-teacher framework for visual anomaly detection, integrating a denoising encoder-decoder and a segmentation network.

DiAD [9]: DiAD is a diffusion-based framework for multi-class anomaly detection, which incorporates the Semantic Guided network to recover anomalies while preserving semantics.

MambaAD [8]: MambaAD is a recently developed multi-class anomaly detection model with a Mamba decoder and locality-enhanced state space module, which captures long-range and local information effectively.

Dinomaly [7]: Dinomaly is a streamlined reverse distillation framework that employs linear attention mechanisms and loose reconstruction to achieve substantial performance gains.

SPADE [3]: SPADE is an early anomaly detection method that aligns anomalous images with normal images using a multi-resolution feature pyramid.

PaDiM [4]: PaDiM utilizes the pre-trained CNN features of normal samples to fit multivariate Gaussian distributions, which is a widely used baseline model.

Patchcore [15]: PatchCore is an important milestone approach. It utilizes a memory bank of core set sampled nominal patch features.

WinCLIP [11]: WinCLIP introduces the first VLM-driven approach for zero-shot anomaly detection. It meticulously crafts a comprehensive suite of custom text prompts, optimized for identifying anomalies, and integrates a window scaling technique to achieve anomaly segmentation.

PromptAD [13]: PromptAD improves few-shot anomaly detection by automating prompt learning for one-class settings. It employs semantic concatenation to generate anomaly prompts and introduces an explicit margin.

C. Complexity Comparisons

Tab. S1 compares the proposed INP-Former with seven SOTA methods in terms of model size and computational complexity. Notably, our method's FLOPs are lower than those of DeSTSeg, DiAD, and Dinomaly, while its performance significantly exceeds theirs. Although our method has a larger parameter size and FLOPs than SimpleNet, UniAD, and MambaAD, it demonstrates a substantial improvement in detection performance. Furthermore, our approach is applicable to multi-class, few-shot, and single-class anomaly detection settings. It is noteworthy that we also report the efficiency and performance of two additional variants of INP-Former (INP-Former-S and INP-Former-S*). INP-Former-S achieves a significant reduction in both parameters and FLOPs, with only a minor performance decline of 0.4↓. **Even more remarkably, INP-Former-S* not only reduces FLOPs compared to MambaAD but also outperforms MambaAD 1.1↑ in terms of performance.** Overall, our method shows significant potential in industrial applications.

Table S4. Influence of the **ViT Architecture** on model performance for the MVTec-AD [2] dataset.

Metric →	Image-level			Pixel-level				Efficiency	
Architecture ↓	AUROC	AP	F1_max	AUROC	AP	F1_max	AUPRO	Params(M)	FLOPs(G)
ViT-Small	99.2	99.7	98.6	98.2	69.1	68.5	94.3	**35.1**	**24.6**
ViT-Base	99.7	**99.9**	99.2	98.5	71.0	69.7	94.9	139.8	98.0
ViT-Large	**99.8**	**99.9**	**99.4**	**98.6**	**72.1**	**70.5**	**95.6**	361.7	263.4

Table S5. **Super-multi-class** anomaly detection performance on different AD datasets. Δ represents the performance change of INP-Former in the super-multi-class setting relative to the multi-class setting.

Dataset →	MVTec-AD [2]		VisA [19]		Real-IAD [16]	
Metric →	Image-level(I-AUROC/I-AP/I-F1_max)		Pixel-level(P-AUROC/P-AP/P-F1_max/AUPRO)			
Setting ↓	Image-level	Pixel-level	Image-level	Pixel-level	Image-level	Pixel-level
Multi-Class	99.7/99.9/99.2	98.5/71.0/69.7/94.9	98.9/99.0/96.6	98.9/51.2/54.7/94.4	90.5/88.1/81.5	99.0/47.5/50.3/95.0
Super-Multi-Class	99.5/99.8/98.9	98.1/69.2/68.1/94.2	97.3/97.8/94.1	98.4/51.4/54.7/92.4	89.8/87.4/80.5	98.9/45.2/48.6/94.4
Δ	0.2↓/0.1↓/0.3↓	0.4↓/1.8↓/1.6↓/0.7↓	1.6↓/1.2↓/2.5↓	0.5↓/0.2↑/0.0/2.0↓	0.7↓/0.7↓/1.0↓	0.1↓/2.3↓/1.7↓/0.6↓

D. Influence of Input Resolution

As shown in Tab. S2, we conducted an ablation study to evaluate the impact of input resolution on model performance. The results demonstrate that our method is robust to variations in image size for image-level anomaly detection. However, the image size has a slight effect on pixel-level anomaly localization performance. This is attributed to the patch size of 14 in the ViT, which results in smaller feature maps when the input image is reduced in size, leading to performance degradation. Therefore, in our study, we default to resizing the image to 448×448 and then applying a center crop to 392×392. Additionally, it is noteworthy that, under the R256^2-C224^2 setting, our method still achieves superior detection and localization performance compared to previous SOTA methods. Additionally, we analyze the effect of input size on the performance of other methods. As shown in Tab. S3, we observe that not all models show improved performance with larger input sizes. For instance, when the input size is increased from 256 to 384, the performance of RD4AD and SimpleNet drops significantly. In contrast, our method consistently demonstrates superior detection performance across various input sizes, further validating the effectiveness of our approach.

E. Influence of ViT Architectures.

Tab. S4 illustrates the effect of the ViT architecture on model performance. Our method demonstrates strong detection performance even with ViT-Small, with performance further improving as the ViT model size increases. Although ViT-Large achieves the best performance, its high FLOPs and parameter count make it less practical. Therefore, we default to using ViT-Base in this study.

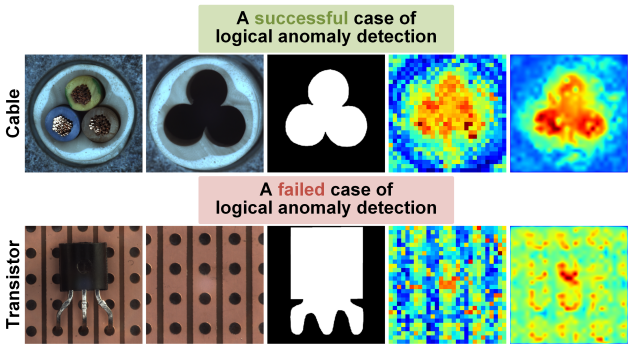

Figure S2. **Limitation of proposed method in detecting logical anomalies similar to the background**. From left to right: Normal Image, Input Anomaly, Ground Truth, Distance Map, and Predicted Anomaly Map.

F. Influence of the Weight of Loss Functions

Fig. S1 illustrates the effect of the weight of loss function on model performance in the MVTec-AD [2] dataset. Our method shows strong robustness to changes in weight of loss function at the image level. However, pixel-level performance initially increases and then decreases as the λ grows. This trend occurs because, when λ is too low, the INP Extractor may fail to consistently capture normal patterns, potentially including some anomalous information. Conversely, when λ is too high, the model focuses excessively on updating the INP Extractor, overlooking updates to the INP-Guided Decoder, which leads to insufficient detail in reconstructed features. Based on these observations, we set λ to 0.2 in our study.

Table S6. **Few-shot (1-shot)** anomaly detection performance on different AD datasets. The best in **bold**, the second-highest is underlined. † indicates the results we reproduced using publicly available code.

Dataset →	MVTec-AD [2]		VisA [19]		Real-IAD [16]	
Metric →	Image-level(I-AUROC/I-AP/I-F1_max)		Pixel-level(P-AUROC/P-AP/P-F1_max/AUPRO)			
Method ↓	Image-level	Pixel-level	Image-level	Pixel-level	Image-level	Pixel-level
SPADE [3]	82.9/91.7/91.1	92.0/-/44.5/85.7	79.5/82.0/80.7	95.6/-/35.5/84.1	51.2†/45.6†/61.4†	59.5†/0.2†/0.5†/19.3†
PaDiM [4]	78.9/89.3/89.2	91.3/-/43.7/78.2	62.8/68.3/75.3	89.9/-/17.4/64.3	52.9†/47.4†/62.0†	84.9†/0.8†/2.3†/52.7†
PatchCore [15]	86.3/93.8/92.0	93.3/-/53.0/82.3	79.9/82.8/81.7	95.4/-/38.0/80.5	59.3†/55.8†/62.3†	89.6†/6.6†/12.3†/60.5†
WinCLIP [11]	93.1/96.5/93.7	95.2/-/55.9/87.1	83.8/85.1/83.1	96.4/-/41.3/85.1	**69.4**†/56.8†/58.8†	91.9†/9.0†/15.3†/71.0†
PromptAD [13]	94.6/-/-	95.9/-/-/87.9	86.9/-/-	**96.7**/-/-/85.8	52.2†/41.6†/52.2†	84.9†/7.6†/14.6†/58.4†
INP-Former	**96.6/98.2/96.4**	**97.0/64.2/64.0/92.6**	**91.4/92.2/88.6**	96.3/**42.5/47.3/89.5**	67.5/**63.1/66.1**	**94.9/20.0/25.8/81.8**

Table S7. **Few-shot (2-shot)** anomaly detection performance on different AD datasets. The best in **bold**, the second-highest is underlined. † indicates the results we reproduced using publicly available code.

Dataset →	MVTec-AD [2]		VisA [19]		Real-IAD [16]	
Metric →	Image-level(I-AUROC/I-AP/I-F1_max)		Pixel-level(P-AUROC/P-AP/P-F1_max/AUPRO)			
Method ↓	Image-level	Pixel-level	Image-level	Pixel-level	Image-level	Pixel-level
SPADE [3]	81.0/90.6/90.3	91.2/-/42.4/83.9	81.7/83.4/82.1	96.2/-/40.5/85.7	50.9†/45.5†/61.2†	59.5†/0.2†/0.5†/19.2†
PaDiM [4]	76.6/88.1/88.2	89.3/-/40.2/73.3	67.4/71.6/75.7	92.0/-/21.1/70.1	55.9†/49.6†/62.9†	88.5†/1.5†/3.8†/61.6†
PatchCore [15]	83.4/92.2/90.5	92.0/-/58.4/79.7	81.6/84.8/82.5	96.1/-/41.0/82.6	63.3†/59.7†/64.2†	92.0†/9.4†/14.1†/66.1†
WinCLIP [11]	94.4/97.0/94.4	96.0/-/58.4/88.4	84.6/85.8/83.0	96.8/-/43.5/86.2	**70.9**†/58.7†/60.3†	93.2†/11.7†/18.3†/74.7†
PromptAD [13]	95.7/-/-	96.2/-/-/88.5	88.3/-/-	97.1/-/-/85.8	57.7†/41.1†/52.9†	86.4†/8.5†/16.2†/61.0†
INP-Former	**97.0/98.2/96.7**	**97.2/66.0/65.6/93.1**	**94.6/94.9/90.8**	**97.2/45.0/50.4/91.8**	70.6/**66.1/69.3**	**96.0/23.8/28.3/83.8**

G. Super-Multi-Class Anomaly Detection

Tab. S5 presents the super-multi-class anomaly detection performance of INP-Former, *i.e.*, training together with MVTec-AD, VisA, and Real-IAD. Compared to the multi-class anomaly detection setting, the performance of INP-Former in the super-multi-class setting only slightly declines. This demonstrates that our method can utilize a unified model to detect a broader range of products, which can significantly reduce memory consumption in industrial applications.

H. Per-Class Multi-Class Anomaly Detection Results

In this section, we present the performance of each class on the MVTec-AD [2], VisA [19], and Real-IAD [16] datasets for multi-class anomaly detection. The performance of the comparison methods is derived from MambaAD [8] and Dinomaly [7]. Tab. S12 and Tab. S13 provide the results for image-level anomaly detection and pixel-level anomaly localization on the MVTec-AD dataset, respectively. Tab. S14 and Tab. S15 further present the corresponding results on the VisA dataset. Tab. S16 and Tab. S17 display the results for image-level anomaly detection and pixel-level anomaly localization on the Real-IAD dataset. These results convincingly demonstrate the superiority of our proposed method.

I. More Few-shot Anomaly Detection Results

Tab. S6 and Tab. S7 show the performance comparison between our method and existing methods across three datasets under 1-shot and 2-shot anomaly detection settings, respectively. Our method achieves state-of-the-art or competitive results across all three datasets, highlighting its superior effectiveness.

J. Per-Class Single-Class Anomaly Detection Results

To support future research, we report the per-class performance of INP-Former in the single-class anomaly detection setting on MVTec-AD [2], VisA [19], and Real-IAD [16] datasets. in Tab. S9, Tab. S10, and Tab. S11, respectively.

K. More Zero-shot Anomaly Detection Results

Tab. S8 compares the zero-shot anomaly detection performance of our method with WinCLIP [11], a method specifically designed for zero-shot anomaly detection. Notably, we utilize INP-Former to extract INPs for images from unseen classes and then directly compare all tokens to these INPs for zero-shot anomaly detection. Although

Table S8. **Zero-shot** anomaly detection performance on different AD datasets. The best in **bold**.

Dataset →	MVTec-AD [2]		VisA [19]	
Metric →	Image-level(I-AUROC/I-AP/I-F1_max)		Pixel-level(P-AUROC/P-AP/P-F1_max/AUPRO)	
Method ↓	Image-level	Pixel-level	Image-level	Pixel-level
WinCLIP [11]	**91.8/96.5/92.9**	85.1/-/31.7/64.6	**78.1/81.2/79.0**	79.6/-/**14.8**/56.8
INP-Former	80.8/90.7/89.1	**88.0/36.1/39.5/76.9**	67.5/71.6/75.0	**88.7/7.8**/11.8/**67.2**

our method is not designed for zero-shot anomaly detection, it still possesses some efficacy for this task, with 88.0 and 88.7 pixel-level AUROCs on MVTec-AD and VisA, respectively. In terms of image-level performance, our method performs weaker than the existing specified method. We believe incorporating INPs with other specified designs can bring better zero-shot anomaly detection performance.

L. More qualitative results

Fig. S3, Fig. S4, and Fig. S5 display the predicted anomaly maps of our method on the MVTec-AD [2], VisA [19], and Real-IAD [16] datasets for multi-class anomaly detection. These results clearly indicate that our approach can accurately localize anomalous regions for a wide range of categories.

M. More detailed analysis of the limitations

Fig. S2 illustrates two examples of logical anomaly detection using our method. Interestingly, the misplaced logical anomaly in Cable is successfully detected, while the misplaced anomaly in Transistor is completely missed. We hypothesize that this is due to the significant difference between the misplaced anomaly and the background in Cable, whereas the misplaced anomaly in Transistor closely resembles the background. As a result, the INP Extractor mistakenly extracts the misplaced anomaly in Transistor as INPs, leading to a missed detection. This highlights a limitation of our method when dealing with logical anomalies that are similar to the background. In future work, we aim to combine pre-stored prototypes with INPs to address this issue. Pre-stored prototypes capture comprehensive semantic information, while INPs exhibit strong alignment. The integration of both is expected to improve the model's performance in detecting logical anomalies that resemble the background.

N. Comparison of INP with handcrafted aggregated prototypes

Although the concept in Reference [1] is similar to our proposed INP, we wish to emphasize that our method is fundamentally distinct. Reference [1] manually aggregates features within a single image as prototypes, and its scope is limited to zero-shot texture anomaly detection. In contrast, we introduce a learnable INP extractor that extracts normal features with adaptable shapes as INPs. This enables our method to be applied not only to textures but also to objects. Additionally, we integrate the INP into a reconstruction framework by proposing an INP-guided decoder, which not only reduces the computational cost of self-attention but also achieves superior detection performance across multiple settings.

O. Comparision of INP with MuSc

It may seem unusual that MuSc [12] performs better in zero-shot settings compared to our INP-Former in few-shot settings. However, this difference stems from the distinct setups of the two methods. MuSc is specifically designed for zero-shot detection and relies on a large number of test images for mutual scoring. In contrast, our INP-Former only requires a single image during the testing phase, making it adaptable to various settings. As such, comparing our method with MuSc is not a fair comparison.

P. More visualizations of INPs

Fig. S6 presents the cross-attention maps between INPs and image patches. This clearly demonstrates that our INPs are able to capture semantic information from various regions, including object regions, object boundaries, and background areas.

References

[1] Toshimichi Aota, Lloyd Teh Tzer Tong, and Takayuki Okatani. Zero-shot versus many-shot: Unsupervised texture anomaly detection. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5553–5561, 2023. 5

[2] Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. The MVTec anomaly detection dataset: A comprehensive real-world dataset for unsupervised anomaly detection. *International Journal of Computer Vision*, 129(4):1038–1059, 2021. 1, 2, 3, 4, 5, 7, 9, 12

[3] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. *arXiv preprint arXiv:2005.02357*, 2020. 1, 2, 4

[4] 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, 2, 4

[5] Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9737–9746, 2022. 1, 2, 9, 10, 11

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 1

[7] Jia Guo, Shuai Lu, Weihang Zhang, Fang Chen, Hongen Liao, and Huiqi Li. Dinomaly: The less is more philosophy in multi-class unsupervised anomaly detection. *arXiv preprint arXiv:2405.14325*, 2024. 1, 2, 4, 9, 10, 11

[8] 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. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 1, 2, 4, 9, 10, 11

[9] 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. 1, 2, 9, 10, 11

[10] 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. 1

[11] Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. Winclip: Zero-/few-shot anomaly classification and segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19606–19616, 2023. 1, 2, 4, 5

[12] Xurui Li, Ziming Huang, Feng Xue, and Yu Zhou. Musc: Zero-shot industrial anomaly classification and segmentation with mutual scoring of the unlabeled images. In *The Twelfth International Conference on Learning Representations*, 2024. 5

[13] Xiaofan Li, Zhizhong Zhang, Xin Tan, Chengwei Chen, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Promptad: Learning prompts with only normal samples for few-shot anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16838–16848, 2024. 1, 2, 4

[14] Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. Simplenet: A simple network for image anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20402–20411, 2023. 1, 2, 9, 10, 11

[15] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14318–14328, 2022. 1, 2, 4

[16] Chengjie Wang, Wenbing Zhu, Bin-Bin Gao, Zhenye Gan, Jiangning Zhang, Zhihao Gu, Shuguang Qian, Mingang Chen, and Lizhuang Ma. Real-iad: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22883–22892, 2024. 1, 3, 4, 5, 8, 10, 11, 14

[17] Zhiyuan You, Lei Cui, Yujun Shen, Kai Yang, Xin Lu, Yu Zheng, and Xinyi Le. A unified model for multi-class anomaly detection. In *Advances in Neural Information Processing Systems*, pages 4571–4584, 2022. 1, 2, 9, 10, 11

[18] 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. 1, 2, 9, 10, 11

[19] 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. 3, 4, 5, 7, 9, 10, 13

Table S9. **Per-Class Performance of the Proposed INP-Former on the MVTec-AD [2] Dataset for Single-Class Anomaly Detection**

Metric →	Image-level			Pixel-level			
Category ↓	I-AUROC	I-AP	I-F1_max	P-AUROC	P-AP	P-F1_max	AUPRO
Bottle	100	100	100	99.1	88.9	82.4	97.2
Cable	100	100	100	98.8	78.9	75.0	95.2
Capsule	98.6	99.7	98.2	98.5	60.1	57.5	97.7
Hazelnut	100	100	100	99.5	82.8	78.4	97.0
Metal Nut	100	100	100	97.1	81.1	86.3	94.3
Pill	99.2	99.9	98.6	96.0	66.8	66.9	97.2
Screw	98.0	99.3	96.7	99.6	63.8	59.9	98.3
Toothbrush	100	100	100	99.1	57.4	67.4	95.8
Transistor	99.9	99.8	98.8	95.6	66.7	63.7	86.4
Zipper	100	100	100	98.1	71.6	68.3	94.6
Carpet	99.9	99.9	99.4	99.4	75.7	72.6	98.0
Grid	100	100	100	99.5	61.6	62.0	97.6
Leather	100	100	100	99.3	52.2	53.3	98.4
Tile	100	100	100	97.8	73.0	75.7	88.9
Wood	99.8	99.9	99.2	97.4	72.1	68.4	94.1
Mean	99.7	99.9	99.4	98.3	70.2	69.2	95.4

Table S10. **Per-Class Performance of the Proposed INP-Former on the VisA [19] Dataset for Single-Class Anomaly Detection**

Metric →	Image-level			Pixel-level			
Category ↓	I-AUROC	I-AP	I-F1_max	P-AUROC	P-AP	P-F1_max	AUPRO
pcb1	98.6	98.5	95.0	99.6	86.7	78.5	95.1
pcb2	98.0	96.6	96.0	98.8	40.0	40.3	91.1
pcb3	99.3	99.4	97.0	99.0	28.4	38.5	93.1
pcb4	100	100	99.0	98.6	51.3	51.6	93.2
macaroni1	97.2	97.3	91.0	99.4	33.1	40.4	94.8
macaroni2	95.0	94.9	89.0	99.7	26.7	36.2	98.4
capsules	99.0	99.3	97.6	99.5	66.2	65.5	98.2
candle	98.8	98.8	94.7	99.4	46.2	50.2	95.7
cashew	98.5	99.3	97.0	93.8	59.9	60.6	89.7
chewinggum	99.4	99.7	96.9	98.8	58.1	63.5	87.1
fryum	99.3	99.7	98.0	95.8	43.5	48.8	93.1
pipe_fryum	99.2	99.6	97.5	98.5	49.8	56.5	95.6
Mean	98.5	98.6	95.7	98.4	49.2	52.6	93.8

Table S11. **Per-Class Performance of the Proposed INP-Former on the Real-IAD [16] Dataset for Single-Class Anomaly Detection**

Metric →	Image-level			Pixel-level			
Category ↓	I-AUROC	I-AP	I-F1_max	P-AUROC	P-AP	P-F1_max	AUPRO
audiojack	92.2	88.2	78.4	99.6	53.3	55.2	97.0
bottle_cap	94.9	94.1	85.0	99.7	40.2	40.4	98.4
button_battery	89.9	91.4	84.4	99.2	51.9	56.3	93.8
end_cap	89.5	89.1	85.5	99.3	23.9	34.9	97.0
eraser	93.9	91.9	83.1	99.8	48.2	50.7	98.3
fire_hood	88.4	81.7	73.3	99.6	47.2	49.6	96.5
mint	82.4	82.1	73.7	98.2	29.2	38.7	86.4
mounts	87.8	75.8	77.8	99.6	43.3	45.2	95.8
pcb	93.9	96.4	89.2	99.4	59.2	59.5	96.4
phone_battery	94.6	93.0	85.3	99.7	67.8	61.9	97.9
plastic_nut	94.4	90.3	83.2	99.8	48.3	48.2	98.5
plastic_plug	91.2	87.9	77.5	99.3	34.4	39.9	96.1
porcelain_doll	86.4	75.4	70.9	99.0	29.7	37.1	94.9
regulator	87.5	78.6	69.2	99.3	44.5	49.2	95.4
rolled_strip_base	99.5	99.7	98.3	99.8	51.7	54.7	98.9
sim_card_set	97.4	97.7	92.1	99.3	59.3	58.9	93.9
switch	98.4	98.7	94.5	99.2	68.7	65.6	97.7
tape	98.2	97.1	91.0	99.8	55.4	56.3	99.1
terminalblock	97.4	98	92.7	99.7	55.8	56.9	99.1
toothbrush	86.5	86.4	81.8	96.2	31.6	40.1	89
toy	89.3	90.9	85.6	96.9	28.1	35.9	94
toy_brick	82.5	78.9	70.8	98.1	41.5	45.6	84.7
transistor1	97.9	98.5	93.9	99.5	54.8	54.6	97.7
usb	95.5	95.0	88.8	99.5	48.6	51.5	98.2
usb_adaptor	87.1	81.7	74.0	99.3	33.6	39.7	95.1
u_block	93.8	90.9	81.2	99.6	53.3	58.1	97.3
vcpill	93.7	93.4	84.7	99.2	71.2	68.7	94.5
wooden_beads	91.6	90.6	82.3	99.3	49.5	52.9	93.7
woodstick	87.4	78.5	70.9	99.4	57	57.7	94.2
zipper	98.4	99.0	95.1	99.1	61.5	64.0	97.1
Mean	92.1	89.7	83.1	99.2	48.1	50.9	95.6

Table S12. Per-Class Results on the **MVTec-AD** [2] Dataset for **Multi-Class Anomaly Detection** with AUROC/AP/F1_max metrics.

Method → Category ↓	RD4AD [5] CVPR'22	UniAD [17] NeurIPS'22	SimpleNet [14] CVPR'23	DeSTSeg [18] CVPR'23	DiAD [9] AAAI'24	MambaAD [8] NeurIPS'24	Dinomaly [7] Arxiv'24	**INP-Former Ours**
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**	**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**	**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	**99.0/99.8/98.6**
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**	**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**	**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**	**99.1**/99.8/97.9
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	97.5/99.2/94.9
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**	**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	99.7/99.5/98.8
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**	**100/100/100**
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	**99.9/100/99.4**
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	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**	**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**	**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	**99.9/100**/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	**99.7/99.9/99.2**

Table S13. Per-Class Results on the **MVTec-AD** [2] Dataset for **Multi-Class Anomaly Localization** with AUROC/AP/F1_max/AUPRO metrics.

Method → Category ↓	RD4AD [5] CVPR'22	UniAD [17] NeurIPS'22	SimpleNet [14] CVPR'23	DeSTSeg [18] CVPR'23	DiAD [9] AAAI'24	MambaAD [8] NeurIPS'24	Dinomaly [7] Arxiv'24	**INP-Former Ours**
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	99.1/**88.7**/83.2/**97.1**
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	**98.8/79.3/75.8/94.4**
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	**98.8**/60.3/58.5/**97.7**
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**	**99.5**/81.8/76.9/**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	97.5/81.2/86.6/**95.1**
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**	97.7/76.1/70.3/**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**	99.5/**61.8**/58.6/97.9
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	**99.1**/58.3/66.6/**95.9**
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	94.7/64.0/62.4/79.0
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**	99.0/75.8/72.7/96.4
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	**99.4/72.5/72.4/97.7**
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	**99.4**/58.1/60.1/**97.7**
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	99.4/56.3/57.4/98.0
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	97.8/76.6/74.4/88.3
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	**97.6**/74.6/68.9/93.7
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	**98.5/71.0/69.7/94.9**

Table S14. Per-Class Results on the **VisA** [19] Dataset for **Multi-Class Anomaly Detection** with AUROC/AP/F1_max metrics.

Method → Category ↓	RD4AD [5] CVPR'22	UniAD [17] NeurIPS'22	SimpleNet [14] CVPR'23	DeSTSeg [18] CVPR'23	DiAD [9] AAAI'24	MambaAD [8] NeurIPS'24	Dinomaly [7] Arxiv'24	**INP-Former 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**	98.8/98.7/96.1
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**	98.8/98.6/**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	**99.2/99.2/97.0**
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	**99.9/99.9/99.0**
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**	**98.5/98.4**/93.9
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	**96.9/96.8/92.8**
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	**99.1/99.4/98.0**
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**	98.4/98.5/93.5
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**	98.6/**99.4**/96.5
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**	99.7/**99.9**/98.5
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	**99.3/99.7/98.0**
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	**99.5/99.8/98.5**
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	**98.9/99.0/96.6**

Table S15. Per-Class Results on the **VisA** [19] Dataset for **Multi-Class Anomaly Localization** with AUROC/AP/F1_max/AUPRO metrics.

Method → Category ↓	RD4AD [5] CVPR'22	UniAD [17] NeurIPS'22	SimpleNet [14] CVPR'23	DeSTSeg [18] CVPR'23	DiAD [9] AAAI'24	MambaAD [8] NeurIPS'24	Dinomaly [7] Arxiv'24	**INP-Former 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	99.6/87.6/80.1/95.2
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	98.7/31.2/40.1/**91.9**
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**	98.8/30.6/39.4/94.3
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**	**98.8/53.2/53.5**/94.2
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**	**99.6/33.9/41.1**/96.0
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**	**99.8/26.8/37.8/98.7**
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	**99.6/67.2**/66.2/**98.0**
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	**99.4/43.9/49.7/95.6**
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**	97.7/66.2/64.0/92.0
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**	98.9/59.6/64.2/86.5
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	96.8/51.2/53.6/**94.2**
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	99.3/63.3/67.2/**95.8**
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**	**98.9**/51.2/54.7/94.4

Table S16. Per-Class Results on the **Real-IAD** [16] Dataset for **Multi-Class Anomaly Detection** with AUROC/AP/F1_max metrics.

Method → Category ↓	RD4AD [5] CVPR'22	UniAD [17] NeurIPS'22	SimpleNet [14] CVPR'23	DeSTSeg [18] CVPR'23	DiAD [9] AAAI'24	MambaAD [8] NeurIPS'24	Dinomaly [7] Arxiv'24	**INP-Former 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	**88.9/84.6/74.0**
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	89.3/86.1/81.1
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	86.2/88.4/82.0
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	**87.0**/87.0/84.2
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	**92.4/90.2/81.2**
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	**86.5**/79.0/72.7
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	**77.2/76.8**/69.9
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	88.1/77.4/77.4
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	**93.9/96.3/89.1**
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	**93.7/92.1/83.7**
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	**91.2/85.3/78.1**
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	**90.9**/87.9/78.9
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	**88.5**/80.9/72.9
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	83.8/75.6/64.9
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	**99.3/99.6/97.2**
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	**96.6/97.0/90.4**
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	**98.0/98.4/93.8**
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	**97.4**/96.1/**89.7**
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	**96.9/97.4/91.7**
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	89.9/91.4/83.2
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	**88.0/90.9/83.9**
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	75.2/69.9/64.6
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	**97.8/98.4/93.8**
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	**91.9/87.5/77.8**
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	**94.4/93.6/86.8**
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	**85.2/78.4/73.1**
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	**92.8/92.2/83.1**
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	**89.8/88.9/80.2**
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	**85.4/75.0/68.0**
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	99.1/99.5/96.3
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	**90.5/88.1/81.5**

Table S17. Per-Class Results on the **Real-IAD** [16] Dataset for **Multi-Class Anomaly Localization** with AUROC/AP/F1_max/AUPRO metrics.

Method → Category ↓	RD4AD [5] CVPR'22	UniAD [17] NeurIPS'22	SimpleNet [14] CVPR'23	DeSTSeg [18] CVPR'23	DiAD [9] AAAI'24	MambaAD [8] NeurIPS'24	Dinomaly [7] Arxiv'24	**INP-Former 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	**99.2/54.6/56.5/95.0**
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**	**99.7/34.2/39.1**/97.8
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**	99.0/39.5/55.8/92.8
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	**99.2**/25.8/32.6/**96.6**
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	**99.7**/47.4/48.2/**97.6**
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	**99.4/44.1/46.6/95.4**
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	**97.2/27.6/37.9/81.1**
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	**99.5**/39.7/43.5/**96.7**
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	**99.5/60.4/59.9/96.7**
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	**99.7/66.0/60.3/97.3**
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	**99.8**/44.3/**45.8/98.4**
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	**99.4/33.6/39.0/96.7**
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	**99.4/37.2/42.3/96.9**
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	**99.3/45.3/51.4/95.7**
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	**99.8**/48.3/**52.9/98.8**
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	**99.3**/60.6/58.5/**94.2**
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	97.5/**63.5**/62.3/**96.3**
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	**99.8**/58.4/**58.1/98.9**
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	**99.8/54.0/53.9/99.0**
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	96.9/**39.7/44.6**/90.8
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	95.3/**26.4/35.3/92.1**
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	97.3/**37.0/41.2**/80.0
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**	**99.6/57.7/55.6/97.8**
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	**99.6**/50.9/53.8/**97.6**
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	**99.4/48.7/50.5/98.1**
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	**99.3/29.9/36.1/94.4**
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	**99.2/71.7/69.0/94.6**
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	**99.2/52.3/53.6/92.4**
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	**99.2**/55.1/54.9/**92.4**
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**	**99.4/71.6/69.2**/97.6
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	**99.0/47.5/50.3/95.0**

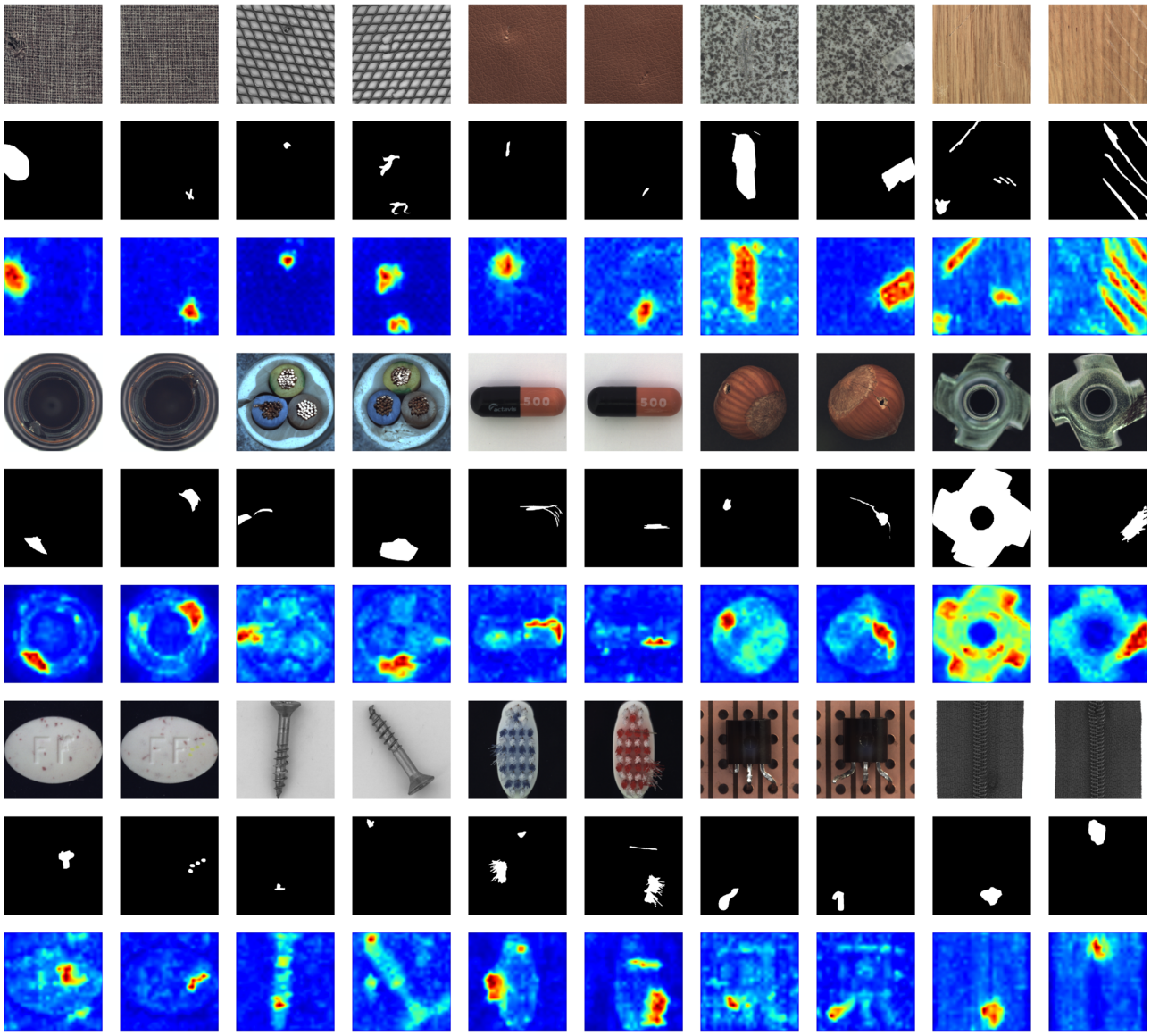

Figure S3. **Anomaly localization results on the MVTec-AD** [2] **dataset under the multi-class anomaly detection setting**. For each tuple, the images from top to bottom represent the anomaly image, ground truth, and predicted anomaly map.

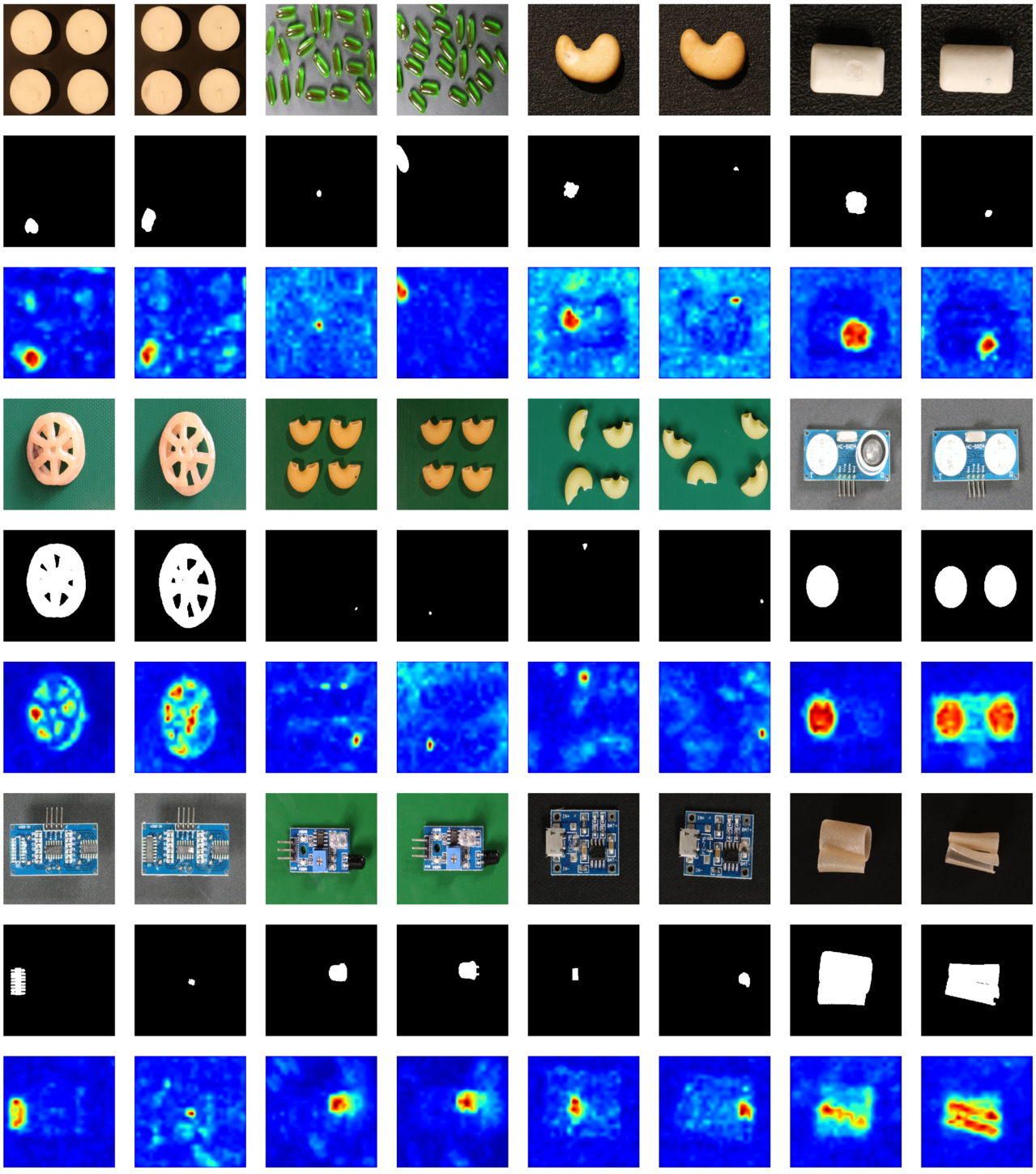

Figure S4. **Anomaly localization results on the VisA** [19] **dataset under the multi-class anomaly detection setting**. For each tuple, the images from top to bottom represent the anomaly image, ground truth, and predicted anomaly map.

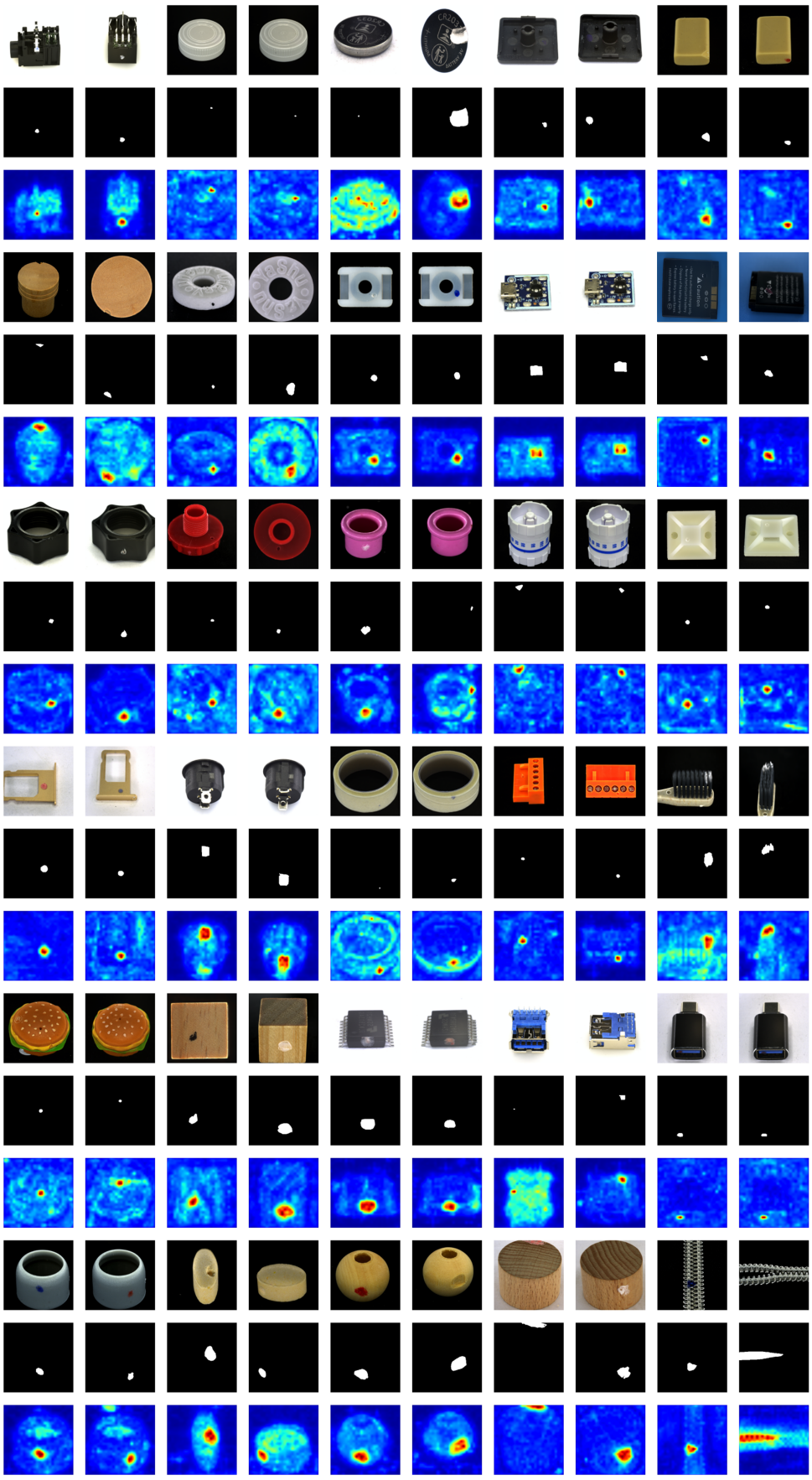

Figure S5. **Anomaly localization results on the Real-IAD [16] dataset under the multi-class anomaly detection setting**. For each tuple, the images from top to bottom represent the anomaly image, ground truth, and predicted anomaly map.

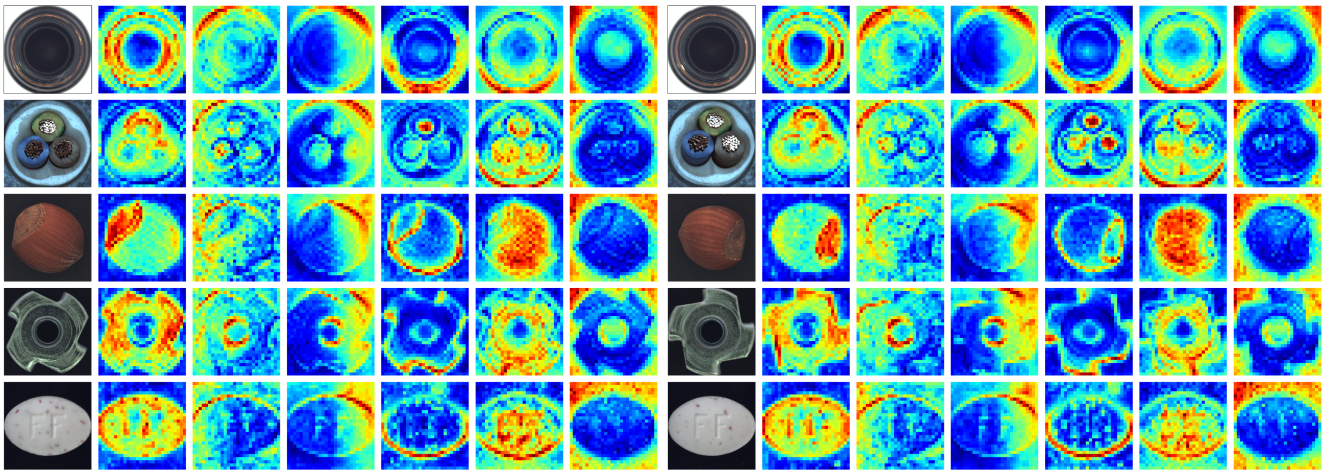

Figure S6. **Cross-attention maps** between INPs and image patches.