PTAD: Pose and Texture Agnostic Anomaly Detection

Supplementary Material

Table 1. Comparison of model complexity between *Looking 3D* and our method. We show the number of trainable parameters for each module as well as the entire network.

Methods	Modules	Trainable Parameters	Total
Looking 3D	ResNet18-FPN	12.77M	30.77M
	VLFA	2.49M	
	CGA	15.5M	
Ours	Pose Alignment	-	**17.62M**
	MMPDN	**17.62M**	

1. Comparison of Parameter Quantities

To comprehensively evaluate the efficiency of our render-for-detect framework, we compare the trainable parameters of our framework with those of the Correspondence Matching Transformer (CMT) from *Looking 3D* [1]. In Tab. 1, our pose alignment module requires only **seven** learnable parameters per 3D model, which is negligible. Meanwhile, the Multi-Modal Pyramid Detection Network (MMPDN) introduces 17.62M additional parameters, including those from ResNet18 [2]. Overall, our framework contains only half the number of parameters compared to *Looking 3D* [1]. Owing to this lightweight design, our model exhibits weaker dependence on large-scale training data, enabling it to maintain stable performance even under data-scarce conditions. In practical applications, our method not only lowers the cost of data acquisition and simplifies deployment but also offers stronger generalization capability.

Table 2. Performance comparison of different methods. Our full method achieves the highest AUC and Accuracy, while the second row corresponds to the unimodal method using *Depth Anything*.

Methods	AUC (%) (↑)	Accuracy (%) (↑)
Looking 3D	84.7	75.4
Ours (with Depth Anything)	87.4	78.8
Ours	**91.0**	**83.0**

2. Anomaly Detection

2.1. Limitations of the Unimodal Method

To mitigate the modality gap, we explore *Depth Anything* [3] to align the query and reference images into a unified modality. Specifically, we use *Depth Anything* to predict the depth map of the query image, while the reference depth map is obtained from our pose-aligned rendering. We apply a Sobel operator to both the query and reference depth maps, converting them into edge representations to address the scale discrepancy between the rendered depth map and the predicted depth map. As reported in Tab. 2, this strategy achieves an accuracy that is 3.4% higher than *Looking 3D* [1] under identical training settings. However, it still lags behind our proposed method by 4.2% in accuracy.

We attribute this gap mainly to two factors. First, as shown in Fig. 1, in predicted depth map, foreground and background are fused, leading to indistinct boundaries and even producing pseudo anomalies caused by missing components. Second, depth estimation from a single view lacks multi-view consistency and cannot faithfully recover the underlying 3D structure, which limits the model's sensitivity to subtle geometric anomalies.

Table 3. Performance of the unimodal method using *Depth Anything* under varying training sampling ratios.

Sampling Ratio	AUC (%) (↑)	Accuracy (%) (↑)
5%	75.2	68.7
10%	78.6	71.1
20%	81.2	73.5
50%	84.9	76.2
80%	86.8	78.2
100%	87.4	78.8

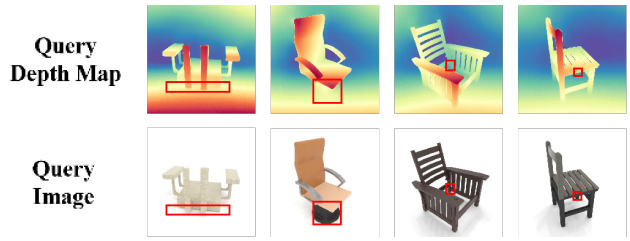

Figure 1. **Visualization of Predicted Depth Maps for Query Images.** In the first two columns, parts of the foreground and background become fused, leading to blurred or missing boundaries. In the last two columns, discontinuities in the predicted depth produce missing object parts, resulting in pseudo anomalies. The problematic regions are highlighted in red boxes.

2.2. Unimodal Performance under Data Sparsity

To further assess the robustness of the unimodal method using *Depth Anything* [3] under limited training data, we conduct experiments across different sampling ratios.

As shown in Tab. 3, when more than 50% of the training data is available, the unimodal method outperforms Looking 3D. However, once the sampling ratio drops below 20%, both AUC and accuracy degrade sharply. The main reason is that the unimodal method using *Depth Anything* is highly sensitive to appearance variations, making it diffi-

cult to suppress reconstruction errors under sparse training data. With insufficient samples, the network struggles to distinguish prediction noise from true geometric anomalies, leading to rapid performance degradation.

References

[1] Ankan Bhunia, Changjian Li, and Hakan Bilen. Looking 3d: Anomaly detection with 2d-3d alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17263–17272, 2024. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[3] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37: 21875–21911, 2024. 1