LayoutAD: Exploring Semantic-Geometric Misalignment Reasoning for Scene Layout Anomaly Detection

Supplementary Material

In this supplementary material, we first introduce the COCOAD benchmark in Sec. 1 and provide a detailed description of our model architecture in Sec. 2. Implementation details are presented in Sec. 3. We then report additional experimental results in Sec. 4 and demonstrate further application results in Sec. 5.

1. COCOAD Benchmark

To evaluate scene layout anomaly detection under diverse yet controllable conditions, we construct COCOAD, a benchmark derived from COCO [11] by synthetically inserting layout-inconsistent objects. As illustrated in Figure 1, given an input COCO image, Qwen2.5-VL [1] first analyzes the scene and selects an object category whose insertion would most plausibly result in a semantic or geometric inconsistency. The model is prompted to produce anomalies belonging to two categories aligned with our task definition: (i) Object–Attribute anomalies, where the inserted object violates intrinsic properties such as size, shape, appearance, or physical characteristics; and (ii) Object–Relation anomalies, where the added object violates spatial or functional relationships with surrounding objects or the environment. Qwen-Image [17] executes the generated insertion instruction through localized, context-aware editing, ensuring that all original pixels outside the edited region remain bitwise identical. SAM-HQ [7] is then applied to segment the newly added object, producing a precise anomaly mask. Each COCOAD sample therefore contains an edited image and an instance mask marking the anomalous object, enabling clean supervision for object-level anomaly localization.

Figure 2 presents an overview of the constructed COCOAD benchmark. Built upon COCO's rich and heterogeneous visual content, COCOAD naturally covers a wide variety of real-world scenes with diverse object compositions, spatial layouts, and contextual structures. Across these varied settings, the benchmark introduces numerous semantic–geometric violations—including physically misplaced objects, inconsistent co-occurrences, and abnormal object interactions. The realism and diversity of these cases highlight that many anomalies cannot be captured by low-level appearance cues alone, underscoring the necessity of high-level structural reasoning. As such, COCOAD serves as a comprehensive and challenging benchmark for evaluating scene layout anomaly detection.

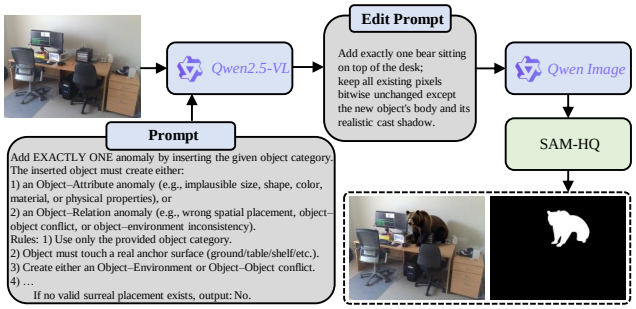

Figure 1. COCOAD generation pipeline. Qwen2.5-VL [1] selects an anomalous object category, generates an insertion instruction, and Qwen-Image [17] performs localized editing. SAM-HQ [7] then extracts the mask of the inserted object, producing a sample containing exactly one attribute- or relation-level anomaly.

2. Additional Model Details

Graph Message Passing. In the Misalignment Reasoning Module (MRM), both the semantic graph G_{sem} and geometric graph G_{geo} undergo a modality-specific message passing stage before entering the cross-graph transformer. As illustrated in Figure 3, the raw node features (s_i or g_i) and edge features are first projected into a latent space using lightweight MLPs, after which two stacked GATv2 [2] layers refine both node and edge representations. In each GATv2 [2] layer, the node i aggregates information from its neighbors j through attention-based messages of the form $\alpha_{ij} W_v v_j$, where v_j denotes the current node embedding, W_v is a learnable linear projection, and α_{ij} is an attention coefficient. The updated node representation is obtained by combining the original embedding with the aggregated messages via residual connection, while each edge embedding is refined through an edge-update MLP that takes $[v'_i \parallel v'_j \parallel e_{ij}]$ as input, where $\parallel$ represents feature concatenation, and e_{ij} is the edge feature encoding semantic or geometric relations. This alternating node–edge update process is applied twice, producing the refined node embeddings Z_{sem} and Z_{geo} used in the main paper. Overall, this message passing mechanism injects structure-aware relational cues into both graphs, ensuring that subsequent semantic-geometric alignment in the transformer operates on contextually enriched graph representations.

Global Feature Aggregation. After obtaining the refined semantic and geometric node embeddings $\hat{Z}_{sem} = \{\hat{z}_i^{sem}\}_{i=1}^N$ and $\hat{Z}_{geo} = \{\hat{z}_i^{geo}\}_{i=1}^N$, we derive a compact scene-level global feature that summarizes the overall

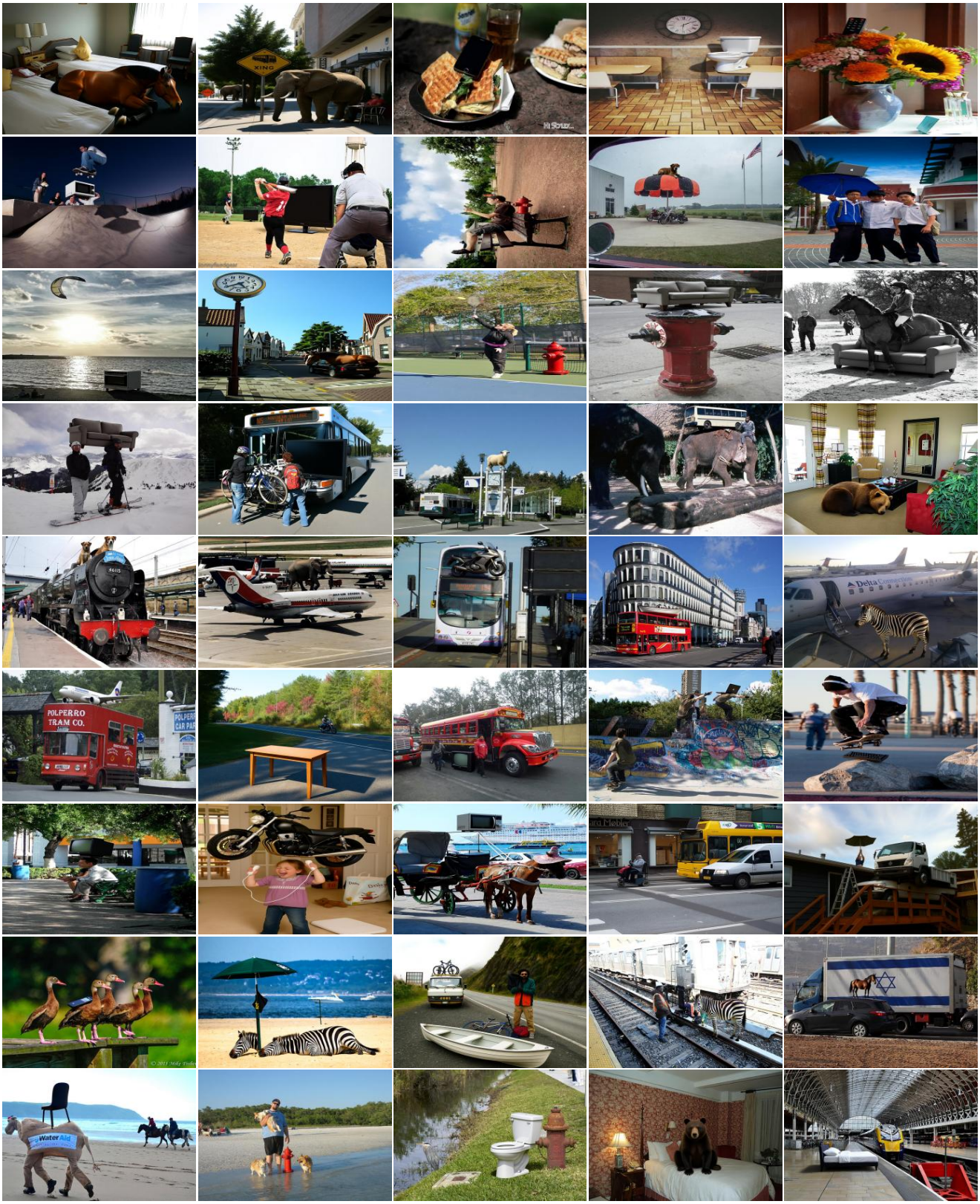

Figure 2. Overview of the COCOAD benchmark. The benchmark contains a large variety of real-world scenes with synthetically inserted layout-inconsistent objects. These examples illustrate the diversity and realism of COCOAD, showcasing numerous semantic–geometric violations such as implausible object placements and inconsistent object relationships.

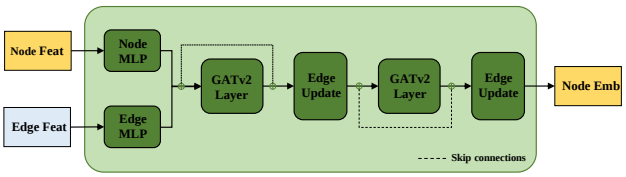

Figure 3. Message passing architecture used in the Misalignment Reasoning Module (MRM). The module first applies separate MLPs to node and edge features, followed by two GATv2 [2] layers that perform attention-based message passing and edge refinement. Skip connections preserve information from earlier layers, and the final refined node embeddings constitute the initial semantic and geometric representations used for cross-modal reasoning in the main framework.

semantic-geometric configuration of the image. For each object i, the semantic and geometric embeddings are first integrated through a gating-based fusion:

$$u_i = \alpha_i \hat{z}_i^{geo} + (1 - \alpha_i)\hat{z}_i^{sem}, \\ \alpha_i = \sigma(MLP([\hat{z}_i^{geo} || \hat{z}_i^{sem}])), \tag{1}$$

producing a unified descriptor u_i for each object. To capture holistic scene context, these fused embeddings are aggregated using a combination of mean pooling, max pooling, and attention pooling across all objects. The concatenated pooled representation is then passed through a projection layer to obtain the final global feature z_{global}, which encodes scene-level semantic–geometric structure and conditions the anomaly likelihood estimation in the Anomaly Ranking Module.

3. Additional Implementation Details

To ensure clarity and reproducibility, we summarize the main implementation configurations used across all experiments in the supplementary material. Unless otherwise noted, all settings follow the same training and inference pipeline introduced in the main paper.

We utilize Mask2Former [3] to extract instance masks and build the scene layout graph as described in the main paper. All geometric features, such as centers, sizes, and spatial relations, are normalized with respect to image dimensions, while CLIP-based semantic embeddings are L2-normalized before graph construction. For efficiency, graph connectivity is determined via k-nearest neighbors (k=4) combined with a distance threshold to preserve long-range interactions. During batching, graphs are padded to the maximum number of objects in the batch, and padding masks ensure that message passing and attention operations remain unaffected. The CLIP encoder remains frozen, while all other modules—including graph message passing, cross-graph transformer, and anomaly ranking networks—are trained jointly. Hyperparameters used for the mixture-density normality estimators follow the main paper, and loss weights remain fixed across all supplementary experiments. For inference, object-level anomaly scores are mapped back to pixel space using the corresponding segmentation masks without additional smoothing or post-processing. All visualizations are generated from raw anomaly maps, linearly normalized to $[0, 1]$. The same inference protocol is applied for additional experiments, ablations, and application results unless otherwise specified.

Method	AP ↑	FPR ↓
SynBoost [4]	0.519	0.781
PixOOD [16]	0.501	0.831
Ours	**0.536**	**0.241**

Table 1. Quantitative comparison of the proposed method with the baselines (i.e., SynBoost [4], PixOOD [16]). We evaluate their performance using AP and FPR. The best results are highlighted in bold.

Metrics	Top 5	Top 10	Top 20	Last 5	Last 10	Last 20	Average
I-AUROC ↑	0.594	0.590	0.587	0.517	0.523	0.525	0.586
P-AUROC ↑	0.892	0.885	0.874	0.852	0.856	0.862	0.871
A-P-AUROC ↑	0.905	0.898	0.870	0.855	0.871	0.883	0.883

Table 2. Quantitative evaluation on long-tail categories. *LayoutAD* demonstrates stable performance across both commonly and rarely appeared object categories.

4. Additional Experimental Results

Quantitative Evaluation. Beyond the AUROC metrics reported in the main paper, additional evaluation protocols offer a more detailed assessment of object-level anomaly localization. To this end, we further evaluate anomaly segmentation performance using instance-level Average Precision (AP) and False Positive Rate (FPR). These supplementary metrics assess the quality of object-level anomaly scores after mask projection. As shown in Table 1, *LayoutAD* achieves consistently higher AP and lower FPR compared to all baseline methods, confirming the effectiveness of semantic–geometric reasoning under stricter and more imbalanced evaluation settings.

Qualitative Evaluation. We further provide extended qualitative comparisons that include additional baseline models. As illustrated in Figure 4, *LayoutAD* more reliably highlights anomalous objects and relational inconsistencies, particularly in cases where existing pixel-level or appearance-based detectors fail to capture high-level semantic or geometric violations. These results reinforce the model's advantage in identifying diverse layout anomalies across more challenging and diverse settings.

Ablation Study. To understand how each design choice contributes to scene layout anomaly detection, we addition-

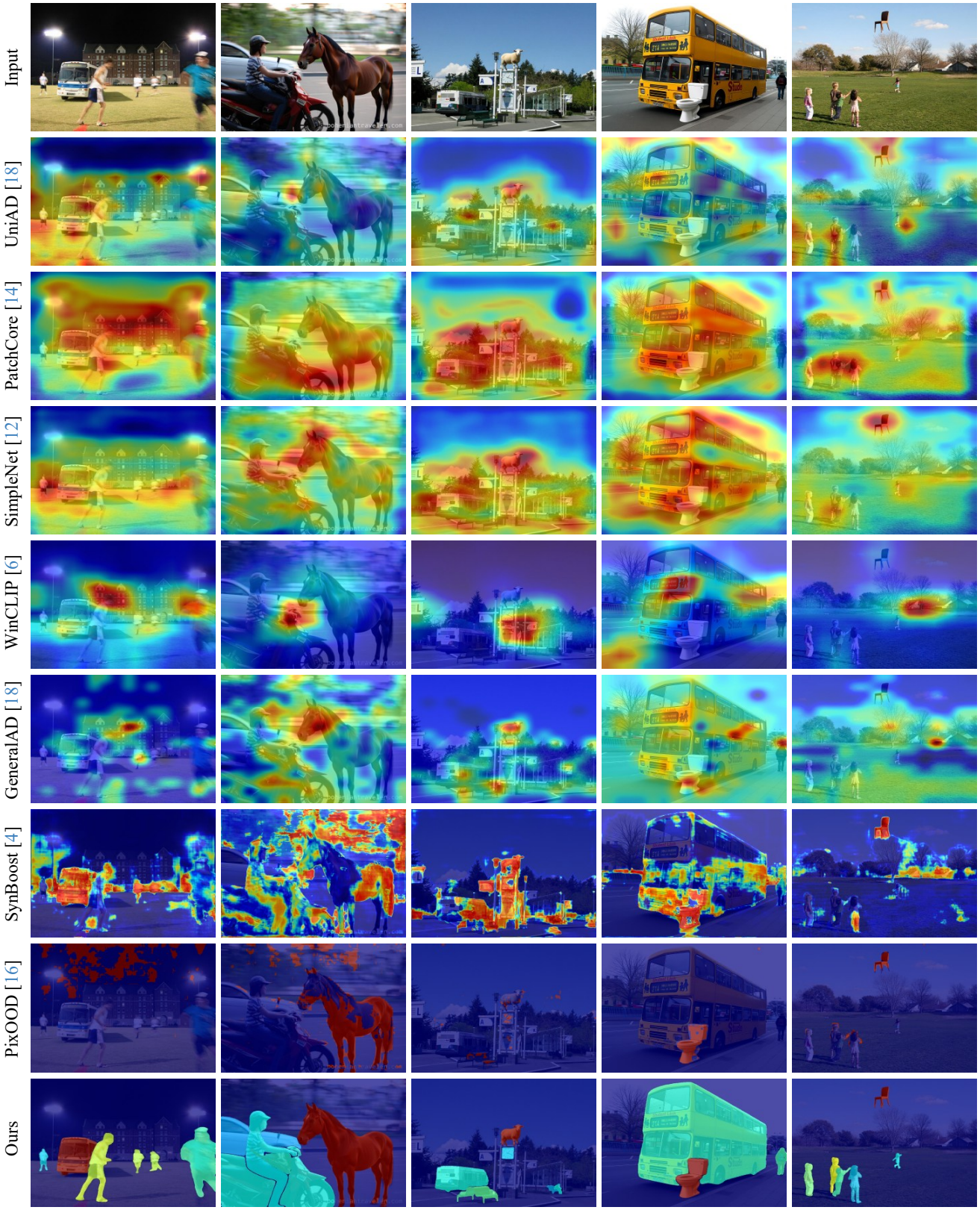

Figure 4. Additional qualitative comparison. We compared the effectiveness of *LayoutAD* against several additional baseline models (i.e., UniAD [18], PatchCore [14], SimpleNet [12], WinCLIP [6], GeneralAD [15], SynBoost [4] and PixOOD [16]) in detecting object-attribute and object-relation anomalies.

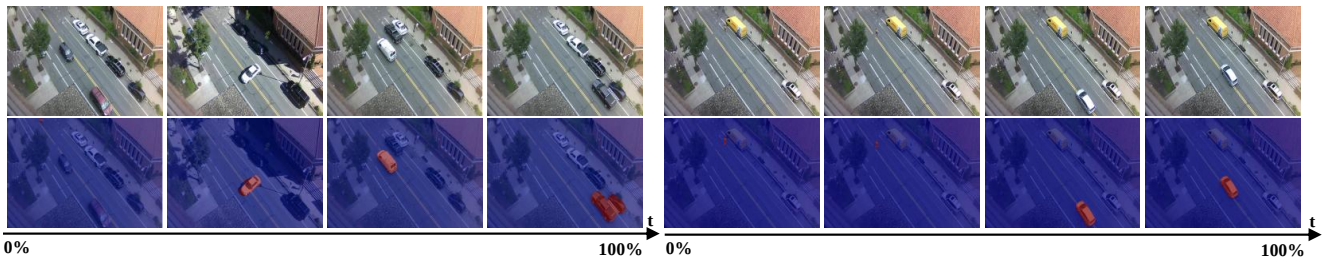

Figure 5. Video anomaly detection. The first row shows input street-scene images, and the second row displays the corresponding anomaly maps. Detected anomalous regions, such as anomalous vehicles, are highlighted in the anomaly maps. Our method effectively identifies spatial layout anomalies such as misplaced vehicles, even without relying on temporal cues, demonstrating the potential of scene layout anomaly detection in video sequences.

Method	I-AUROC ↑	P-AUROC ↑	A-P-AUROC ↑
G+G	0.473	0.765	0.772
S+S	0.535	0.842	0.843
w/o Object Loss	0.527	0.857	0.864
w/o Relation Loss	0.530	0.851	0.866
Full (S+G)	**0.586**	**0.871**	**0.883**

Table 3. Additional ablation results. The best results are highlighted in bold.

Metrics	Segmentation Models		Mask2Former Backbones		
	Panoptic FPN[9]	EoMT[8]	Swin-S	Swin-B	**Swin-L (Ours)**
I-AUROC ↑	0.563	0.566	0.552	0.567	**0.586**
P-AUROC ↑	0.854	0.867	0.847	0.844	**0.871**
A-P-AUROC ↑	0.866	0.880	0.849	0.858	**0.883**

Table 4. Robustness analysis across different segmentation models. *LayoutAD* maintains stable performance despite the varying quality of segmentation backbones.

ally conduct ablation studies by disabling or modifying key components of our framework. Beyond the ablation results reported in the main paper, we conduct additional studies that isolate the effect of further architectural and modeling components. Specifically, we consider the following variants:

- *G+G*, where both graph branches use geometric features only, removing semantic cues.
- *S+S*, where both graph branches use semantic features only, removing geometric cues.
- *w/o Object Loss*, where the object-level likelihood term in the Anomaly Ranking Module is removed.
- *w/o Relation Loss*, where the relation-level likelihood is removed.

As shown in Table 3, using G+G leads to the most severe degradation, indicating that geometric cues alone are insufficient for detecting many layout anomalies. Without semantic information, the model struggles to identify context-incompatible object appearances or inappropriate object categories, resulting in a substantial drop across all metrics. The S+S variant also shows notable performance decline. Although semantic features help recognize category-level inconsistencies, the absence of geometric cues prevents the model from capturing spatial misplacement or physically implausible configurations, which are prevalent in COCOAD. Removing the object-level likelihood (w/o Object Loss) weakens the model's ability to assess attribute–geometry coherence within individual objects, leading to reduced detection of attribute-centric anomalies. In contrast, removing the relation-level term (w/o Relation Loss) causes a more pronounced drop in relational cases, confirming the importance of modeling object–object and object–environment interactions. Together, these observations demonstrate that semantic and geometric cues provide complementary benefits, and that both object-centric and relation-centric likelihoods are essential for achieving robust scene layout anomaly detection. The full S+G model, which integrates all components, achieves the highest performance across all evaluation metrics.

To further evaluate the practical applicability of *LayoutAD*, we investigate its robustness to imperfect input masks. Specifically, we replace the default Mask2Former (Swin-L) with other off-the-shelf segmentation models, including Panoptic FPN [9], EoMT [8], and Mask2Former with smaller backbones (Swin-S, Swin-B). As shown in Table 4, while lightweight or earlier segmentation models inevitably introduce ambiguous boundaries or mask errors, the performance of *LayoutAD* remains stable with minimal fluctuations.

Robustness to Long-tail Categories. To investigate whether the reliance on CLIP representations introduces biases that affect anomaly judgment on long-tail categories or atypical scene configurations, we evaluate our model's performance on the most commonly appeared (Top 5, 10, 20) and rarely appeared (Last 5, 10, 20) object categories in the COCO dataset. As shown in Table 2, *LayoutAD* maintains robust and consistent detection capabilities across both

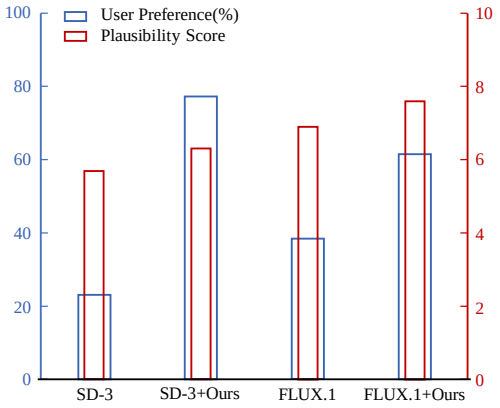

Figure 6. Quantitative comparison of self-corrected image generation. Our proposed pipeline (+Ours) significantly improves both User Preference and Plausibility Score compared to the vanilla SD-3 and FLUX.1 models.

common and rare categories. The minor performance fluctuations indicate that our model successfully leverages message passing and cross-graph attention to capture complex semantic–geometric alignments, rather than merely overfitting to the appearance priors of frequent categories.

5. Additional Application Results

We introduce scene layout anomaly detection as a more general, prompt-free task that assesses the intrinsic semantic and geometric plausibility of a scene using only the image itself. Such a problem formulation enables our proposed *LayoutAD* to address a significantly broader range of applications, including video anomaly detection, and self-corrected image generation.

5.1. Video Anomaly Detection

Although *LayoutAD* is designed for static scene layout reasoning, it can also be applied to video anomaly detection by operating on individual frames. We evaluate this capability on the Street Scene dataset [13], which contains long video sequences of complex road environments. Since our method does not model temporal dynamics, we focus on single frames that contain clear layout-related irregularities, such as vehicles appearing in incorrect lanes or occupying implausible spatial positions. As shown in Figure 5, *LayoutAD* successfully highlights anomalous vehicles that deviate from the expected road layout, despite the absence of motion cues. These qualitative results demonstrate that semantic–geometric reasoning provides a complementary perspective to traditional motion-based VAD approaches, and can effectively detect layout violations even when temporal information is not utilized.

5.2. Self-corrected Image Generation

We provide further qualitative results of our self-corrected image generation pipeline. As shown in Figure 7, these additional cases cover a wider range of prompts and generative models [5, 10], including scenes with complex multi-object interactions and challenging semantic–geometric compositions. Across diverse prompts, *LayoutAD* serves as a reliable structural verifier that filters out layout-deficient images before they are accepted. The additional visual results demonstrate that this mechanism significantly improves spatial plausibility and reduces the occurrence of object-attribute and object-relation anomalies. These observations further confirm that integrating *LayoutAD* into generative workflows provides a practical and scalable strategy for enforcing semantic–geometric coherence in T2I synthesis.

In addition to the qualitative results, we conduct a quantitative evaluation to further assess the effectiveness of our self-corrected image generation pipeline. We recruited 13 human participants to rate the user preference between the original images generated by vanilla T2I models (SD-3 [5] and FLUX.1 [10]) and our self-corrected results. Furthermore, we utilize GPT-4o as an automated judge to evaluate the overall layout plausibility of the generated scenes.

As illustrated in Figure 6, images corrected by *LayoutAD* consistently achieve significantly higher user preference and plausibility scores across different foundational models. We also assess the computational overhead of our pipeline: a standard generation step typically takes about 20 seconds, while our scene layout anomaly detection process takes only about 2 seconds per iteration. This acceptable runtime cost, combined with the substantial improvement in semantic–geometric coherence, confirms the practical viability of integrating *LayoutAD* as a reliable structural verifier for T2I systems.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 1

[2] Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? *arXiv preprint arXiv:2105.14491*, 2021. 1, 3

[3] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 3

[4] Giancarlo Di Biase, Hermann Blum, Roland Siegwart, and

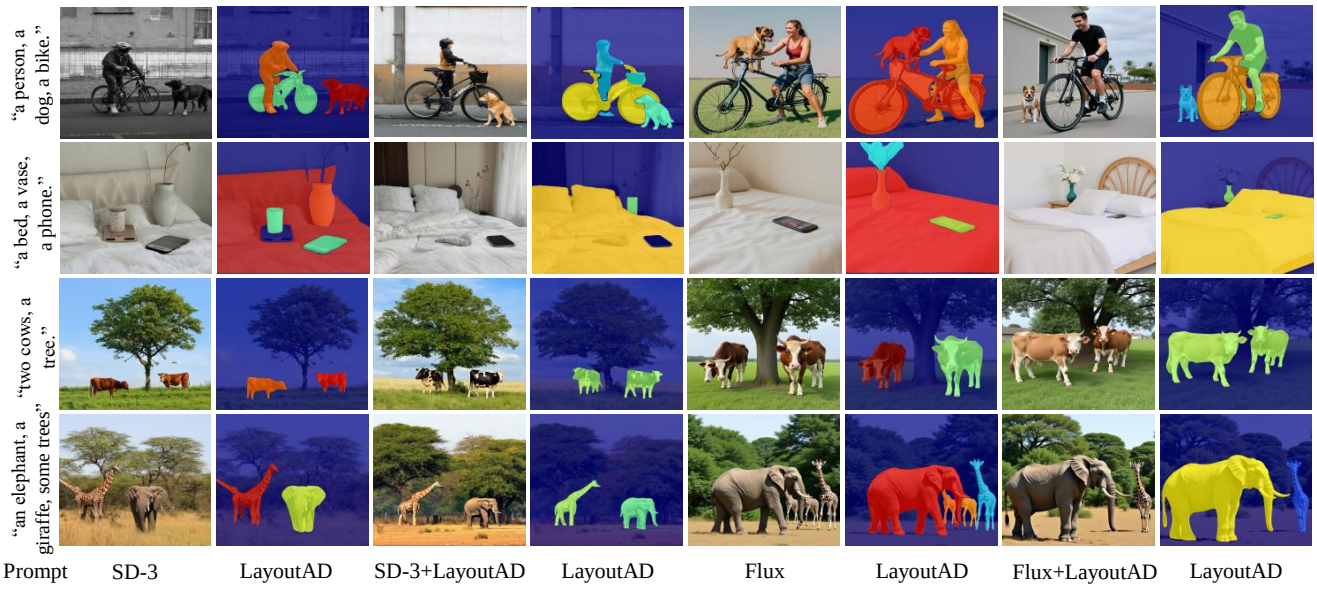

Figure 7. Additional experiments on self-corrected image generation. *LayoutAD* identifies and corrects factually defective hallucinations that appear in generative text-to-image models (e.g., SD-3 [5] and Flux [10]).

Cesar Cadena. Pixel-wise anomaly detection in complex driving scenes. In *CVPR*, 2021. 3, 4

[5] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024. 6, 7

[6] Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. Winclip: Zero-/few-shot anomaly classification and segmentation. In *CVPR*, 2023. 4

[7] Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Segment anything in high quality. In *NeurIPS*, 2023. 1

[8] Tommie Kerssies, Niccolo Cavagnero, Alexander Hermans, Narges Norouzi, Giuseppe Averta, Bastian Leibe, Gijs Dubbelman, and Daan de Geus. Your vit is secretly an image segmentation model. In *CVPR*, 2025. 5

[9] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, 2019. 5

[10] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 6, 7

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1

[12] Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. Simplenet: A simple network for image anomaly detection and localization. In *CVPR*, 2023. 4

[13] Bharathkumar Ramachandra and Michael Jones. Street scene: A new dataset and evaluation protocol for video anomaly detection. In *WACV*, 2020. 6

[14] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *CVPR*, 2022. 4

[15] Luc PJ Sträter, Mohammadreza Salehi, Efstratios Gavves, Cees GM Snoek, and Yuki M Asano. Generalad: Anomaly detection across domains by attending to distorted features. In *ECCV*, 2024. 4

[16] Tomáš Vojíř, Jan Šochman, and Jiří Matas. Pixood: Pixel-level out-of-distribution detection. In *ECCV*, 2024. 3, 4

[17] Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025. 1

[18] Zhiyuan You, Lei Cui, Yujun Shen, Kai Yang, Xin Lu, Yu Zheng, and Xinyi Le. A unified model for multi-class anomaly detection. *NeurIPS*, 2022. 4