A APPENDIX

Algorithm 1 provides pseudocode for the full **PIRN** pipeline.

Algorithm 1 Pseudocode of the proposed **PIRN** framework.

Input: RGB image and surface-normal map
Output: Anomaly map (heatmap)
Extract multi-scale features E_{rgb} and E_{sn} using frozen ViT encoders
Initialize patch tokens: $Z_{rgb} \leftarrow E_{rgb}, \quad Z_{sn} \leftarrow E_{sn}$.
Randomly initialize prototypes P_{rgb} and P_{sn}.
for each decoder layer $\ell = 1$ to L **do**
  Adaptive Prototype Refinement (APR)
  Compute Γ^*_{rgb} and Γ^*_{sn} by solving the balanced OT in Eq. (1) for (Z_{rgb}, P_{rgb}) and (Z_{sn}, P_{sn})
  Compute context vectors c_k^{rgb} and c_k^{sn} for each prototype by column-normalized OT-weighted averaging of patch tokens
  Update each prototype p_k^{rgb} and p_k^{sn} using $\mathrm{GRU}(p_k, c_k)$
  Balanced Prototype Assignment (BPA)
  Compute T^*_{rgb} and T^*_{sn} by solving Eq. (1) for (Z_{rgb}, P_{rgb}) and (Z_{sn}, P_{sn})
  Reconstruct patch tokens: $Z^{bpa}_{rgb} = T^*_{rgb} P_{rgb}, \quad Z^{bpa}_{sn} = T^*_{sn} P_{sn}$ (Eq. (2))
  MNC Stage 1: 2D and 3D Prototype Alignment
  Align prototypes via cross-modal graph attention to obtain refined P'_{rgb} and P'_{sn}
  MNC Stage 2: Cross-Modal Feature Injection
  Purify patch tokens: $Z'_{rgb} = Z_{rgb} \cdot \sigma(Z^{bpa}_{rgb}), \quad Z'_{sn} = Z_{sn} \cdot \sigma(Z^{bpa}_{sn})$
  Compute cross-attention outputs: $\mathrm{CA}(Z'_{rgb}, P'_{sn})$ and $\mathrm{CA}(Z'_{sn}, P'_{rgb})$ (Eq. (3))
  Apply gating to obtain Z^{mnc}_{rgb} and Z^{mnc}_{sn} (Eq. (4))
  Fuse reconstructions: $Z^{rec}_{rgb} = Z^{bpa}_{rgb} + Z^{mnc}_{rgb}, \quad Z^{rec}_{sn} = Z^{bpa}_{sn} + Z^{mnc}_{sn}$
  Update $Z_{rgb} \leftarrow Z^{rec}_{rgb}, \quad Z_{sn} \leftarrow Z^{rec}_{sn}$
end for
Compute per-modality patch anomaly scores: $d_i^{(rgb)} = 1 - \cos(E_i^{(rgb)}, Z^{rec}_{rgb,i})$ and $d_i^{(sn)} = 1 - \cos(E_i^{(sn)}, Z^{rec}_{sn,i})$
Fuse modalities to obtain the final anomaly map: $d_i = d_i^{(rgb)} + d_i^{(sn)}$

B MORE IMPLEMENTATION DETAILS

B.1 DETAILS OF GATED PROTOTYPE UPDATE VIA GRU.

To ensure that prototypes coherently represent normal features, we update each prototype by incorporating this context vector $\{c_k\}$ through a gated recurrent unit (GRU) update. We treat the original prototype p_k as the hidden state and its context c_k as the input to a GRU cell, producing an updated prototype p'_k.

The GRU's gating mechanism dynamically controls the extent to which each prototype is updated, ensuring that only normal context is integrated while minimizing the incorporation of anomalous information during testing. Formally, the update for prototype p_k is given by:

$$u_k = \sigma\Big(W_z\,[p_k;\ c_k] + b_z\Big), \tag{5}$$

$$r_k = \sigma\Big(W_r\,[p_k;\ c_k] + b_r\Big), \tag{6}$$

$$\tilde{p}_k = \tanh\Big(W\,[r_k \odot p_k;\ c_k] + b\Big), \tag{7}$$

$$p'_k = u_k \odot p_k \ + \ (1 - u_k) \odot \tilde{p}_k\ , \tag{8}$$

where $[\cdot;\ \cdot]$ denotes vector concatenation, $\odot$ is element-wise multiplication, $\sigma(\cdot)$ is the sigmoid activation, and W_z, W_r, W (with corresponding biases b_z, b_r, b) are learnable weights. Eqs. (5)–(8)

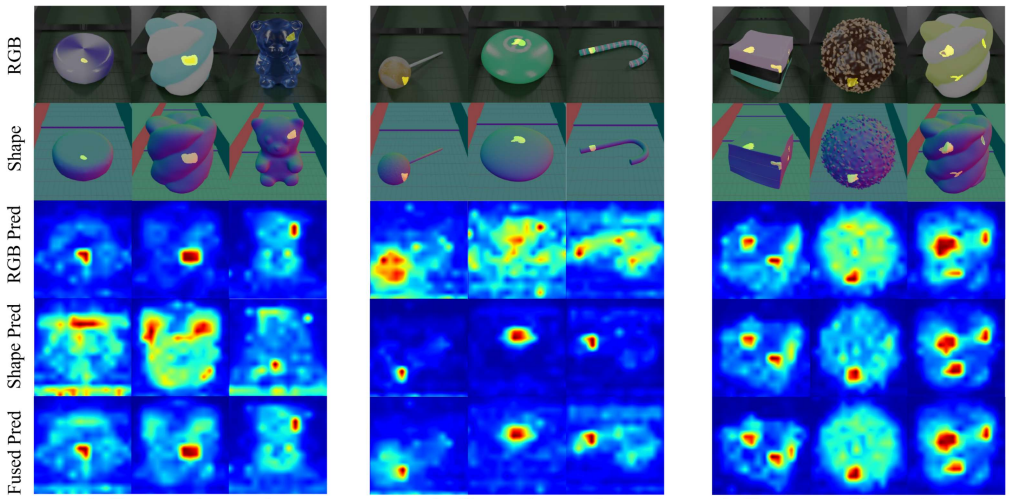

Figure 5: Qualitative results on Eyecandies with various types of anomalies, showing the complementary roles of the 2D and 3D branch in **PIRN**.

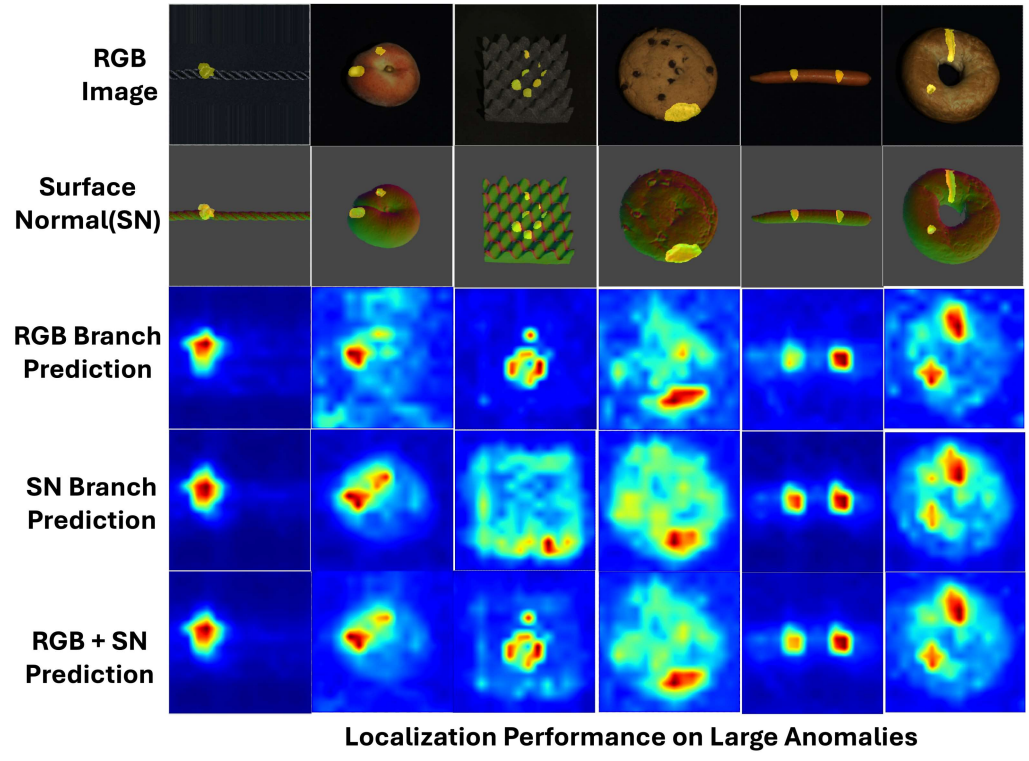

Figure 6: Localization performance on test samples with large or pervasive anomalies (10-shot normal training). PIRN accurately highlights extensive defects, demonstrating the APR module's robustness against prototype corruption even when anomalies dominate the input.

are the standard GRU equations adapted to our prototype refinement setting: u_k is the update gate that decides how much of the previous prototype p_k to keep, r_k is the reset gate that modulates the influence of the past prototype when computing a candidate update, and $\tilde{p}_k$ is the candidate new

Method	AUROC$_I$	AUROC$_P$	AUPRO
Softmax Attention	0.832	0.967	0.929
Linear Attention	0.845	0.968	0.931
Sigmoid Attention	0.878	0.976	0.954
Balanced Optimal Transport	0.922	0.991	0.966

Table 9: Ablation of prototype assignment in BPA on **MVTec-3D-AD**.

Backbone	AUROC$_I$	AUROC$_P$	AUPRO
DINOv1 ViT-B/8	0.892	0.974	0.946
DINOv2 ViT-B/14	0.923	0.993	0.968
DINOv2 ViT-L/14	0.928	0.994	0.970

Table 10: Comparison of anomaly detection and localization performance on **MVTec-3D-AD** under different backbones (10 shots).

prototype state. The final refined prototype p'_k is a convex combination of the old prototype and the candidate state, weighted by the update gate.

This GRU-based update mechanism is crucial for maintaining robustness during prototype refinement. If the context c_k aligns well with the original prototype, the GRU will produce a small u_k, allowing the new information $\tilde{p}_k$ to significantly override the old state p_k. Otherwise, if the context c_k is unreliable due to an anomalous region that does not match any learned prototype, the update gate u_k will be high (near 1) to keep the prototype unchanged. In this way, the GRU acts as a learnable gate: it adaptively suppresses anomalous information and only injects context when it is deemed normal.

B.2 Generating surface normal images from Point Cloud.

We follow FIND's Li et al. (2025) procedure to generate surface normal maps. The MVTec 3D-AD dataset provides per-pixel 3D point cloud data for each sample. Using these organized point clouds, we compute a corresponding surface normal map. According to FIND Li et al. (2025), background pixels in point cloud are removed following the standard procedure in M3DM (Wang et al., 2023). For the foreground pixels, we use Open3D to compute normals by fitting a local plane to each point's neighborhood (KD-tree search with nearest neighbors $k = 30$). This produces an initial normal vector. We further enforce directional coherence using Open3D's `orient_normals_consistent_tangent_plane` with a connectivity setting of 50. After surface normal estimation, we project the computed normals back onto the image grid to form a normal map. For model training, we convert the normal vectors into a color image by linearly mapping each normal component to the $[0, 255]$ range. The resulting normal maps are saved as PNG images.

B.3 Hardware for Training.

All experiments were conducted on a workstation running Ubuntu 20.04 LTS. The system was equipped with two NVIDIA RTX 4090 GPUs with 128 GB of RAM. Our implementation was in Python 3.9 using PyTorch 1.13 (with CUDA 11.6 and cuDNN 8.1 for GPU acceleration). We also utilized OpenCV 4.8 for image processing and Open3D 0.17 for 3D data handling.

C Ablation Studies

C.1 Additional Ablations on BPA and Backbone

We provide additional ablations analyzing (i) prototype assignment strategies in BPA and (ii) backbone architectures for the frozen encoders on **MVTec-3D-AD**. As shown in Tab. 9, softmax and linear attention yield the weakest results (AUROC$_I < 85\%$), suggesting that enforcing balanced prototype usage is crucial for stable reconstruction under limited data. As PIRN uses frozen encoders, feature quality impacts performance (Tab. 10). On 10-shot MVTec 3D-AD, DINOv2 (ViT-B/14) outperforms DINOv1 (ViT-B/8), improving AUROC$_I$ from 0.892 to 0.923 (+3.1%) due to richer semantic representations. Scaling to ViT-L/14 yields marginal gains (0.928). Crucially, PIRN remains robustly competitive even with the sub-optimal DINOv1.

C.2 Robustness to Large-Scale Anomalies.

Fig. 6 shows qualitative results on test samples where anomalies occupy a large portion of the object. PIRN remains robust because APR performs a constrained, single-step refinement around the learned

Table 11: Comparisons of per-category anomaly detection performance on MVTec-3D-AD.

Method	Bagel	Cable Gland	Carrot	Cookie	Dowel	Foam	Peach	Potato	Rope	Tire	Mean
AUROC$_I$											
BTF (Horwitz & Hoshen, 2023)	0.938	0.765	0.972	0.888	0.960	0.664	0.904	0.929	0.982	0.726	0.865
AST (Rudolph et al., 2023)	0.983	0.873	0.976	0.971	0.932	0.885	0.974	**0.981**	**1.000**	0.797	0.937
M3DM (Wang et al., 2023)	0.994	0.909	0.972	0.976	0.960	0.942	0.973	0.899	0.972	0.850	0.945
CFM (Costanzino et al., 2024)	0.994	0.888	**0.984**	**0.993**	**0.980**	0.888	0.941	0.943	0.980	**0.953**	0.954
3D-ADNAS (Long et al., 2025)	**0.997**	**1.000**	0.971	0.986	0.966	0.948	0.897	0.873	**1.000**	0.867	0.951
Shape Guided (Chu et al., 2023)	0.986	0.894	0.983	0.991	0.976	0.857	0.990	0.965	0.990	0.869	0.947
PIRN	0.971	0.973	0.941	0.957	0.975	**0.993**	**0.992**	0.950	0.996	0.880	**0.963**
AUPRO@30%											
BTF (Horwitz & Hoshen, 2023)	0.976	0.969	0.979	**0.973**	0.933	0.888	0.896	0.912	0.950	0.971	0.959
AST (Rudolph et al., 2023)	0.970	0.947	0.981	0.939	0.913	0.906	0.979	0.982	0.889	0.940	0.944
M3DM (Wang et al., 2023)	0.970	0.971	0.979	0.950	0.941	0.932	0.977	0.971	0.971	0.975	0.964
CFM (Costanzino et al., 2024)	0.979	0.972	0.982	0.945	0.950	0.968	0.980	0.943	0.950	**0.981**	0.971
Shape Guided (Chu et al., 2023)	**0.981**	0.973	0.982	0.971	0.962	**0.978**	**0.981**	**0.983**	**0.974**	0.975	**0.976**
PIRN	0.966	**0.978**	**0.983**	0.972	**0.976**	0.971	**0.981**	0.978	**0.974**	0.951	0.973

normal codebook rather than unconstrained test-time re-learning. First, APR extracts prototype contexts via entropy-regularized optimal transport, which assigns negligible mass to patches that are dissimilar to all normal prototypes, thereby limiting the influence of anomalous regions. Second, the GRU gate (trained only on normal data) tends to close when the context is unreliable, preventing prototype corruption.

C.3 Per-category Results on MVTec-3D-AD.

PIRN demonstrates strong detection performance on the MVTec-3D-AD dataset (Table 11). It achieves a mean AUROC$_I$ of **0.963**, surpassing the strongest baseline CFM (0.954) and the previous state-of-the-art 3D-ADNAS (0.951). It also outperforms other representative baselines such as M3DM (0.945) and AST (0.937). These results demonstrate that PIRN not only improves average detection accuracy but also localizes diverse texture and shape defects in complex anomaly detection scenarios.

D Visualization of Anomaly Localization.

Fig. 7 and Fig. 8 illustrate PIRN's anomaly localization results on the Eyecandies and MVTec 3D-AD datasets, respectively. These qualitative results highlight PIRN's robustness in capturing fine-grained defect details while avoiding false positives.

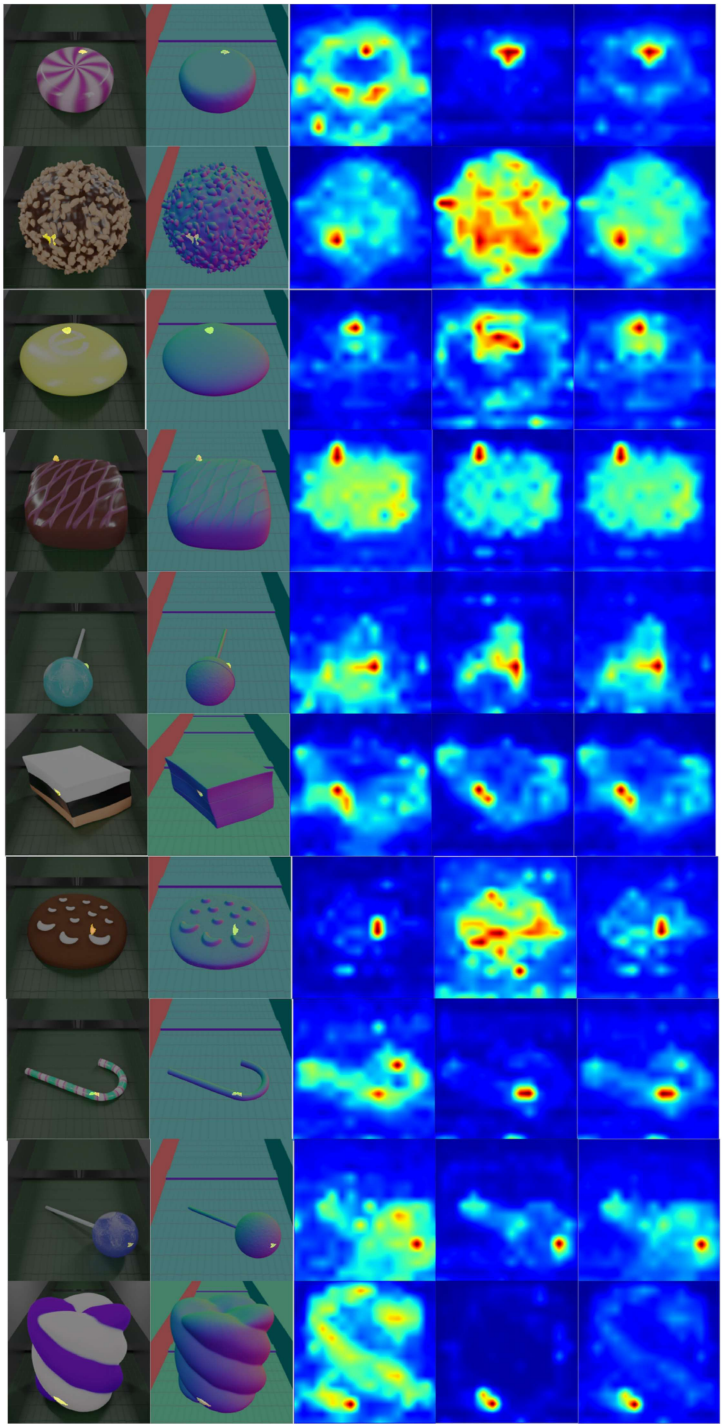

Figure 7: Visualization of localization performance on the Eyecandies dataset (5-shot normal training). From left to right: RGB images, surface normals, anomaly maps predicted by the RGB branch, anomaly maps predicted by the surface-normal branch, the fused anomaly map.

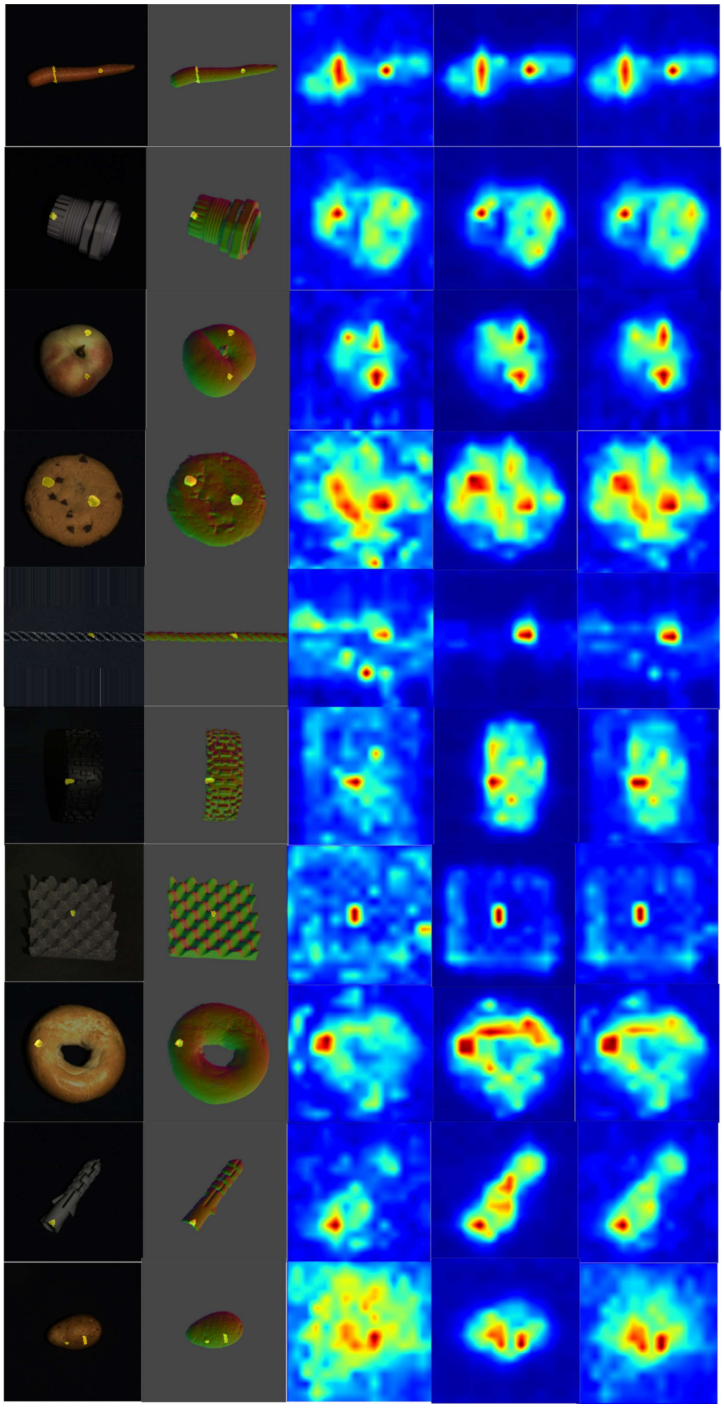

Figure 8: Visualization of localization performance on the MVTec 3D-AD dataset (5-shot normal training). From left to right: RGB images, surface normals, anomaly maps predicted by the RGB branch, anomaly maps predicted by the surface-normal branch, the fused anomaly map.