Appendix

A Datasets

MVTec 3D-AD MVTec 3D-AD is a comprehensive real-world dataset designed for unsupervised 3D anomaly detection, inspired by industrial inspection scenarios. It comprises 10 distinct object categories with a total of 4,147 high-resolution 3D scans. A key feature of this dataset is the provision of both 3D point clouds and their corresponding 2D RGB color images for each sample.

Eyecandies Eyecandies is a novel synthetic dataset focusing on unsupervised multimodal anomaly detection. It features 10 categories of "candy" objects generated programmatically, characterized by photo-realistic appearances and significant intra-class variations. As anomalies are directly injected during the 3D rendering pipeline, the pixel-level annotations are automatically generated and perfectly accurate, thus avoiding biases from manual labeling.

Real3D-AD Real3D-AD is a real-world point cloud anomaly detection dataset that emphasizes high-precision and high-resolution industrial scenarios. It contains 1,254 high-resolution 3D objects across 12 categories, aiming to address the limitations of existing datasets whose point cloud precision and integrity may not meet the demands of precision manufacturing.

Anomaly-ShapeNet Anomaly-ShapeNet is a large-scale synthetic 3D anomaly dataset created to tackle the scarcity and lack of diversity in real-world 3D anomaly data, thereby supporting research on scalable 3D anomaly detection models. Built upon the well-known large-scale 3D model repository ShapeNet, this dataset includes 1,600 point cloud samples across 40 categories, offering far greater class diversity than other datasets.

B Experimental Settings

Evaluation Metrics At the object level, we use the Area Under the Receiver Operating Characteristic (AUROC) and Average Precision (AP). At the point level, we use AUROC and Per-Region Overlap (PRO) for evaluation.

Implementation Details We adopt the "one-vs-rest" setting for fair comparison with prior methods, where we train on one class from a dataset and test on the remaining classes. Since the aforementioned datasets are designed for unsupervised anomaly detection and their training splits contain only normal samples, we use the test set of the training class as auxiliary data to obtain anomaly signals. We cycle through three designated classes for each dataset and report the average of the evaluation results to ensure a fair and comprehensive assessment. Specifically, we use 'carrot', 'cookie', and 'dowel' for MVTec3D-AD; 'Confetto', 'LicoriceSandwich', and 'PeppermintCandy' for Eyecandies; 'seahorse', 'shell', and 'starfish' for Real3D-AD; and 'ashtray0', 'bag0', 'bottle0' for Anomaly-ShapeNet. In the first stage of training, we only use rendered images instead of depth maps due to the domain gap between depth maps and the pretrained CLIP. We align the global and local visual features from the rendered images, processed by the frozen visual encoder, with the text prompts. In this stage, the learning rate is set to 0.001 for 15 epochs. In the second stage, with the learned PointNet++, APAM, and learnable prompts from Stage 1 frozen, the learning rate is set to 0.0005 for 10 epochs. The batch size is 4 for all experiments, and the temperature coefficient is set to 0.07.

Baselines As ZS3DAD is an emerging field with few existing works, we include methods from adjacent fields and adapt them to the ZS3DAD task for a thorough comparison.

- **CLIP:** Since CLIP is trained on natural images, it is inherently adapted to rendered images. We use rendered images generated with the same method and angles as in GS-CLIP as input. Similar to AA-CLIP, we integrate anomaly semantics into CLIP using two text prompt templates: "A photo of a normal [cls]" and "A photo of an anomalous [cls]", where [cls] denotes the target class name.

- **AA-CLIP (CVPR'25):** A state-of-the-art method for zero-shot 2D anomaly detection. It enhances CLIP's anomaly discrimination ability in both text and vision spaces while preserving its generalization capability. The method employs a two-stage training process, first creating anomaly-aware text anchors to clearly distinguish normal and anomalous semantics, and then aligning patch-level visual features with these anchors for precise anomaly localization. We use the same rendered images as input.
- **3DzAL (WACV'25):** A novel patch-level contrastive learning framework based on pseudo-anomalies generated from task-agnostic 3D data to learn more representative feature representations. It follows a more advanced zero-shot paradigm that does not use the "one-vs-rest" setting, instead synthesizing pseudo-anomalies directly from normal samples for training.
- **MVP-PCLIP (arXiv'24):** A CLIP-based ZS3DAD method using multi-view projection. It utilizes depth maps and fine-tunes CLIP by integrating learnable visual and adaptive text prompts. For a fair comparison, we re-trained the official open-source code under our "one-vs-rest" setting.
- **PointAD (NeurIPS'24):** A CLIP-based ZS3DAD method using multi-view projection. It employs rendered images and optimizes learnable text prompts from both 3D and 2D via hybrid representation learning. It also proposes a plug-and-play method to integrate RGB information to further enhance anomaly detection performance.

C Ablation on Hyperparameter

To further validate the robustness of our proposed GS-CLIP framework and to analyze the sensitivity of its key hyperparameters, we conducted a series of supplementary ablation studies on the MVTec3D-AD dataset.

C.1 Rank r of Depth-LoRA

We studied the effect of Depth-LoRA rank r used for fine-tuning depth map branches, and the results are shown in Table 1. We observed a significant improvement in performance when r increased from 2 to 8, with P-PRO increasing from 84.9 to 86.4. This suggests that lower ranks may not be sufficient to capture enough adaptive information to bridge domain differences. However, as r further increased from 8 to 16, although the number of trainable parameters doubled from 4.8M to 9.6M, the performance improvement was already very small (P-PRO only increased by 0.1) and fully saturated at r = 32. Therefore, choosing r = 8 is the best sweet spot between model performance and parameter efficiency, which also reflects the superiority of LoRA technology in achieving efficient domain adaptation with only a small number of parameters.

Table 1: Ablation on the rank (r) of Depth-LoRA.

r	**Params (M)**↓	**O-AUROC**	**O-AP**	**P-AUROC**	**P-PRO**
2	**1.2**	82.5	94.8	95.5	84.9
4	2.4	83.1	95.9	95.9	85.8
8	4.8	83.6	96.5	96.3	86.4
16	9.6	83.7	96.3	96.4	86.5
32	19.2	**83.7**	**96.8**	**96.6**	**86.9**

C.2 Learnable Prompt Length

The Table 2 shows the impact of the general-purpose learnable prompt length q. A shorter prompt lengthl imits the model's capacity to learn general, class-agnostic anomaly knowledge. Performance is optimal at a length of 12. Further increases in length do not yield additional improvements, indicating that $q = 12$ provides sufficient contextual capacity for the model.

Table 2: Ablation on the learnable prompt length (q).

q	O-AUROC	O-AP	P-AUROC	P-PRO
4	82.5	95.1	95.4	85.0
8	83.0	95.8	95.9	85.7
12	**83.6**	**96.5**	**96.3**	**86.4**
16	83.1	96.4	96.0	86.2
20	82.5	96.0	95.6	85.8

C.3 Ablation on Weight α of Cross-view Consistency Loss

As shown in Table 3, model performance consistently improves as α increases from 0 to 1.0, demonstrating the effectiveness of this regularization term. The best overall performance is achieved at $\alpha = 1.0$. We observe a slight performance degradation when α is further increased to 1.5, which suggests that an excessively large weight may harm the model's ability to localize view-specific details. Thus, we select $\alpha = 1.0$ as the optimal value.

Table 3: Ablation study on the weight α for the cross-view consistency loss L_{con} on MVTec3D-AD.

α	O-AUROC	O-AP	P-AUROC	P-PRO
0 (w/o L_{con})	82.8	95.9	95.8	85.9
0.1	83.0	96.0	96.0	86.0
0.5	83.3	96.3	96.2	86.3
1.0	**83.6**	**96.5**	**96.3**	**86.4**
1.5	83.4	96.4	96.2	86.2

D Ablation on Projection Conditions

D.1 Image Resolution

The resolution of 2D projection images is a key factor affecting model performance and efficiency. As shown in Table4, we evaluated the performance of the model at different resolutions. The results clearly indicate that as the resolution increases from 112x112 to 336x336, all accuracy indicators of the model have significantly improved, with P-PRO increasing from 80.1 to 86.4. This proves that higher resolution can provide richer detailed information for anomaly recognition. However, as the resolution continues to increase to 518x518, the marginal benefit of accuracy decreases (P-PRO only increases by 0.2), while the inference time sharply increases from 0.51 seconds to 0.85 seconds. Therefore, considering both detection accuracy and computational efficiency, we have chosen 336x336 as the default resolution for all experiments, which is the best balance between performance and cost.

Table 4: Performance and speed comparison at different 2D image resolutions.

Resolution	O-AUROC	O-AP	P-AUROC	P-PRO	Time (s)↓
112x112	78.2	89.5	91.3	80.1	**0.39**
224x224	82.5	94.8	95.1	84.7	0.44
336x336	83.6	**96.5**	96.3	86.4	0.51
518x518	**83.9**	96.3	96.5	86.6	0.85
798x798	83.0	95.7	**96.9**	**86.6**	1.52

D.2 Point Cloud Resolution

To evaluate the robustness of the model to the quality of the input point cloud, we randomly downsampled the original point cloud at different ratios. The results are shown in Table5 and

Figure1. It is worth noting that the performance of the model smoothly decreases as the point cloud becomes sparse, without a cliff like performance collapse. Even if the point cloud is downsampled to only 40% of the original quantity, the performance indicators of the model only show a slight decline and remain at a high level (for example, O-AUROC still has 82.1 and P-PRO is 84.0). This fully demonstrates the strong robustness of our method to low-density or missing point cloud data, which is crucial for processing data from low-cost sensors or collected in harsh environments.

Table 5: Performance comparison at different point cloud downsampling ratios.

Downsample Ratio	O-AUROC	O-AP	P-AUROC	P-PRO
100% (Original)	**83.6**	**96.5**	**96.3**	**86.4**
80%	83.2	96.1	95.9	85.8
60%	82.9	95.7	95.5	85.1
40%	82.1	94.8	94.7	84.0
20%	80.5	91.9	86.8	78.7

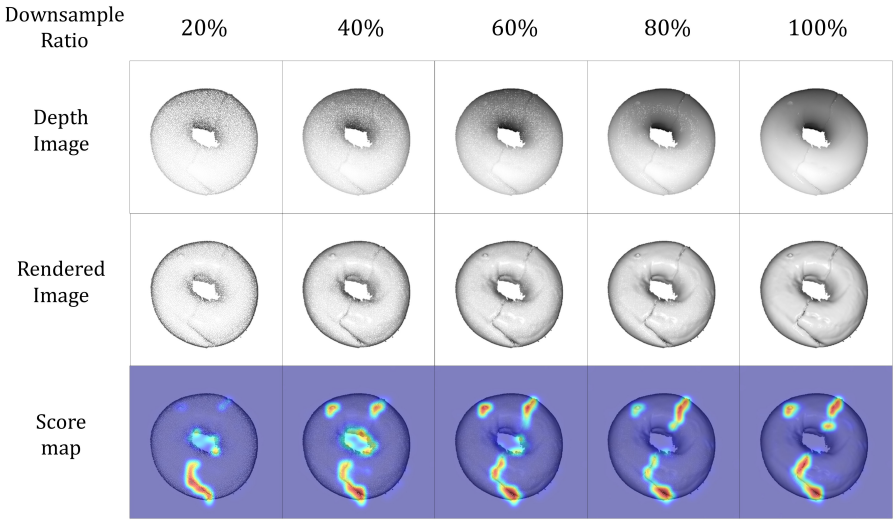

Figure 1: Qualitative results of downsampling. The top two rows show the rendered and depth images becoming sparser. The bottom row shows that the anomaly score map remains stable and accurate even at low point densities.

D.3 Point Size in Projection

When projecting point clouds onto 2D images, the size of rendered points is a key factor affecting image quality. As shown in Figure2, when the point size is too small (such as 1 or 3), there are obvious gaps in the generated image, which cannot visually form a complete object surface, which will interfere with the visual encoder's extraction of continuous surface features. On the contrary, when the point size is too large (such as 9), the rendered points will overlap with each other, causing fine geometric details and small defects to be blurred or completely covered. As shown in Table6, the performance reaches its peak at a point size of 7. This indicates that a point size of 7 achieves the best balance between ensuring object integrity and detail fidelity in the projected image, providing the highest quality input for subsequent feature extraction.

D.4 Number of Views

To investigate the impact of the number of projection views (N_v) on model performance, we conducted a series of experiments, with results detailed in Table 7. The number of views directly determines the comprehensiveness of the 3D information captured in the 2D modality.

Table 6: Performance comparison for different point sizes during rendering.

Point Size	O-AUROC	O-AP	P-AUROC	P-PRO
1	81.5	93.9	93.1	82.6
3	82.8	95.5	95.2	85.0
5	83.4	96.2	96.0	86.1
7	**83.6**	**96.5**	**96.3**	**86.4**
9	83.1	95.8	95.9	85.7

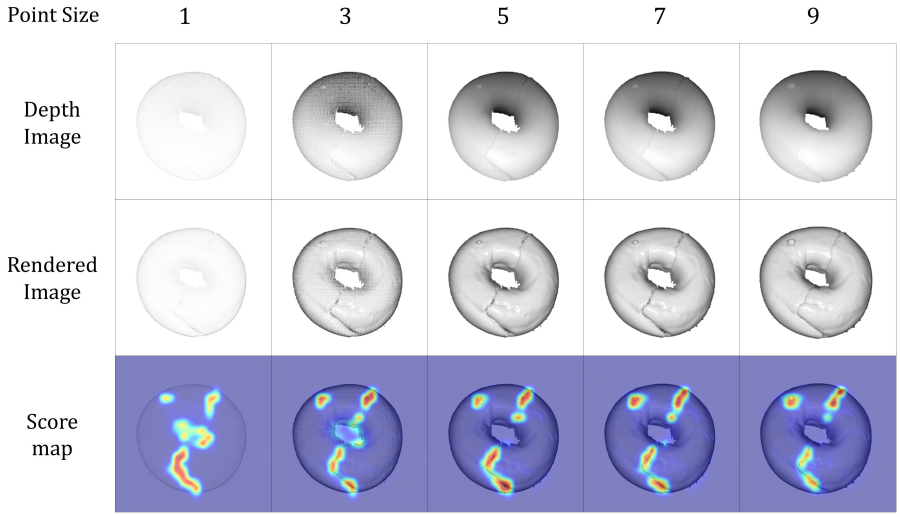

Figure 2: Qualitative results for different point sizes. A point size of 1 is too sparse, while a size of 9 causes details to be blurred. A size of 7 offers the best balance of completeness and detail preservation.

As shown in the table, there is a clear trend of performance improvement as the number of views increases from 1 to 9. For instance, the object-level AUROC dramatically rises from 73.6 to 83.6, and the crucial point-level PRO metric improves from 79.1 to 86.4. This demonstrates that incorporating more views allows the model to build a more complete and robust representation of the 3D object, effectively mitigating information loss from self-occlusion in single-view projections. However, when the number of views is further increased to 11, we observe a performance saturation and even a slight decline in some metrics (e.g., O-AP from 96.5 to 96.3). We attribute this to the introduction of redundant information from highly overlapping adjacent views, which may slightly interfere with the feature aggregation process. Therefore, we selected $N_v = 9$ as our default setting, as it provides the best trade-off between comprehensive 3D information coverage and computational efficiency.

Table 7: Performance comparison for different numbers of projection views. The rotation angles for the X-axis are listed for each setting.

Views	Rotation Angles	O- (AUROC, AP)	P- (AUROC, PRO)
1	$\{0\}$	73.6, 91.1	94.1, 79.1
3	$\{-\frac{1}{2}\pi, 0, \frac{1}{2}\pi\}$	76.8, 93.1	95.2, 83.5
5	$\{-\frac{2}{3}\pi, -\frac{1}{3}\pi, \ldots, \frac{2}{3}\pi\}$	81.8, 94.8	95.3, 85.3
7	$\{-\frac{3}{4}\pi, -\frac{2}{4}\pi, \ldots, \frac{3}{4}\pi\}$	82.3, 94.9	96.3, 85.9
9	$\{-\frac{4}{5}\pi, -\frac{3}{5}\pi, \ldots, \frac{4}{5}\pi\}$	83.6, **96.5**	**96.3**, 86.4
11	$\{-\frac{5}{6}\pi, -\frac{4}{6}\pi, \ldots, \frac{5}{6}\pi\}$	**83.7**, 96.3	96.1, **86.5**

E Other Ablation Study

E.1 Different CLIP Pre-trained Weights

We compared the performance of CLIP pre-trained models of different scales as the backbone network of our framework. As shown in Table8, the experimental results clearly indicate that larger and stronger pre-trained models can bring significant performance advantages. We default to using ViT-L/ 14@336px The model outperforms smaller ViT-B/16@224px significantly in all metrics, such as O-AUROC, increased from 80.5 to 83.6. This proves that the powerful visual linguistic prior knowledge relied upon by our method mainly comes from large-scale pre-trained models. It is worth noting that although the parameter count of larger models has increased, the entire CLIP backbone network (including image and text encoders) remained frozen throughout our training period, so this did not significantly increase training costs or the risk of overfitting.

Table 8: Performance comparison using different CLIP backbones.

CLIP Backbone	O-AUROC	O-AP	P-AUROC	P-PRO	Params (M)↓
ViT-B/16@224px	80.5	92.1	93.1	82.5	**149.62**
ViT-L/14@224px	81.9	94.0	94.8	84.3	427.62
ViT-L/14@336px	**83.6**	**96.5**	**96.3**	**86.4**	890.82

E.2 Two-Stage Training Strategy

This experiment is designed to answer a critical question: why not train all components jointly in an end-to-end fashion? We use the second stage loss L_{stage2} as the end-to-end training loss. Table 9 compares the performance of our two-stage strategy against a single-stage, end-to-end training baseline. The results show a significant performance degradation when using the end-to-end approach. At the same time, we observed that the loss function decreased very fast during the end-to-end training. Therefore, we speculate that the simultaneous fine-tuning of the visual end and the text end on the small dataset will lead to overfitting or model collapse. In contrast, our two-stage strategy first establishes a high-quality, stable optimization target by training the geometry-aware prompt generator, which then provides a clear and effective guide for the subsequent vision-language alignment in Stage 2. This experiment provides strong evidence for the necessity of our decoupled two-stage training strategy to avoid model collapse and achieve meaningful learning.

Table 9: Comparison between our two-stage training strategy and a single-stage end-to-end baseline.

Training Strategy	O-AUROC	O-AP	P-AUROC	P-PRO
End-to-End Training	51.5	63.8	64.1	43.2
Ours (Two-Stage)	**83.6**	**96.5**	**96.3**	**86.4**

F Visualization Results

We provide visualization results for multiple categories, as shown in Figure 3 and Figure 4.

G Multi-view Projection

To leverage the powerful feature extraction capabilities of CLIP's image encoder, we transform the input 3D point cloud P, into a 2D modality comprising multi-view images. We use the rendering method presented in CPMF (Cao, Xu, and Shen 2024). This following content details this process.

Given a virtual camera model parameter set C and a rendering function $\mathcal{R}_i$, the process of rendering a 3D point cloud P into a 2D image I_i can be formalized as:

$$I_i = \mathcal{R}_i(P, C), \tag{1}$$

where i is i-th view of multi-view projection.

To accurately align features extracted from 2D images back to the 3D space in subsequent steps, it is crucial to obtain the correspondence between a 3D point and its location in 2D pixel coordinates. Let P_j be the spatial coordinate of the j-th 3D point and $I_{i,j}$ be its corresponding 2D pixel coordinate in i-th view. This 2D-3D projection correspondence is given by:

$$[I_{i,j}, 1]^T = \frac{1}{Z_c} K_i T_i [P_j, 1]^T, \tag{2}$$

where K_i and T_i are the intrinsic and extrinsic matrices of the rendering camera, respectively, and Z_c is a normalization coefficient. At the same time, we record the visibility matrix H_i of the point cloud P for the i-th view. By comparing the depth of the points projected to the same pixel in the view, we set the corresponding visibility mask of the point with the smallest depth to 1, and set other points to 0.

In our implementation, the rotation matrices R_k are primarily generated by rotating around the X-axis by different angles $\theta_{x,k}$, simulating the common industrial scenario of an object tumbling on a conveyor belt. We set $v = 9$ views, with rotation angles $\theta_{x,k}$ uniformly sampled from $\{\frac{4}{5}\pi, \frac{3}{5}\pi, \ldots, -\frac{4}{5}\pi\}$.

For each rotated point cloud P_k, we simultaneously generate two types of images:

1. **Rendered Image (I_i^R):** Generated by the Filament rendering engine in 'Open3D', this image includes lighting, shadows, and material information.

2. **Depth Image (I_i^D):** Extracted from the depth buffer of the rendering pipeline, this image records the distance from each pixel's corresponding point to the camera.

Through this process, we generate a comprehensive multi-modal, multi-view 2D dataset $\{I_i^R, I_i^D\}_{i=1}^v$ for a single input point cloud, while preserving the precise 2D-3D mapping relationships $\{I_i\}_{i=1}^v$, thereby laying a foundation for subsequent feature extraction and anomaly localization.

H Detailed Results

Table 10, 11, 12, 13 provides the detailed per-class performance metrics of our method on all four datasets.[]

Table 10: Detailed results on MVTec3D-AD.

Object	P-AUROC	P-PRO	O-AUROC	O-AP
Bagel	99.7	99.0	99.8	99.9
Cable Gland	97.7	93.9	82.6	94.9
Carrot	99.6	98.4	98.2	99.6
Cookie	95.7	91.4	77.4	92.6
Dowel	95.9	76.7	72.9	91.2
Foam	94.9	80.6	88.0	97.0
Peach	99.6	98.7	96.5	99.2
Potato	99.8	98.7	95.9	98.7
Rope	98.7	90.9	96.4	98.4
Tire	98.4	90.0	75.8	93.0
Mean	**96.3**	**86.4**	**83.6**	**96.5**

References

Table 11: Detailed results on Eyecandies.

Object	P-AUROC	P-PRO	O-AUROC	O-AP
CandyCane	96.4	86.6	48.3	51.1
ChocolateCookie	97.7	90.3	90.4	93.7
ChocolatePraline	93.1	84.6	92.0	93.4
Confetto	98.4	94.7	92.6	95.3
GummyBear	92.7	78.5	80.9	82.9
HazelnutTruffle	89.2	62.1	72.3	76.8
LicoriceSandwich	94.1	71.1	83.7	87.0
Lollipop	97.0	87.5	77.9	71.5
Marshmallow	97.4	88.2	86.4	89.5
PeppermintCandy	96.0	85.7	90.4	93.6
Mean	**93.1**	**73.8**	**71.5**	**75.9**

Table 12: Detailed results on Real3D-AD.

Object	P-AUROC	O-AUROC	O-AP
Airplane	58.9	49.1	59.1
Car	78.3	80.6	81.0
Candybar	77.7	76.8	80.4
Chicken	71.7	53.4	57.1
Diamond	86.4	98.7	98.7
Duck	52.7	80.6	81.0
Fish	84.6	86.8	88.4
Gemstone	85.8	89.8	88.1
Seahorse	75.6	76.4	81.0
Shell	80.3	92.0	92.7
Starfish	83.6	89.1	92.5
Toffees	76.5	80.1	84.2
Mean	**76.3**	**76.4**	**77.7**

Table 13: Detailed results on Anomaly-ShapeNet.

Object	P-AUROC	O-AUROC	O-AP
ashtray0	79.5	97.1	97.1
bag0	74.1	89.0	85.3
bottle0	85.9	100.0	100.0
bottle1	80.6	89.8	90.7
bottle3	74.7	93.3	96.0
bowl0	82.4	85.2	87.4
bowl1	75.9	83.0	82.4
bowl2	74.9	73.0	80.3
bowl3	76.3	79.6	81.0
bowl4	71.9	75.2	77.9
bowl5	78.8	89.5	93.1
bucket0	83.1	98.4	99.0
bucket1	80.9	84.4	89.3
cap0	81.2	93.3	94.4
cap3	70.5	86.7	91.8
cap4	64.5	69.1	79.3
cap5	76.5	79.3	78.0
cup0	82.9	98.1	98.4
cup1	74.9	88.1	91.2
eraser0	81.3	100.0	100.0
headset0	81.3	98.7	98.9
headset1	75.0	89.5	91.5
helmet0	74.9	78.6	85.5
helmet1	70.1	75.7	72.8
helmet2	68.6	64.1	67.5
helmet3	78.1	83.3	90.0
jar0	76.7	88.1	88.6
microphone0	89.5	100.0	100.0
shelf0	76.1	72.5	84.0
tap0	84.6	76.1	86.6
tap1	60.6	60.7	66.9
vase0	67.1	82.1	85.5
vase1	77.3	78.1	83.5
vase2	69.6	86.2	82.3
vase3	81.3	80.9	87.2
vase4	71.6	77.6	86.3
vase5	71.4	76.7	79.6
vase7	80.0	95.2	95.6
vase8	78.1	82.7	91.2
vase9	57.3	62.4	70.4
Mean	**75.2**	**84.1**	**86.8**

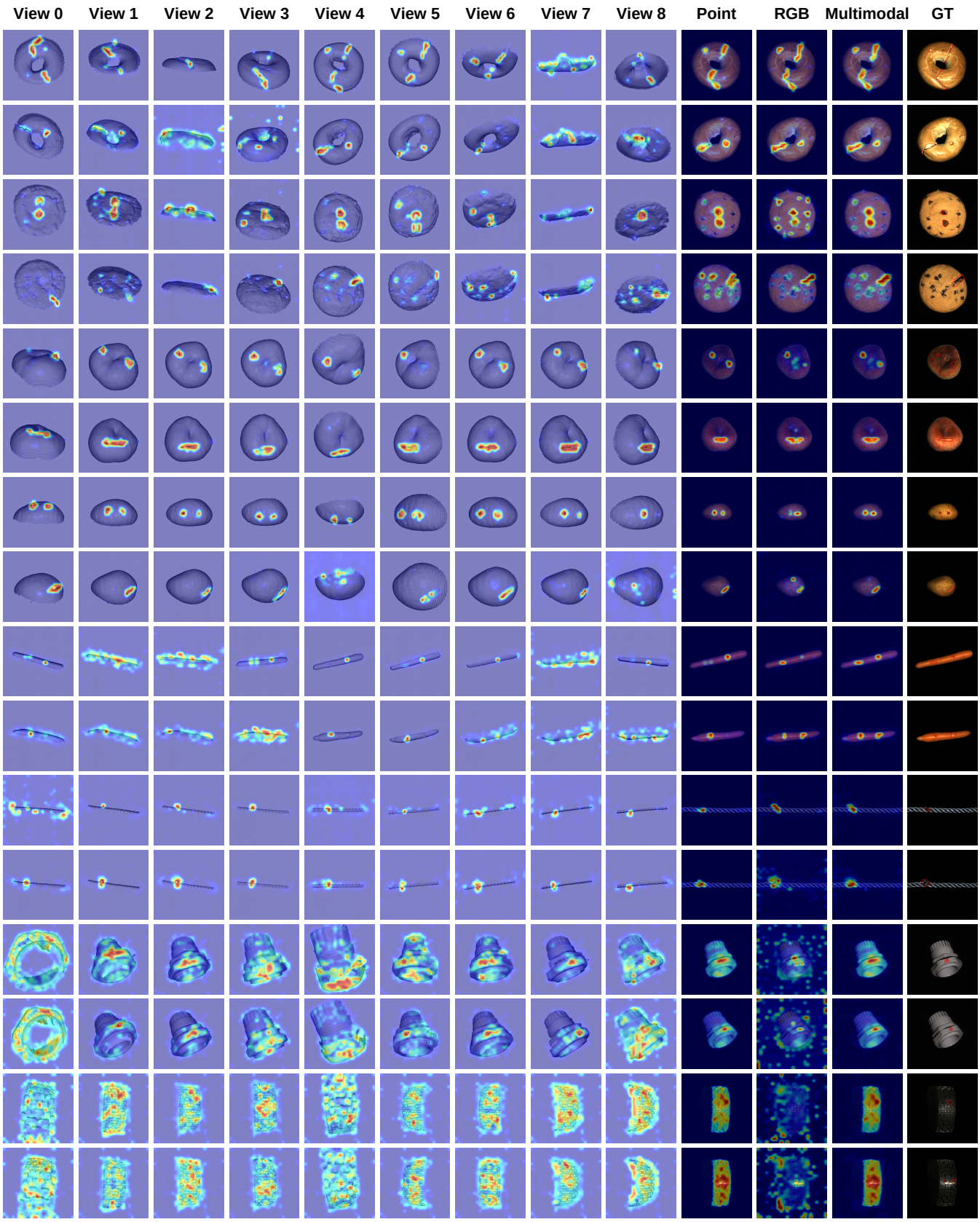

Figure 3: Visualization of anomaly score maps in MvTec3D-AD dataset. The point's anomaly score map can be obtained by integrating images from 9 views. When color RGB images are available, the score map from the point cloud can be further combined with the score map from the RGB image to produce a more accurate multimodal score map.

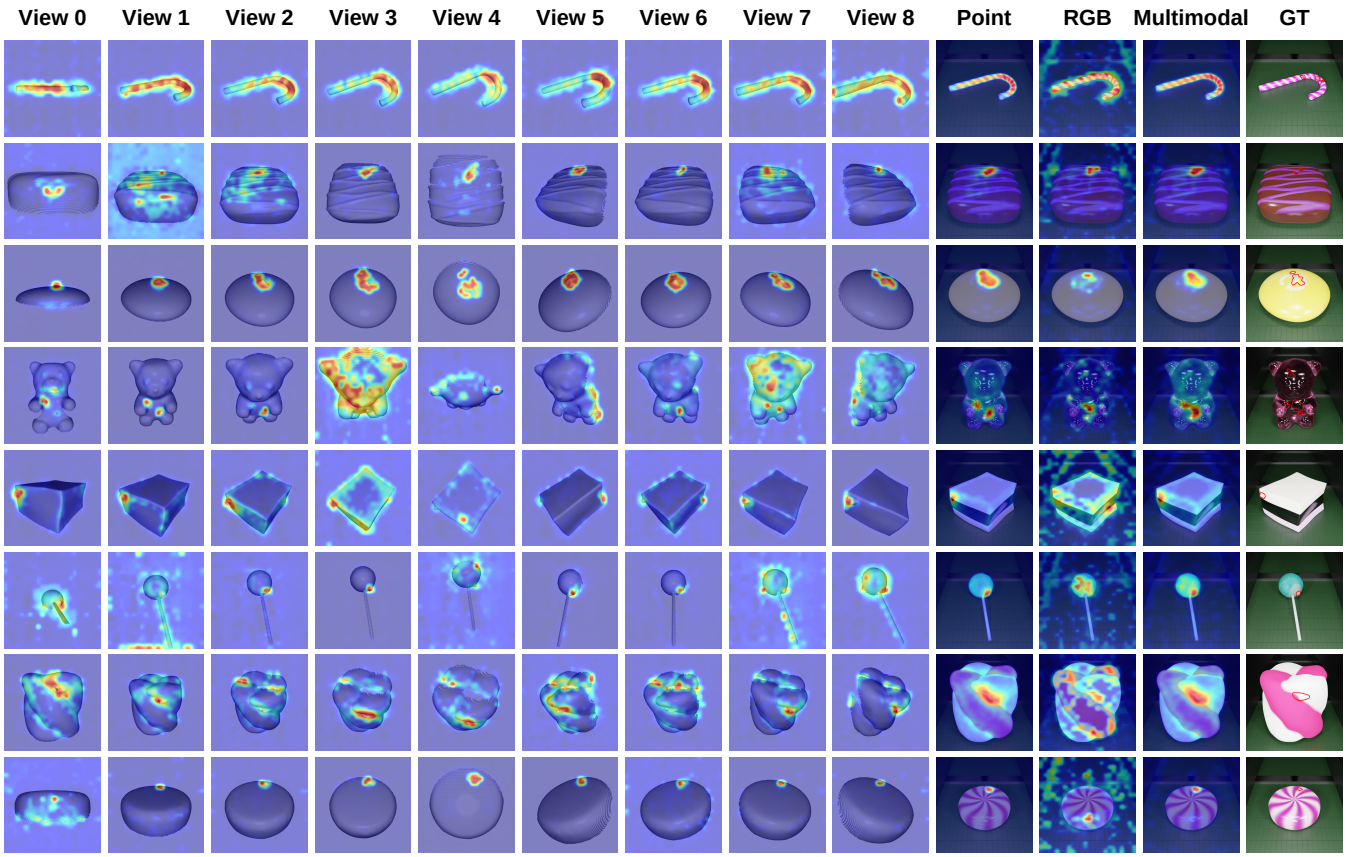

Figure 4: Visualization of anomaly score maps in Eyecandies dataset.