LAYGRAPH: GRAPH GUIDED DIFFUSION WITH CONSISTENT INFERENCE FOR REMOTE SENSING LAYOUT-TO-IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Layout-to-image generation can synthesize remote sensing images from oriented layouts and expand training data for detectors. Existing diffusion models improve box-level placement, but encode objects largely independently and leave terrain to a global caption or unconstrained synthesis. Consequently, prescribed boxes need not yield compatible object–background pairings or plausible local arrangements. To address these failures, LayGraph first adopts a novel graph-structured condition that constrains the complex relations in layout-to-image generation, and then constructs a geometry features bank both to bind instance descriptors to this graph and to enable cluster retrieval at inference. Specifically, the graph represents each oriented layout with object nodes linked to spatial neighbors and to a background supernode, so relative geometry, category co-occurrence, and object–terrain couplings condition the denoiser rather than independent box tokens. To bind these nodes to instance descriptors and to keep the same slots when only a layout is given, the bank stores training-crop features on the graph; source-matched rows are read during training, and geometry-aware cluster retrieval fills those slots at layout-only inference. We evaluate this design on DIOR-R and DOTA against recent layout-conditioned generators. Relative to the best competing score on each metric, LayGraph reduces FID by 6.0% and raises YOLOScore by 27.3%; the same synthetic images further improve Oriented R-CNN mAP_{50} by 8.9% when the detector is trained without real photographs. Controlled ablations further show that the relation graph and the geometry features bank contribute complementary gains.

1 INTRODUCTION

Object detection in remote sensing underpins urban monitoring, disaster response, and land-use analysis. Unlike natural images, aerial scenes exhibit extreme scale variation, dense packing, and arbitrary object orientations, so horizontal boxes often fail to localize targets tightly. Oriented detectors (Xie et al., 2021; Liu et al., 2017) and large-scale benchmarks such as DOTA (Xia et al., 2018), DIOR (Li et al., 2020), and DIOR-R (Cheng et al., 2022) have advanced the field, yet progress remains constrained by annotation cost: labeling oriented boxes demands expert judgment of category, extent, and angle under clutter and long-tailed category frequencies, making high-quality training sets expensive to scale.

These annotation bottlenecks have motivated synthetic data for training detectors (Chen et al., 2024; Tang et al., 2025). Denoising and latent diffusion (Ho et al., 2020; Rombach et al., 2022) now dominate high-fidelity synthesis, and layout generators such as GLIGEN (Li et al., 2023), ReCo (Yang et al., 2023), LayoutDiffusion (Zheng et al., 2023), and InstanceDiffusion (Wang et al., 2024) show that boxes or masks can steer composition—yet they target ground-level scenes rather than overhead geometry.

Remote-sensing methods begin to close this gap: AeroGen (Tang et al., 2025) conditions on category tokens and oriented boxes, while CRS-Diff (Tang et al., 2024), DiffusionSat (Khanna et al., 2024), and SatSynth (Toker et al., 2024) add metadata, foundation-model priors, or semantic layouts. Per-box tokens still encode category and extent, not relations to terrain or neighbors; without explicit relational conditions, objects can land on unsuitable land cover and local arrangements can drift even when

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

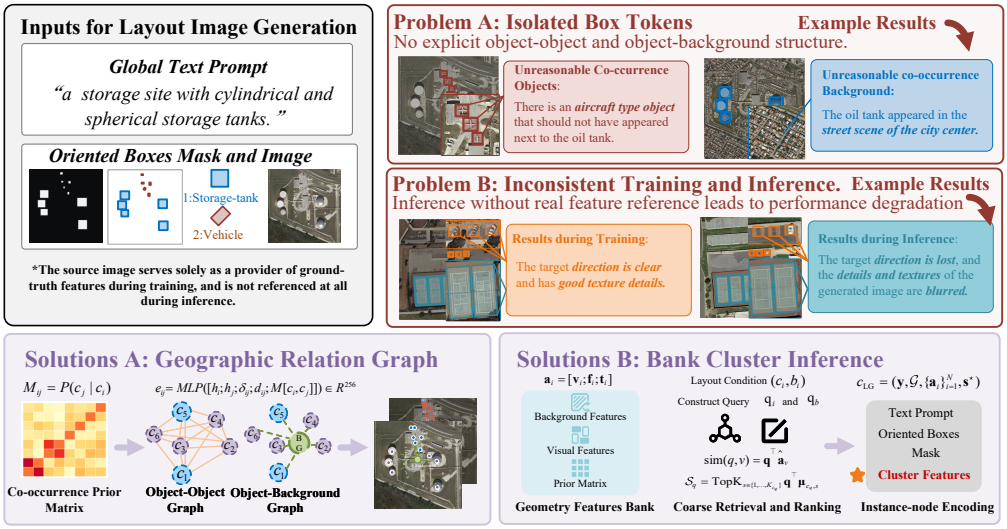


Figure 1: Problem and solution. Left: box-only layout conditioning. Right: LayGraph.

boxes are respected. AeroGen strengthens oriented-box control but neither treats the background as a graph node nor models relative-position structure. Figure 1 illustrates this gap and the proposed solution.

LayGraph (*Layout Graph*) addresses this limitation by conditioning diffusion on layout structure rather than on boxes alone. First, given an oriented layout, it builds a relation graph whose object nodes connect to spatial neighbors through edges that encode relative geometry and category co-occurrence, and links every object to a scene-level background super-node so that object–object and object–terrain couplings jointly condition the denoiser. Second, because these graph slots must still carry instance descriptors, LayGraph constructs an offline geometry features bank that binds training-crop features to the corresponding nodes and edges; at layout-only inference, when no source image is available, geometry-aware cluster retrieval fills the same slots from the bank so the conditioning interface remains consistent. Our contributions are as follows.

- We propose LayGraph, a layout-conditioned latent diffusion framework for remote-sensing layout-to-image generation that addresses relational and scene–object mismatches in synthesized images and mitigates performance drops caused by missing real-image features at inference.
- We propose a graph-structured conditioning method that encodes each oriented layout as a relation graph with neighbor edges and a background super-node, so object–object and object–terrain couplings guide the denoiser beyond independent box tokens.
- We build an offline geometry features bank that stores each training crop's descriptor on its graph node. At inference, when only a layout is given, cluster retrieval fills those same nodes with the closest stored descriptors.
- Extensive experiments on DIOR-R and DOTA show that LayGraph improves generation fidelity, layout consistency, and Oriented R-CNN performance under synthetic-only training, while ablations confirm complementary gains from the graph and the bank.

2 Related Work

Image generation. Generative adversarial networks synthesize images by training a generator against a discriminator (Goodfellow et al., 2014). Later style-based and large-scale models raised the fidelity of samples drawn from noise or a class label (Karras et al., 2019; Brock et al., 2019). Variational autoencoders take a different route: they learn an explicit latent distribution and decode samples from it (Kingma & Welling, 2014), which gives a structured latent space but typically blurrier outputs than adversarial training. Denoising diffusion closes much of that gap by learning to reverse a gradual noising process (Ho et al., 2020). Latent diffusion carries the same objective in an

autoencoder latent space (Rombach et al., 2022), and classifier-free guidance strengthens adherence to a conditioning signal without a separately trained classifier (Ho & Salimans, 2021). High-fidelity synthesis now commonly uses this diffusion family.

Layout-to-image generation. Layout-to-image synthesis renders an image whose objects occupy prescribed boxes or masks. Early work used VAEs or GANs with category-labeled boxes (Zhao et al., 2019; Sun & Wu, 2019); diffusion models are now the usual backbone. GLIGEN (Li et al., 2023) injects grounding tokens into a frozen latent diffusion network, ReCo (Yang et al., 2023) adapts text tokens for region control, and LayoutDiffusion (Zheng et al., 2023) fuses layout and image patches. InstanceDiffusion (Wang et al., 2024) adds instance-masked attention, and GeoDiffusion (Chen et al., 2024) serializes boxes into text for detection-oriented synthesis. The same per-instance box interface continues in recent work: ConsistCompose (Shi et al., 2026) writes normalized box coordinates into the language prompt, and EchoGen (Zou et al., 2026) trains layout-to-image generation jointly with grounding so the two tasks share layout tokens. Evaluations of these layout-guided models now separate semantic agreement from spatial fidelity (Parolari et al., 2026). Scene-graph methods make relations explicit (Johnson et al., 2018; Liu & Liu, 2024), yet their predicates describe indoor or everyday scenes, and objects remain largely independent tokens whose neighbor and background coupling is left to ordinary attention.

Remote sensing layout-to-image generation. Remote-sensing synthesis has moved from text and metadata toward spatial control. DiffusionSat (Khanna et al., 2024) and RSDiff (Sebaq & ElHelw, 2024) generate satellite images from text, while SatSynth (Toker et al., 2024) and CRS-Diff (Tang et al., 2024) add semantic maps or ControlNet-style conditions. These controls shape global appearance but do not place oriented instances. AeroGen (Tang et al., 2025) conditions latent diffusion on category tokens and horizontal or oriented boxes; later work adds relation constraints (Qiao et al., 2026) or object fidelity without a reference image at inference (Ye et al., 2026). ChangeBridge (Zhao et al., 2026) conditions a pre-event to post-event bridge on instance layouts, coordinate text, or semantic masks, and TODSynth (Yang et al., 2026) conditions a multimodal transformer on semantic masks for segmentation data. Instances are still encoded largely as independent box tokens, so neighbor relations, object–background coupling, and appearance that stays consistent under layout-only inference remain under-explored.

3 METHOD

3.1 PRELIMINARIES

Layout-to-image generation maps a global prompt $\mathbf{y}$ and an oriented layout to an image $\mathbf{x}$. The layout lists N instances,

$$\mathcal{L} = \big\{(c_i, \mathbf{b}_i)\big\}_{i=1}^{N}, \tag{1}$$

where $c_i \in \{1, \dots, C\}$ is the category of instance i and $\mathbf{b}_i \in \mathbb{R}^8$ stores the four corner coordinates of its oriented box on an $S \times S$ grid. A generator G_θ realizes the mapping $\mathbf{x} = G_\theta(\mathbf{y}, \mathcal{L})$.

We instantiate G_θ as a latent diffusion model (Rombach et al., 2022; Ho et al., 2020). A variational autoencoder encodes $\mathbf{x}$ as a clean latent $\mathbf{z}_0$. At timestep t, the forward process forms a noisy latent $\mathbf{z}_t = \sqrt{\bar{\alpha}_t}\,\mathbf{z}_0 + \sqrt{1-\bar{\alpha}_t}\,\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\bar{\alpha}_t$ is the cumulative noise schedule. A denoiser $\boldsymbol{\epsilon}_\theta$ is trained to predict this noise from $\mathbf{z}_t$, the timestep, and a condition c:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{\mathbf{z}_0, \boldsymbol{\epsilon}, t, c}\big[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, c)\|_2^2\big]. \tag{2}$$

In a standard layout-conditioned model, c is assembled from $\mathbf{y}$ and one token per box in $\mathcal{L}$. Each token records the category and the box geometry of a single instance. Neighbor relations and object–terrain coupling are left for the denoiser to infer from these tokens. Section 3.2 replaces this flat condition with the LayGraph condition c_{LG}.

3.2 FRAMEWORK OVERVIEW

Figure 2 shows the LayGraph pipeline. Given a prompt $\mathbf{y}$ and an oriented layout $\mathcal{L}$, an offline geometry features bank supplies an instance descriptor $\mathbf{a}_i$ for each box and a scene descriptor $\mathbf{s}$ for

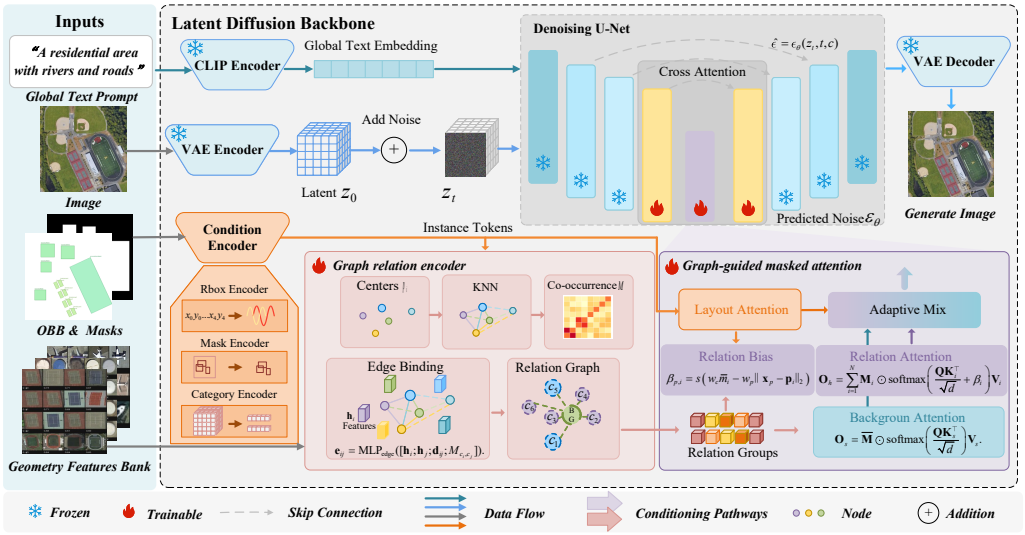


Figure 2: LayGraph pipeline. From a prompt and an oriented layout, a geometry features bank and a relation graph form the diffusion condition; training reads source-matched bank rows, and layout-only inference fills the same slots by cluster retrieval.

the region outside all boxes. During training, these descriptors are read from the bank row matched to the source image; when only a layout is available at inference, geometry-aware cluster retrieval writes into the same fields.

Each instance node v_i combines its oriented box with $\mathbf{a}_i$, and a background super-node v_0 is built from $\mathbf{s}$. The resulting relation graph is

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}), \qquad \mathcal{V} = \{v_i\}_{i=1}^{N} \cup \{v_0\}, \tag{3}$$

where object–object edges carry relative position and category co-occurrence, and edges to v_0 carry the displacement of each object from the image center. The condition passed to the denoiser is

$$c_{\mathrm{LG}} = \big(\mathbf{y}, \mathcal{G}, \{\mathbf{a}_i\}_{i=1}^{N}, \mathbf{s}\big). \tag{4}$$

In the UNet spatial transformer, instance tokens act inside oriented-box masks, relation tokens and a layout bias inject neighbor structure, a scene summary acts outside the masks, and the text prompt provides global context. Substituting $c = c_{\mathrm{LG}}$ into Eq. equation 2, we optimize

$$\min_{\theta} \; \mathcal{L}_{\mathrm{diff}} + \lambda_e \, \mathcal{L}_e(\mathcal{G}; \theta), \tag{5}$$

with λ_e=0.05, where $\mathcal{L}_e$ is an auxiliary objective on the retained edges. The VAE decoder then maps the denoised latent to an RGB image. The following subsections detail the geometry features bank and relation graph, then the layout attention and inference-time retrieval.

3.3 Geometry features bank and graph relation encoding

Geometry features bank. We construct an offline geometry features bank from the training images. For each annotated crop, we store a CLIP visual feature $\mathbf{v}_i$ (Radford et al., 2021), a compact spectral feature $\mathbf{f}_i$ from color statistics and the crop spectrum, and a text feature $\mathbf{t}_i$ that encodes the category, size, and heading. The instance descriptor is

$$\mathbf{a}_i = [\mathbf{v}_i; \mathbf{f}_i; \mathbf{t}_i], \tag{6}$$

and a scene descriptor $\mathbf{s}$ is computed on the region outside all labeled boxes from visual and spectral features. Bank rows are indexed by category and box geometry. During training, each layout instance and its background receive the row matched to the source image. When only a layout is available, the same fields are filled by geometry-aware cluster retrieval.

Graph relation encoder. For object i, the node feature combines the oriented box with its bank descriptor:

$$\mathbf{h}_i = \mathrm{MLP}_{\mathrm{node}}\big([\gamma(\mathbf{b}_i); \mathbf{a}_i]\big), \tag{7}$$

where γ is the Fourier encoding of the four box corners. Neighbor information is kept in edge features rather than pooled into $\mathbf{h}_i$. Let $\mathbf{p}_i$ be the center of box $\mathbf{b}_i$. We build a symmetric k-nearest-neighbor graph over valid instances and retain an edge (i, j) when the category co-occurrence prior M_{c_i,c_j} exceeds threshold τ; same-category pairs are always retained. Each retained edge uses the relative displacement and distance

$$\mathbf{d}_{ij} = \big[\mathbf{p}_i - \mathbf{p}_j;\ \|\mathbf{p}_i - \mathbf{p}_j\|_2\big], \tag{8}$$

which are combined with the endpoint features to form an edge embedding:

$$\mathbf{e}_{ij} = \mathrm{MLP}_{\mathrm{edge}}\big([\mathbf{h}_i; \mathbf{h}_j; \mathbf{d}_{ij}; M_{c_i,c_j}]\big). \tag{9}$$

Here M_{c_i,c_j} is the empirical co-occurrence of the two categories. We map $\mathbf{s}$ to a background supernode $\mathbf{g} = \mathrm{MLP}_{\mathrm{bg}}(\mathbf{s})$ at the image center and connect every object to this node with the same edge encoder and the displacement from the center. An auxiliary binary objective $\mathcal{L}_e$ is applied to the edge embeddings: it predicts whether a candidate edge is retained by the kNN and co-occurrence rule above. These edge embeddings are packed into relation tokens and injected into the UNet in the next subsection. Figure 8 in Appendix F visualizes one such graph.

3.4 Layout attention and inference-time retrieval

Graph-guided masked attention. Layout conditions enter the UNet spatial transformer by cross-attention with a shared query $\mathbf{Q}$ from spatial features $\mathbf{U}$. Instance control is

$$\mathbf{O}_h = \sum_{i=1}^{N} \mathbf{M}_i \odot \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}_i^\top}{\sqrt{d}} + \boldsymbol{\beta}_i\right)\mathbf{V}_i, \tag{10}$$

where $\mathbf{K}_i$ and $\mathbf{V}_i$ come from instance token $\mathbf{h}_i$, and $\mathbf{M}_i$ is the rasterized oriented-box mask of instance i (Wang et al., 2024). Outside $\mathbf{M}_i$, the logits of that instance do not enter the softmax. Object–object and object–background edge embeddings are packed as relation tokens $\{\mathbf{r}_r\}$. Relation control uses the same query,

$$\mathbf{O}_r = \mathbf{M}_r \odot \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}_r^\top}{\sqrt{d}}\right)\mathbf{V}_r, \tag{11}$$

where $\mathbf{K}_r$ and $\mathbf{V}_r$ come from $\{\mathbf{r}_r\}$, and $\mathbf{M}_r$ is a validity mask scaled by a relation warmup. The instance logits are further modulated by

$$\beta_{p,i} = s\big(w_c\,\bar{m}_i - w_p\,\|\mathbf{x}_p - \mathbf{p}_i\|_2\big), \tag{12}$$

where $\mathbf{x}_p$ is the coordinate of spatial position p, $\bar{m}_i$ is the mean co-occurrence of instance i with its graph neighbors, w_c and w_p are learned scalars, and s is a layout warmup that is small early in training. Background control uses the same query with scene keys and values:

$$\mathbf{O}_s = \bar{\mathbf{M}} \odot \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}_s^\top}{\sqrt{d}}\right)\mathbf{V}_s. \tag{13}$$

Here $\mathbf{K}_s$ and $\mathbf{V}_s$ come from background tokens $\mathbf{B}$ that stack a scene summary with $\mathbf{g}$, and $\bar{\mathbf{M}}$ is the complement of all instance masks. The text prompt continues to supply global context through the pretrained text cross-attention.

Appearance retrieval for layout-only inference. When only a layout is available, bank rows are filled by retrieval instead of a source-image lookup. Offline, instance rows are clustered per category into geometry-aware subtypes, and scene rows are clustered by a layout signature (class histogram and coverage). For instance i, the query is the class prototype of c_i together with the area, aspect, and heading of $\mathbf{b}_i$. Retrieval first restricts candidates to the nearest subtype clusters, then ranks a bank row v by

$$S(q, v) = \mathrm{sim}(q, v) + \alpha\, g(q, v) + \lambda\, \bar{M}(q, v), \tag{14}$$

where sim is cosine similarity to the stored descriptor, g scores geometry-bin agreement, and $\bar{M}$ is mean co-occurrence between the query's spatial neighbors and the candidate. Scene $\mathbf{s}$ is retrieved analogously from scene clusters. The selected descriptors occupy the bank-descriptor slots throughout denoising. Figure 3 summarizes bank construction and inference-time cluster retrieval. Appendix E gives cluster sizes, geometry bins, and the retrieval score used in the reported runs.

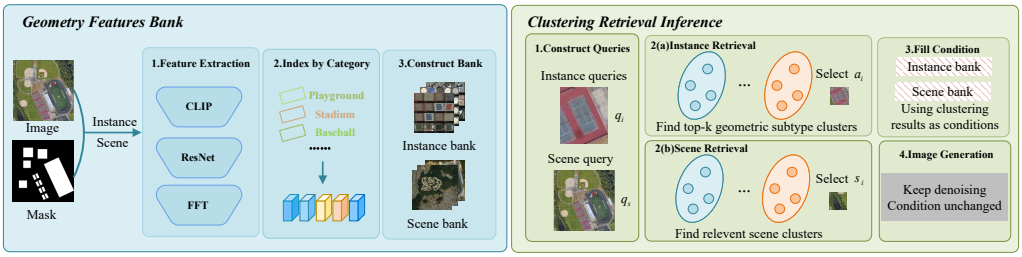


Figure 3: Geometry features bank and cluster retrieval. Training uses source-matched bank rows; inference retrieves descriptors from geometry-aware clusters.

4 Experiments

4.1 Experimental settings

Datasets. DIOR (Li et al., 2020) contains 23,463 optical remote-sensing images of size 800×800, with 20 categories and 192,472 instances, split into 5,862 training, 5,863 validation, and 11,738 test images. DIOR-R (Cheng et al., 2022) keeps this image set and split and replaces the horizontal boxes with oriented boxes, so it is the primary benchmark. DOTA-v1.0 (Xia et al., 2018) contains 2,806 aerial images with side lengths from about 800 to 4,000 pixels, 15 categories, and 188,282 instances; its public split has 1,411 training and 458 validation images, while the 937 test images are held out. Oriented generation and detection use the oriented boxes of DIOR-R and DOTA; horizontal detection uses the horizontal boxes of DIOR and DOTA. All images are resized to 512×512 for training and sampling.

Baselines. We compare against LayoutDiffusion (Zheng et al., 2023) (LayoutDiff in the tables), GLIGEN (Li et al., 2023), AeroGen (Tang et al., 2025), CC-Diff (Zhang et al., 2025), and OF-Diff (Ye et al., 2026). The set covers layout control, remote sensing synthesis, contextual coherence, and instance fidelity. All methods share the same splits, layouts, and metric definitions on each dataset.

Implementation details. We initialize LayGraph from Stable Diffusion 1.5 and train with AdamW. Rank 8 LoRA adapters with scaling 16 update the spatial transformer attention projections; the layout, relation, and scene modules are trained in full. Appendix G contrasts this choice with full-parameter updates of the same projections. The batch size is 8, the learning rate is 2×10^{-5}, and training lasts 100 epochs. The relation consistency loss in Eq. equation 5 is weighted by $\lambda_e = 0.05$. Sampling uses DDIM (Song et al., 2021) for 50 steps. The co-occurrence prior M is estimated from training annotations. The geometry features bank is built only from the training split. Training reads the row matched to the source training image. At inference, when only a layout is given, cluster retrieval queries this same training bank and does not read validation or test images.

Metrics. We report four families of scores. Distributional fidelity uses FID, KID, and CMMD against real images. Layout and category fidelity use CAS and YOLOScore. Oriented shape uses Contour IoU, Dice, Chamfer distance, Hausdorff distance, and SSIM on paired instance edges. Downstream utility uses detection mAP when a detector is trained on synthetic images only, and Relation-Eval for claims that FID and mAP leave open: OTAcc for terrain compatibility, RelCooc and RelF1 for neighbor co-occurrence, and AvgBox for recovered layout boxes. Relation-Eval details are in Appendix A; real+synthetic mixing is in Appendix D.

4.2 Qualitative results

Figure 4 compares generations from the same oriented layouts on DIOR-R. Box-conditioned baselines often break either the surrounding terrain or the local packing: ships and tank farms sit on incompatible land cover, and dense groups such as storage tanks or marina slips lose their regular arrangement. LayGraph keeps these instances on compatible backgrounds and preserves the neighbor structure in the layout. The comparison is consistent with the two conditioning interfaces: the relation graph for terrain and adjacency, and geometry-aware retrieval for oriented instance appearance. Appendix C repeats the comparison on denser DIOR-R layouts.



Figure 4: Qualitative comparison on DIOR-R under identical layouts. The last column is the corresponding real image.

4.3 Quantitative results

Generation quality. Tables 1 and 2 compare layout-conditioned generators on DIOR-R and DOTA. FID and KID are distances between the generated collection and real images in a feature space, so the lowest scores are the evidence that the generated distribution lies closer to real imagery; LayGraph is lowest on both datasets. CAS scores category agreement with the layout and YOLOScore scores how strongly a detector fires on the requested classes, so the highest scores are the evidence that those categories remain more detectable; LayGraph leads both. Table 2 shows the same pattern for oriented shape: IoU, Dice, Chamfer distance, and Hausdorff distance are best on both corpora.

CMMD and SSIM are the exceptions. GLIGEN records the best CMMD and CC-Diff the best SSIM, because these scores reward CLIP or pixel agreement with one paired real crop. Two other views of the same images point the other way. Detectors trained only on synthetic data (Table 3) rank LayGraph above GLIGEN and CC-Diff for both oriented and horizontal detection, so a tighter match to a reference crop does not yield a more useful training set. Figure 5 shows the same split in feature space: a generator can lie close to individual real samples while covering the real cloud narrowly, whereas LayGraph overlaps the real features with a broader spread.

Table 1: Generation fidelity and layout consistency on DIOR-R and DOTA. LayoutDiff denotes LayoutDiffusion.

	DIOR-R					DOTA				
Method	FID↓	KID↓	CMMD↓	CAS↑	YOLOScore↑	FID↓	KID↓	CMMD↓	CAS↑	YOLOScore↑
LayoutDiff	46.93	0.024	0.902	52.69	33.63	51.29	0.019	0.755	52.14	37.09
GLIGEN	28.29	0.010	**0.306**	47.04	19.93	41.04	0.012	**0.427**	49.03	24.65
AeroGen	39.09	0.025	0.573	64.21	31.16	46.98	0.020	0.574	58.48	35.73
CC-Diff	50.91	0.029	0.703	48.21	41.73	53.48	0.022	0.645	49.68	41.54
OF-Diff	24.18	0.013	0.566	73.53	25.31	38.79	0.013	0.570	63.58	32.51
LayGraph	**22.74**	**0.008**	0.418	**76.83**	**53.11**	**38.28**	**0.012**	0.497	**65.37**	**47.82**

Downstream detection. Table 3 tests whether synthetic images alone can supervise detectors. Oriented R-CNN on OBB and YOLOv8 on HBB both peak with LayGraph on DIOR-R, DIOR, and DOTA. Methods with stronger CMMD transfer less well, which is consistent with Table 1: agreement with a paired real crop does not imply that a detector trained on the synthetic set will generalize.

Table 2: Contour level shape fidelity on paired instance edge maps of DIOR-R and DOTA.

	DIOR-R					DOTA				
Method	IoU↑	Dice↑	CD↓	HD↓	SSIM↑	IoU↑	Dice↑	CD↓	HD↓	SSIM↑
LayoutDiff	0.057	0.084	47.03	55.82	0.216	0.083	0.114	49.13	58.38	0.310
GLIGEN	0.077	0.113	37.04	46.71	0.220	0.094	0.130	43.61	53.37	<u>0.313</u>
AeroGen	0.093	0.142	26.96	37.34	0.210	0.103	0.145	38.09	48.21	0.307
CC-Diff	0.081	0.116	38.94	47.94	**0.241**	0.096	0.131	44.68	54.04	**0.324**
OF-Diff	<u>0.096</u>	<u>0.153</u>	<u>18.27</u>	<u>29.06</u>	<u>0.197</u>	<u>0.105</u>	<u>0.151</u>	<u>33.31</u>	<u>43.66</u>	0.301
LayGraph	**0.099**	**0.161**	**13.95**	**24.65**	0.195	**0.106**	**0.154**	**26.97**	**38.36**	0.258

Table 3: Detection with synthetic training data only. (a) Oriented R-CNN on OBB. (b) YOLOv8 on HBB.

(a) Oriented R-CNN

	DIOR-R			DOTA		
Method	mAP	mAP_{50}	mAP_{75}	mAP	mAP_{50}	mAP_{75}
GLIGEN	26.83	45.17	27.36	29.91	51.76	29.52
AeroGen	31.18	52.46	33.17	36.64	59.27	35.23
CC-Diff	31.65	50.74	34.47	36.31	58.03	35.78
OF-Diff	<u>32.34</u>	<u>54.21</u>	<u>34.76</u>	<u>37.12</u>	<u>62.39</u>	<u>37.21</u>
LayGraph	**33.71**	**59.06**	**37.42**	**39.08**	**65.76**	**39.84**

(b) YOLOv8

	DIOR			DOTA		
Method	mAP	mAP_{50}	mAP_{75}	mAP	mAP_{50}	mAP_{75}
GLIGEN	27.38	47.82	28.07	31.17	53.36	29.81
AeroGen	32.96	55.09	34.23	37.97	62.18	36.44
CC-Diff	32.47	53.69	35.36	37.41	60.61	37.64
OF-Diff	<u>33.01</u>	<u>56.63</u>	<u>36.14</u>	<u>38.03</u>	<u>63.92</u>	<u>38.47</u>
LayGraph	**34.58**	**60.76**	**38.79**	**39.84**	**68.58**	**41.28**

Appendix B breaks the same synthetic-only protocol into per-class scores. Appendix D reports two further uses of the same synthetic images: as a DIOR-R augmentation source next to Flip, CopyPaste, and AeroGen, and as a 10k mix into the real DIOR and DOTA training sets.

Relation-Eval. FID and detection mAP do not isolate the two failures that LayGraph targets: instances on the wrong terrain, and neighboring categories that do not follow the input graph. Table 4 therefore reports Relation-Eval on DIOR-R and DOTA under shared layouts. OTAcc scores whether instances occupy terrain permitted for their category; RelCooc and RelF1 score whether neighboring categories co-occur as in the input graph; AvgBox is the mean number of matched boxes per image. LayGraph leads all generators on both corpora and is nearest to Real on RelCooc. Figure 5 makes a complementary point: overlap with real features in t-SNE can coexist with weak Relation-Eval, whereas LayGraph stays on the real manifold with broader appearance coverage.

Table 4: Relation-Eval on DIOR-R and DOTA under shared layouts. OTAcc: instances on terrain permitted for their category; RelCooc/RelF1: co-occurrence among neighboring objects; AvgBox: matched boxes per image. Higher is better. Best and second best among generators in bold and underlined.

(a) DIOR-R

Method	OTAcc↑	RelCooc↑	RelF1↑	AvgBox↑
LayoutDiff	0.498	0.081	0.361	1.16
GLIGEN	0.529	0.007	0.244	0.27
CC-Diff	0.513	0.143	0.456	1.72
AeroGen	0.525	0.164	0.492	1.93
OF-Diff	<u>0.540</u>	<u>0.196</u>	<u>0.528</u>	<u>2.76</u>
LayGraph	**0.558**	**0.289**	**0.665**	**5.08**
Real	0.617	0.309	0.706	6.53

(b) DOTA

Method	OTAcc↑	RelCooc↑	RelF1↑	AvgBox↑
LayoutDiff	0.481	0.074	0.338	1.05
GLIGEN	0.511	0.009	0.221	0.34
CC-Diff	0.497	0.131	0.428	1.58
AeroGen	0.508	0.152	0.461	1.78
OF-Diff	<u>0.522</u>	<u>0.181</u>	<u>0.498</u>	<u>2.51</u>
LayGraph	**0.541**	**0.268**	**0.628**	**4.76**
Real	0.591	0.287	0.672	7.12

4.4 Ablation Study

We compare selected combinations of the relation graph (RG), the background super-node (BG), the geometry features bank (GB), and cluster retrieval (CR) on DIOR-R (Table 5), under the same Relation-Eval protocol as Table 4. RG and BG correct different errors: RG mainly raises neighbor scores (RelCooc, RelF1), whereas BG mainly raises OTAcc, and either factor alone leaves the other family weak. Using both cuts FID from 36.82 to 24.58. The bank plays a separate role. GB together with CR, and without the graph, improves detectability while neighbor scores stay low; once RG and BG are already on, GB and then CR supply the remaining gains in fidelity and RelF1, with CR used when inference provides only a layout. Relational edges therefore account for which neighbors and terrain an instance respects, and retrieved appearance for how that instance is rendered.

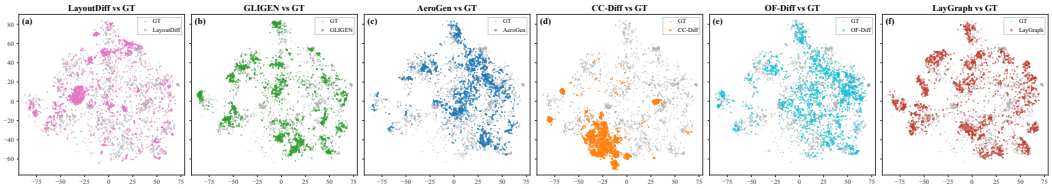


Figure 5: t-SNE of generated vs. real ResNet-50 features on DIOR-R. Gray: ground truth; color: each generator.

Table 5: Component ablation on DIOR-R. RG: relation graph; BG: background super-node; GB: geometry features bank; CR: cluster retrieval. Shape metrics follow Table 2.

Components				Fidelity / Det.				Relation-Eval				Shape				
RG	BG	GB	CR	FID↓	CAS↑	YOLO↑	mAP_{50} ↑	OTAcc↑	RelCooc↑	RelF1↑	AvgBox↑	IoU↑	Dice↑	CD↓	HD↓	SSIM↑
✗	✗	✗	✗	36.82	66.04	31.81	52.17	0.501	0.092	0.318	1.42	0.052	0.083	21.37	34.12	0.145
✓	✗	✗	✗	30.18	70.37	38.51	54.08	0.512	0.198	0.512	3.18	0.089	0.145	18.23	29.87	0.215
✗	✓	✗	✗	28.47	68.76	38.11	53.79	0.541	0.118	0.356	1.89	0.065	0.103	20.14	32.48	0.168
✓	✓	✗	✗	24.58	74.13	46.11	56.37	0.549	0.251	0.598	4.21	0.095	0.153	17.42	28.47	0.231
✗	✗	✓	✓	32.76	73.08	44.91	55.18	0.518	0.141	0.402	2.64	0.061	0.098	20.53	32.86	0.161
✓	✓	✓	✗	23.41	74.82	52.37	57.83	0.554	0.276	0.641	4.72	0.098	0.158	15.76	26.38	**0.238**
✓	✓	✓	✓	**22.74**	**76.83**	**53.11**	**59.06**	**0.558**	**0.289**	**0.665**	**5.08**	**0.099**	**0.161**	**13.95**	**24.65**	0.195

4.5 Limitations

The co-occurrence prior is estimated once on the training set, and an edge is retained only when the two categories co-occur often enough. A rare but valid pairing can therefore fall below the threshold and never enter the relation tokens, even though that pairing appears in a real scene. The graph itself is a symmetric k-nearest-neighbor graph, so it records local spacing and leaves distant objects unconnected when their arrangement is still part of the scene. Retrieval has a separate limit. Each query returns a coarse cluster built from training crops, and a distinctive texture in that cluster can be copied onto the generated instance. The image then stays close to real imagery, while the variety of appearances inside a category is narrower than in the training set.

5 Conclusion

Oriented boxes locate each instance but leave terrain, neighbor structure, and the appearance inside the box unspecified. LayGraph couples every object to its spatial neighbors and to a background super-node, so relative geometry and object–terrain pairing condition the denoiser as edges rather than as independent box tokens, and fills oriented appearance from an offline geometry features bank, using cluster retrieval when inference provides only a layout. On DIOR-R and DOTA, this improves generation fidelity, layout consistency, and detector training relative to box-conditioned generators. Further progress on layout-to-image generation depends on relations that reach beyond local co-occurrence, and on appearance retrieval that preserves diversity when the input is boxes alone.

AI use statement

Large language models were used as writing and literature-organization assistants for this manuscript (section outlining, related-work clustering, and LaTeX drafting). They were not used to generate experimental numbers. All architectural claims were checked against the implementation (oriented box encoder, relation graph, background super-node, adaptive weights, and losses). The authors take full responsibility for the text.

References

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019.

Kai Chen, Enze Xie, Zhe Chen, Yibo Wang, Lanqing Hong, Zhenguo Li, and Dit-Yan Yeung. GeoDiffusion: Text-prompted geometric control for object detection data generation. In *International Conference on Learning Representations*, 2024.

Gong Cheng, Jiabao Wang, Ke Li, Xingxing Xie, Chunbo Lang, Yanqing Li, and Junwei Han. Anchor-free oriented proposal generator for object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2022.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS Workshop on Deep Generative Models and Downstream Applications*, 2021.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1219–1228, 2018.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

Samar Khanna, Patrick Liu, Linqi Zhou, Chenlin Meng, Robin Rombach, Marshall Burke, David Lobell, and Stefano Ermon. DiffusionSat: A generative foundation model for satellite imagery. In *International Conference on Learning Representations*, 2024.

Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

Ke Li, Gang Wan, Gong Cheng, Liqiu Meng, and Junwei Han. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169:157–171, 2020.

Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. GLIGEN: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22511–22521, 2023.

Jinxiu Liu and Qi Liu. R3CD: Scene graph to image generation with relation-aware compositional contrastive control diffusion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 3657–3665, 2024.

Zikun Liu, Liu Yuan, Lubin Weng, and Yiping Yang. A high resolution optical satellite image dataset for ship recognition and some new baselines. In *Proceedings of the International Conference on Pattern Recognition Applications and Methods*, pp. 324–331, 2017.

Luca Parolari, Nicla Faccioli, and Lamberto Ballan. Benchmarking layout-guided diffusion models through unified semantic-spatial evaluation in closed and open settings. In *Findings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1742–1751, 2026.

Yuan-gan Qiao, Xiang Zhang, Chenji Wang, Shengmei Chen, Yanzi Shi, Hangzai Luo, and Sheng Zhong. Layout-controlled synthetic data generation for remote sensing object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 2026. doi: 10.1109/TGRS.2026.3679408.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Ahmad Sebaq and Mohamed ElHelw. RSDiff: Remote sensing image generation from text using diffusion model. *Neural Computing and Applications*, 36(36):23103–23111, 2024. doi: 10.1007/s00521-024-10363-3.

Xuanke Shi, Boxuan Li, Xiaoyang Han, Zhongang Cai, Lei Yang, Quan Wang, and Dahua Lin. ConsistCompose: Unified multimodal layout control for image composition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 495–505, 2026.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

Wei Sun and Tianfu Wu. Image synthesis from reconfigurable layout and style. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10531–10540, 2019.

Datao Tang, Xiangyong Cao, Xingsong Hou, Zhongyuan Jiang, Junmin Liu, and Deyu Meng. CRS-Diff: Controllable remote sensing image generation with diffusion model. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–14, 2024.

Datao Tang, Xiangyong Cao, Xuan Wu, Jialin Li, Jing Yao, Xueru Bai, Dongsheng Jiang, Yin Li, and Deyu Meng. AeroGen: Enhancing remote sensing object detection with diffusion-driven data generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3614–3624, 2025.

Aysim Toker, Marvin Eisenberger, Daniel Cremers, and Laura Leal-Taixé. SatSynth: Augmenting image-mask pairs through diffusion models for aerial semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27695–27705, 2024.

Xudong Wang, Trevor Darrell, Sai Saketh Rambhatla, Rohit Girdhar, and Ishan Misra. InstanceDiffusion: Instance-level control for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. DOTA: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3974–3983, 2018.

Xingxing Xie, Gong Cheng, Jiabao Wang, Xiwen Yao, and Junwei Han. Oriented R-CNN for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3520–3529, 2021.

Yunkai Yang, Yudong Zhang, Kunquan Zhang, Jinxiao Zhang, Xinying Chen, Haohuan Fu, and Runmin Dong. Task-oriented data synthesis and control-rectify sampling for remote sensing semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42147–42157, 2026.

Zhengyuan Yang, Jianfeng Wang, Zhe Gan, Linjie Li, Kevin Lin, Chenfei Wu, Nan Duan, Zicheng Liu, Ce Liu, and Lijuan Wang. ReCo: Region-controlled text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14246–14255, 2023.

Ziqi Ye, Shuran Ma, Jie Yang, Xiaoyi Yang, Yi Yang, Ziyang Gong, Xue Yang, and Haipeng Wang. Object fidelity diffusion for remote sensing image generation. In *International Conference on Learning Representations*, 2026.

Mu Zhang, Yunfan Liu, Yue Liu, Yuzhong Zhao, and Qixiang Ye. CC-Diff++: Spatially controllable text-to-image synthesis for remote sensing with enhanced contextual coherence. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–16, 2025. doi: 10.1109/TGRS.2025.3616376.

Bo Zhao, Lili Meng, Weidong Yin, and Leonid Sigal. Image generation from layout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8584–8593, 2019.

Zhenghui Zhao, Chen Wu, Xiangyong Cao, Di Wang, Hongruixuan Chen, Datao Tang, Liangpei Zhang, and Zhuo Zheng. ChangeBridge: Spatiotemporal image generation with multimodal controls for remote sensing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27761–27771, 2026.

Guangcong Zheng, Xianpan Zhou, Xuewei Li, Zhongang Qi, Ying Shan, and Xi Li. LayoutDiffusion: Controllable diffusion model for layout-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22432–22443, 2023.

Kai Zou, Hongbo Liu, Dian Zheng, Jianxiong Gao, Zhiwei Zhao, and Bin Liu. EchoGen: Cycle-consistent learning for unified layout-image generation and understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 14068–14076, 2026.

The appendix follows the order in which a reader needs the material. Appendix A defines the metric behind Table 4. Appendix B reports per-class detection under the synthetic-only protocol of Table 3. Appendix C compares generators on dense DIOR-R layouts. Appendix D reports detector training when synthetic images augment real data. Appendix E specifies the bank and the retrieval score. Appendix F records masked attention, prompts, and the pairwise graph. Appendix G explains the LoRA choice. Appendix H states which factors Table 5 does and does not isolate.

A Relation-Eval protocol

Relation-Eval is a post-hoc score on generated pixels. Every method, and the real image, is evaluated on the same oriented layouts. The scorer never reads a generator's internal graph, edge features, or bank rows. A detector trained only on real images is frozen and shared across methods. Terrain labels come from a frozen CLIP land-cover head (ViT-L/14 at 336 px when that checkpoint is available, otherwise ViT-B/32) with a fixed prompt list. Allow-sets are frozen before any method is compared and are not retuned per generator.

Matching and the three relation scores do not use category in the same way. The assignment ignores category and keeps a detection only when its oriented box overlaps a layout box. OTAcc then asks whether the terrain around that layout box is allowed for the *layout* category. RelCooc and RelF1 ask whether the *detected* categories of matched boxes co-occur as the input layout does. A correctly placed box with the wrong detected class can therefore raise AvgBox and still lower RelCooc and RelF1.

Matching. Detections are matched to layout boxes by the Hungarian algorithm. The cost of pairing layout box i with detection j is $1 - u_{ij}$, where u_{ij} is the intersection-over-union of the two oriented boxes, approximated by rasterizing both polygons on a local canvas. A pair is kept only when $u_{ij} \geq 0.3$. Category agreement is not required for a match. Algorithm 1 states the rule.

Algorithm 1 Match detections to one layout.

Require: Layout boxes $\{\mathbf{b}_i\}_{i=1}^{N}$, detections $\{\hat{\mathbf{b}}_j\}_{j=1}^{M}$, threshold $\delta = 0.3$

1: Rasterize each pair and set $u_{ij} \leftarrow \mathrm{IoU}(\mathbf{b}_i, \hat{\mathbf{b}}_j)$
2: Solve a one-to-one assignment with cost $1 - u_{ij}$
3: **return** pairs (i, j) with $u_{ij} \geq \delta$

OTAcc. For each layout instance, CLIP is applied to the ring outside the oriented box, not to the box interior. Interior pixels are dominated by the object itself and confound terrain. The ring is the set of pixels within 32 px of the box and outside it. The predicted terrain is the land-cover label with the highest CLIP score. The instance is a hit when that label lies in the allow-set $T(c)$ of its layout category. OTAcc is the micro-average of these hits over instances. On DIOR-R the land-cover vocabulary is water, harbor, road, parking, bare soil, vegetation, built-up, runway, apron, sports field, and agriculture. Table 6 lists $T(c)$. DOTA uses the same ring, the same CLIP head, and the same matching constants; its allow-sets are corpus-specific, fixed before comparison, and are not listed here.

RelCooc, RelF1, and AvgBox. Let n_m be the number of matched boxes in an image. AvgBox is the mean of n_m over images. If $n_m < 2$, that image contributes 0 to both RelCooc and RelF1. This

Table 6: Frozen DIOR-R CLIP prompt allow-sets $T(c)$. An instance is a hit when the highest-scoring ring label lies in this set. The vocabulary is a fixed prompt list for scoring, not a land-cover taxonomy.

Category	Allowed terrain
airplane, airport	runway, apron
baseball field, basketball court, ground track, stadium, tennis court	sports field
bridge	water, road
chimney	built-up, bare soil
dam	water
expressway service area, toll station	road, parking
golf course	vegetation, sports field
harbor, ship	water, harbor
overpass	road
storage tank	bare soil, parking, built-up
train station	built-up, parking, road
vehicle	road, parking
windmill	vegetation, agriculture, bare soil

Algorithm 2 Relation-Eval on one image.

Require: Image, layout, frozen detector, CLIP, allow-sets T, prior M, $k = 4$, $\tau = 0.05$
Match detections to the layout (Algorithm 1); let n_m be the number of pairs
OTAcc hits: for each layout box, score the 32 px outer ring and test the argmax label against $T(c)$
if $n_m < 2$ **then**
  RelCooc $\leftarrow 0$, RelF1 $\leftarrow 0$
else
  RelCooc $\leftarrow$ mean M on kNN edges among matched detections
  Build reference edges on the input layout and predicted edges on matched nodes
  if both edge sets are empty **then**
    RelF1 $\leftarrow 1$
  else
    RelF1 $\leftarrow$ F1 of the two undirected edge sets
  end if
end if
return OTAcc hits, RelCooc, RelF1, n_m

rule is applied first. A generator receives a zero neighbor score from a single matched box, and the empty-edge convention below applies only when $n_m \geq 2$.

When $n_m \geq 2$, RelCooc is the mean training co-occurrence M_{c_i,c_j} on the k=4 nearest neighbors among the matched detections, using detected categories and detected centers. The dataset score is the mean of the image scores, including the zeros from images with $n_m < 2$.

RelF1 compares two undirected edge sets on images with $n_m \geq 2$. Reference edges are the k=4 nearest-neighbor pairs of the input layout whose co-occurrence satisfies $\max(M_{ij}, M_{ji}) \geq 0.05$. Predicted edges are rebuilt with the same k and the same threshold on the matched nodes, using detected categories and detected centers. Precision, recall, and F1 are the usual set scores. If both sets are empty, F1 is 1; this case is distinct from $n_m < 2$, which remains 0. RelF1 is the mean over all images. The training graph in Section 3.3 also keeps every same-category pair. The metric does not: an edge enters the reference set only when it passes the threshold above. Algorithm 2 assembles one image.

These scores are not a copy of the conditioning edges. OTAcc reads pixels outside the prescribed boxes and scores them against the layout category. RelCooc and RelF1 read categories from the frozen detector. A method that ignores the layout matches few boxes and is then driven toward zero by the $n_m < 2$ rule.



Figure 6: Per-class detection under the synthetic-only protocol of Table 3. Top: Oriented R-CNN on OBB. Bottom: YOLOv8 on HBB.



Figure 7: Qualitative comparison on dense DIOR-R layouts under identical oriented boxes. The last column is the corresponding real image.

B PER-CLASS DETECTION UNDER SYNTHETIC-ONLY TRAINING

Table 3 reports mean detection scores when Oriented R-CNN and YOLOv8 are trained only on synthetic images. Figure 6 expands that protocol to per-class average precision: the upper panel is Oriented R-CNN on oriented boxes, and the lower panel is YOLOv8 on horizontal boxes. The figure does not invent a new training setup. It shows whether the mean gains in Table 3 are concentrated on a few categories or appear across the category list.

C DENSE DIOR-R LAYOUTS

Figure 4 uses representative DIOR-R layouts. Figure 7 repeats the same generator set on denser scenes, where many oriented boxes share a small region. Columns are the input layout, LayoutDiff (LayoutDiffusion), GLIGEN, CC-Diff, AeroGen, OF-Diff, LayGraph, and the corresponding real image. The comparison isolates packing failures that mean FID and mAP do not show: missing instances, broken alignments, and terrain that does not match the prescribed boxes.

D DOWNSTREAM MIXING AND AUGMENTATION

Table 3 trains detectors on synthetic images alone. Tables 7 and 8 ask whether the same images still help when real labeled data are available. The two protocols use different baseline sets, so each ranking stays inside its own table. Table 7 compares Flip, CopyPaste, AeroGen, and LayGraph

Table 7: DIOR-R augmentation under YOLOv8 with 50k synthetic images (or classical augmentations). Best and second-best in bold and underlined.

Strategy	mAP↑	mAP_{50} ↑
Flip	37.39	60.21
CopyPaste	38.25	61.79
AeroGen	41.32	63.98
LayGraph	<u>42.15</u>	<u>64.85</u>
CopyPaste+Flip	38.75	62.11
AeroGen+Flip	41.69	64.12
LayGraph+Flip	**42.58**	**65.37**

Table 8: YOLOv8 after mixing 10k synthetic images into real DIOR and DOTA training sets. Best and second-best among augmented rows in bold and underlined. Real is the unaugmented baseline.

	DIOR				DOTA			
Setting	mAP_{50}	AP_s	AP_l	mAP	mAP_{50}	AP_s	AP_l	mAP
Real	72.13	38.17	68.36	52.28	66.28	31.47	58.16	38.07
Real+AeroGen	73.82	39.46	70.37	53.67	67.68	32.66	59.67	39.18
Real+CC-Diff	74.18	40.07	71.28	54.13	68.07	33.18	60.46	39.62
Real+OF-Diff	<u>74.67</u>	<u>40.58</u>	<u>72.16</u>	<u>54.52</u>	<u>68.53</u>	<u>33.87</u>	<u>61.37</u>	<u>40.08</u>
Real+LayGraph	**76.83**	**43.17**	**75.08**	**57.12**	**71.36**	**36.47**	**65.18**	**42.76**

as DIOR-R augmentation under YOLOv8, including combinations with Flip, using 50k synthetic images. Table 8 mixes 10k synthetic images from AeroGen, CC-Diff, OF-Diff, or LayGraph into the real DIOR and DOTA training sets and evaluates YOLOv8. In Table 7, LayGraph with Flip is highest and LayGraph without Flip is second. In Table 8, LayGraph is highest on every reported column and OF-Diff is second. Real is an unaugmented baseline and is not marked as best or second-best.

E GEOMETRY FEATURES BANK AND CLUSTER RETRIEVAL

This section specifies the bank behind Eq. equation 6 and the score behind Eq. equation 14. Training and layout-only inference share the same descriptor slots. They differ only in which row fills each slot.

What is stored. Each training crop writes one instance row. The row is the concatenation $\mathbf{a}_i = [\mathbf{v}_i; \mathbf{f}_i; \mathbf{t}_i]$ with dimensions 384, 384, and 256 (1024 in total): a CLIP visual feature, a spectral feature from color statistics and the crop spectrum, and a text feature of category, size, and heading. The scene row $\mathbf{s}$ has dimension 480 and is computed on the region outside all labeled boxes, again from visual and spectral features. At training time the loader reads the row whose image identifier matches the source image. No retrieval is used.

Geometry bins. Every instance row also stores three discrete bins, used later as the geometry term g. Area is small, medium, or large according to the ratio of box area to a category reference: below 0.6, above 1.4, or in between. Aspect is elongated when the axis-aligned aspect ratio is above 1.8 or below 0.55, and compact otherwise. Heading is one of eight 45° bins. The continuous geometry vector used for clustering is

$$\big[\, a,\ \log r,\ \sin\theta,\ \cos\theta,\ c_x,\ c_y \,\big],$$

where a is the area ratio, r is the aspect ratio, θ is the heading, and (c_x, c_y) is the box center normalized by the 512×512 frame.

Offline clusters. Instance rows are clustered independently per category. The vector is the concatenation of the ℓ_2-normalized descriptor, scaled by 0.85, and the ℓ_2-normalized geometry vector, scaled by 0.15. k-means uses k=8 centers per category, at most 50 iterations, and a fixed seed. Scene rows are clustered once, with k=48, on a layout signature. That signature concatenates an ℓ_1-normalized class histogram with four scalars: the instance count, capped at 20 and divided by 10; mean box area; background coverage; and histogram entropy divided by 3. Cluster construction does not pass boxes, so the mean-area coordinate is zero and coverage is the value stored with the scene row. When instance descriptors exist, their mean visual block is appended after the histogram block is scaled by

Algorithm 3 Build the geometry features bank.

Require: Training images, oriented annotations, frozen CLIP
1: **for** each annotated instance **do**
2: Crop the box; store $\mathbf{a} = [\mathbf{v}; \mathbf{f}; \mathbf{t}]$ and the area, aspect, and heading bins
3: **end for**
4: **for** each image **do**
5: Store $\mathbf{s}$ from pixels outside all boxes
6: **end for**
7: Per category, k-means (k=8) on the joint descriptor–geometry vector
8: k-means (k=48) on scene layout signatures
9: **return** instance table, scene table, and both sets of centers

Algorithm 4 Retrieve bank rows from a layout.

Require: Layout $\{(\mathbf{b}_i, c_i)\}$, bank, subtype centers, scene centers, prior M
1: Build the scene signature with mean area 0 and coverage clipped to $[0, 1]$ as $1 - \sum_i a_i$
2: Keep the 3 nearest scene clusters; pick $\mathbf{s}$ by signature similarity plus co-occurrence
3: **for** each instance i **do**
4: Set q_i to the stored row of c_i with the best bin agreement
5: Keep the 3 nearest subtype centers of the joint descriptor–geometry vector
6: Pick $\mathbf{a}_i$ by Eq. equation 15 inside those subtypes
7: **end for**
8: **return** $\{\mathbf{a}_i\}$ and $\mathbf{s}$

0.75 and the visual block by 0.25. Centers and membership are written once and reused for every reported run. Algorithm 3 summarizes the build.

Layout-only retrieval. Given only a layout, the query descriptor q_i for instance i is the stored row of category c_i whose bins agree most with $\mathbf{b}_i$. Area agreement and aspect agreement each add 2, and heading agreement adds 1. Subtype assignment does not compare q_i with the centers by itself. It compares the same joint vector used to build the centers, 0.85 times the normalized descriptor concatenated with 0.15 times the normalized geometry vector, and keeps the three nearest of the eight centers.

Rows inside those three subtypes are ranked by the candidate-dependent score

$$S(q, v) = \mathrm{sim}(q, v) + 0.05\, g(q, v) + 0.20\, \bar{M}_{\mathrm{scene}}(q, v). \tag{15}$$

Here sim is the cosine similarity between q_i and the stored descriptor, and $g = 2 \cdot \mathbf{1}_{\mathrm{area}} + 2 \cdot \mathbf{1}_{\mathrm{aspect}} + 1 \cdot \mathbf{1}_{\mathrm{heading}}$. $\bar{M}_{\mathrm{scene}}$ is the mean training co-occurrence between the four spatial neighbors of instance i and the categories in the training image that owns row v. A second term, $0.25\, \bar{M}_{\mathrm{layout}}$, is added to every candidate of the same query. It is the mean co-occurrence between c_i and those same neighbors. The term is identical for every candidate of one query, so the selected row is determined by Eq. equation 15. Equation equation 14 in the main text is this candidate-dependent score: descriptor similarity, geometry-bin agreement, and co-occurrence with the bank image.

Scene retrieval uses a separate query. The signature matches the one used to build the scene clusters, including a zero mean-area coordinate. Coverage is one minus the sum of the query box-area ratios, clipped to $[0, 1]$. The three nearest of the 48 centers are kept, and rows inside them are ranked by cosine similarity of signatures plus 0.30 times the mean co-occurrence between the query categories and the categories of the bank image. The chosen $\mathbf{a}_i$ and $\mathbf{s}$ occupy the bank slots for the whole denoising trajectory. Algorithm 4 states both lookups.

F Additional architectural details

Instance-masked layout attention. Let $\mathbf{M} \in \{0, 1\}^{P \times N}$ indicate whether spatial token p falls inside the oriented mask of instance i. Masks are 64×64 and are mapped onto the attention grid by adaptive max pooling, then thresholded at 0.5. Logits for instance keys at uncovered locations are set to $-\infty$. A row covered by no instance is set to zero after the softmax, so empty terrain is not attended as a union of every instance. Text and background streams cover those locations.

Table 9: Bank and retrieval constants used for all reported runs. The layout co-occurrence offset is identical for every candidate of one query and does not change which row is selected.

Constant	Value
Instance descriptor	$384 + 384 + 256 = 1024$
Scene descriptor	480
Subtype clusters per category	8
Scene clusters	48
Subtype / scene shortlist	top 3
Neighbor count in $\bar{M}$	k=4
Geometry weight α	0.05
Layout co-occurrence offset	0.25, same for every candidate of one query
Bank-scene co-occurrence weight	0.20
Scene-retrieval co-occurrence weight	0.30
Bin score g	area 2, aspect 2, heading 1

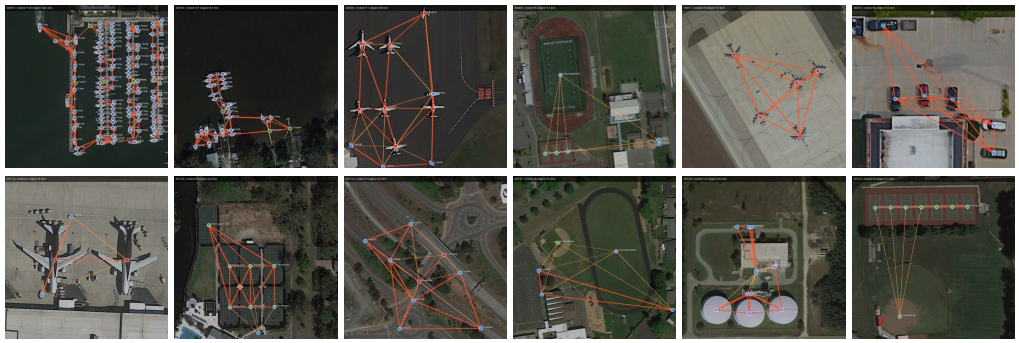


Figure 8: Relation-graph visualization for one oriented layout: instance nodes, kNN edges, and the background super-node.

Spatial gates. The outside map is the complement of the instance masks, resized to the attention grid by nearest-neighbor interpolation. Background outputs are multiplied by this map, so scene tokens do not write inside boxes. Text cross-attention is unchanged outside boxes. Inside boxes it is multiplied by $1 - s$, where s is the layout warmup in Eq. equation 12. Early in training s is small, so the text stream stays nearly intact while layout control is introduced.

Prompting. Training prompts follow a fixed template: an aerial-image phrase from the classes present, up to three style phrases, size and location mentions for at most eight instances (a median-area split and 3×3 location buckets), and a count and mix statistic, truncated to 65 words. Classifier-free dropout uses three branches, with probabilities 0.10, 0.05, and 0.05: drop the text, drop the layout and bank descriptors, or drop both. On a text-only drop, the prompt is a fixed negative prompt with probability 0.5 and the empty string otherwise. A layout drop replaces instance and scene descriptors by stored null vectors and zeroes the boxes and masks.

What the graph is. The background super-node and the edges that touch it form a star. Each instance is tied to the scene, and instances are tied to each other only by the sparse kNN graph in Section 3.3. A single hyperedge over all instances and the scene would induce a clique, the 2-section of that hyperedge, and would spend the relation-token budget on distant pairs. The binder in Eq. equation 9 stays pairwise. Figure 8 shows one layout with those edges drawn. The drawing is the conditioning graph used by the generator. Relation-Eval in Appendix A scores generated pixels with a separate detector and CLIP head.

G Full-parameter fine-tuning

The reported model updates rank-8 LoRA adapters, with scaling 16, on the spatial-transformer attention projections of Stable Diffusion 1.5. The layout, relation, and scene modules are trained in

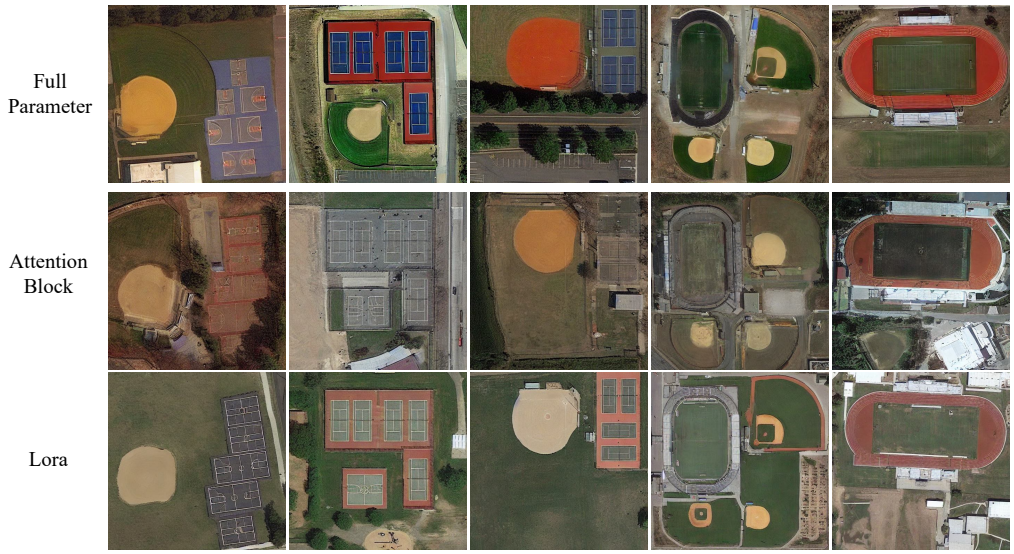


Figure 9: Full-parameter update of the spatial-transformer attention projections versus the rank-8 LoRA adapters used in the reported model.

full. The comparison here uses the same data, loss, and schedule, and instead updates those attention projections without LoRA. Figure 9 contrasts a full update of the attention block with the LoRA adapters used in the paper.

On matched layouts, the full update shifts object hues and background contrast away from the aerial images, while boxes can remain roughly in place. The remote-sensing corpus is much smaller than the data on which Stable Diffusion 1.5 was pretrained. A plausible account is that a full update of the attention weights fits the color statistics of this smaller set. Separating color statistics from other effects of updating every weight would require a further control. LoRA leaves the pretrained weights frozen and learns a low-rank increment, which is the setting used for the numbers in the paper.

H Scope of the reported ablation

Table 5 varies four factors: the relation graph, the background super-node, the geometry features bank, and cluster retrieval. Those are the comparisons this paper reports. Relation tokens and the layout-attention bias belong to the relation graph. The scene summary belongs to the background super-node. Adaptive instance and background weights, the edge loss, and leaving instance message passing off are part of the default graph in Section 3.3. Replacing bank descriptors by class-name embeddings is the same comparison as removing the geometry features bank, which the table already reports.