SPATIALLY STRUCTURED CODEBOOK LEARNING FOR 3D INSTANCE-LEVEL PERCEPTION OF AERIAL-GROUND SCENES

Anonymous authors
Paper under double-blind review

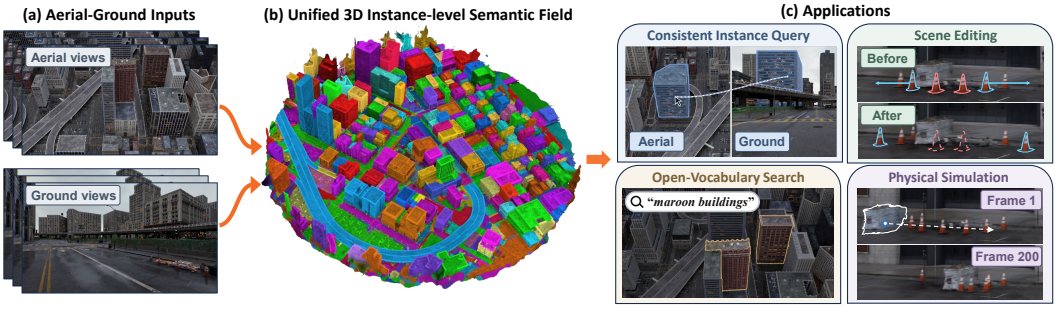

Figure 1: Given aerial and ground observations (a), our method constructs a unified 3D instance-level semantic field (b) that combines globally consistent instance identities with language-grounded semantics and can support applications including consistent cross-view instance querying, open-vocabulary search, scene editing, and physical simulation (c).

ABSTRACT

Recent advances in unified aerial–ground reconstruction have enabled photorealistic modeling of large-scale environments, yet fine-grained scene understanding remains underexplored. Achieving consistent instance-level semantics is particularly challenging due to sparse per-view observations over a large identity space and severe cross-view variations in projected scale and visible detail. These factors complicate scene-wide optimization and lead to over- or under-segmented 2D instance masks with inconsistent supervision. To address these challenges, we propose Spatially Structured Codebook Learning (SSCL), a scalable framework for instance-level aerial–ground scene understanding. SSCL first organizes the scene into spatial root regions, each with a compact codebook of leaves representing local instance identities, and jointly learns cross-root connectivity to fuse leaves into globally consistent instances. A granularity-aware codebook learning method is further introduced to heuristically mitigate conflicting instance supervision. We further introduce Aerial–Ground Segmentation (AG-Seg), a new benchmark dataset for evaluating instance segmentation and open-vocabulary retrieval in aerial–ground scenes. Extensive experiments on AG-Seg demonstrate substantial improvements over state-of-the-art methods in both tasks, enabling fine-grained interaction with large aerial–ground environments.

1 INTRODUCTION

Unified aerial–ground scene modeling has gained increasing attention for its ability to integrate complementary observations across altitudes and produce more complete 3D representations. Such comprehensive scene models support a wide range of applications, including real-to-sim simulation, autonomous navigation, and immersive VR/AR. Recent studies (Zhang et al., 2024; Jiang et al., 2025; Wang et al., 2026a) have primarily focused on unified aerial-ground reconstruction with Gaussian splatting representations (Kerbl et al., 2023; Lu et al., 2024; Ren et al., 2025), enabling high-fidelity

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.

novel-view synthesis. However, recovering fine-grained, instance-level semantics remains largely unexplored, which severely limits meaningful interaction with reconstructed environments. In this work, our objective is to endow reconstructed aerial–ground scenes with fine-grained 3D scene understanding, enabling individual scene entities to be consistently identified and semantically interpreted in 3D. This extends aerial–ground modeling beyond photorealistic reconstruction toward machine-interpretable 3D environments.

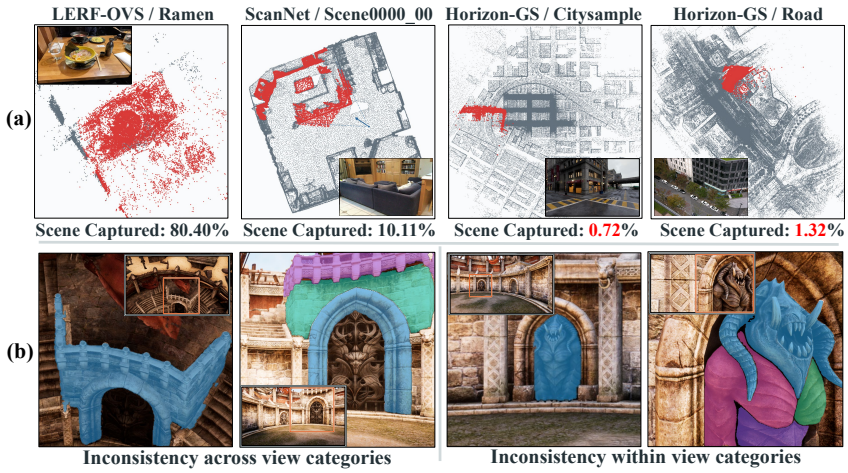

Figure 2: **Challenges in aerial–ground semantic modeling.** (a) **Sparse instance observability:** each view covers a small region (red, with percentages denoting observed/total Gaussians), providing sparse supervision over a large identity space. (b) **Cross-view granularity inconsistency:** scale and detail variations induce mask merging and fragmentation.

Compared with indoor scenes (Dai et al., 2017; Yeshwanth et al., 2023) or relatively compact environments (Kerr et al., 2023; Liu et al., 2023), unified aerial–ground scenes pose two pronounced challenges for 3D scene understanding (Fig. 2). First, **sparse instance observability** arises from the large number of scene instances and limited per-view coverage, making scene-wide instance representations difficult to optimize. Second, **cross-view granularity inconsistency** arises from the much larger variation in projected object scale than in typical indoor scenes, causing the same instance to appear merged or fragmented across views.

Although existing methods for 3D Gaussian scene understanding achieve promising results in indoor or relatively compact environments, the above challenges can limit their effectiveness when directly applied to aerial–ground scenes. Early methods compress high-dimensional language features to reduce storage and rendering costs (Qin et al., 2024; Zhou et al., 2024), at the expense of semantic fidelity. Subsequent methods learn instance features through contrastive learning and recover identities by clustering (Cen et al., 2025; Li et al., 2025a; Wang et al., 2025). Such grouping is sensitive to feature noise, similarity metrics, and clustering hyperparameters. OpenGaussian (Wu et al., 2024) further uses a two-level codebook but still requires additional non-differentiable feature clustering for identity recovery and lacks cross-cluster identity fusion, leaving objects spanning coarse clusters without a unified identity. More recent approaches introduce learnable global codebooks to explicitly structure scene-level semantics during optimization (Qu et al., 2024; Li et al., 2025b; Li et al., 2026a). Yet, in large aerial–ground scenes, each view covers only a limited region, leaving most codebook entries inactive and sparsely supervised during each update and making scene-wide organization difficult to learn. Meanwhile, variations in scale and visible detail can merge or fragment masks across views, introducing conflicting instance assignments that the structured representation itself cannot resolve.

To address these challenges, we propose a Spatially Structured Codebook Learning (SSCL) framework for aerial-ground scenes understanding. Instead of maintaining a scene-wide global codebook, SSCL partitions the Gaussian scene into spatial root regions and learns a compact local codebook of leaves representing local instance identities within each region, substantially reducing the identity space involved in local optimization. A learned cross-root connectivity mechanism then associates corresponding leaves across regions to recover globally consistent 3D instances. To handle inconsistent mask granularity across views, we further introduce a granularity-aware codebook learning method, which explicitly accounts for merged and fragmented masks during both mask-to-leaf as-

signment and supervision. Finally, we introduce Aerial–Ground Segmentation (AG-Seg), a new benchmark dataset for evaluating instance segmentation and open-vocabulary retrieval in aerial–ground scenes. We build AG-Seg by augmenting seven scenes from Horizon-GS (Jiang et al., 2025) and UC-GS (Zhang et al., 2024) with per-image instance mask annotations and natural-language queries spanning both aerial and ground views. The benchmark supports complementary evaluation protocols: click-based selection evaluates the segmentation of individual instances, while text-based retrieval evaluates regions matching open-vocabulary queries. Experiments on AG-Seg demonstrate substantial improvements over state-of-the-art methods in both tasks.

In summary, the contributions of our methods are listed below:

- We introduce Spatially Structured Codebook Learning (SSCL), a novel framework that combines compact local codebooks with learned cross-root connectivity to recover globally consistent 3D instances in large aerial–ground scenes.
- We develop a granularity-aware codebook learning method to heuristically mitigate conflicting instance supervision caused by cross-view segmentation-granularity variations.
- We introduce AG-Seg, the first benchmark dataset for both instance segmentation and open-vocabulary retrieval in unified aerial–ground scenes, to the best of our knowledge. Extensive experiments demonstrate substantial gains over existing baselines across both tasks.

2 Related Work

2.1 Aerial-Ground Scene Reconstruction With Gaussian Splatting

3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) has driven rapid progress in urban scene modeling by combining high-fidelity rendering with real-time performance. Recent work improves large-scale reconstruction (Xue et al., 2026), with methods tailored to aerial imagery (Li et al., 2026b; Gao et al., 2026) or driving scenes (Wang et al., 2026b; Shimomura et al., 2026; Chen & Peng, 2026). Complementing these efforts, recent systems extend 3DGS to unified aerial–ground reconstruction with cross-altitude observations. UC-GS (Zhang et al., 2024) uses aerial imagery to assist road-view reconstruction, Horizon-GS (Jiang et al., 2025) adopts staged optimization, and Urban-GS (Wang et al., 2026a) combines aerial–street adaptive densification with anchor pruning and global-to-local refinement. Despite these advances, existing aerial–ground methods remain centered on photometric reconstruction and novel-view synthesis, leaving instance-level semantic understanding largely unexplored.

2.2 3D Scene Understanding with Gaussian Splatting

To enable instance-level semantic retrieval in 3D scenes, recent methods transfer semantic knowledge from pretrained 2D vision and vision-language models (Radford et al., 2021; Kirillov et al., 2023) into Gaussian representations, supporting language-based querying, localization, and segmentation. We organize these approaches along four non-exclusive design axes.

Feature Distillation and Compression. High-dimensional semantic features incur substantial storage and rendering costs when attached to individual Gaussians. LangSplat (Qin et al., 2024) compresses SAM-masked CLIP features with a scene-specific autoencoder, Feature 3DGS (Zhou et al., 2024) stores low-dimensional distilled features and reconstructs dense feature maps after rendering, and GAGS (Peng et al., 2026) decodes compact Gaussian features into the language space through an MLP. Although efficient, compression can weaken fine-grained semantic distinctions. COS3D (Zhu et al., 2025b) instead couples instance and language fields through bidirectional interaction, whereas our method separates instance structure learning from language grounding.

Feature Aggregation and Temporal Association. Dr. Splat (Kim et al., 2025), Occam's LGS (Cheng et al., 2025), VALA (Wang et al., 2026c), and SLAG (Szilágyi et al., 2025) aggregate view-level language features using visibility or rendering contributions, while Di-LEGS (Li et al., 2026c) directly optimizes a sparse semantic Gaussian field from synchronized masks. Without explicit association or conflict resolution, inconsistent view-level semantics may propagate into the resulting 3D field. Beyond feature aggregation, ObjectGS (Zhu et al., 2025a) and

ExtrinSplat (Ding et al., 2026) propagate masks and identities across temporally adjacent frames. Such sequential cues cannot be assumed across heterogeneous aerial–ground cameras.

Feature Clustering. Feature clustering methods recover object structures by grouping compact semantic or identity features learned on Gaussians. SAGA (Cen et al., 2025) learns scale-aware identity features from multi-view SAM masks, while SuperGSeg (Liang et al., 2026) reduces redundancy through semantically coherent super-primitives with shared embeddings. OpenGaussian (Wu et al., 2024) recovers identities through additional non-differentiable feature clustering in a position-aware two-level codebook, but lacks cross-cluster identity fusion, leaving objects spanning coarse clusters without a unified instance identity. AG^{2}aussian (Wang et al., 2025) propagates features over an anchor graph, while InstanceGaussian (Li et al., 2025a) forms instance groups through feature sampling and connected-component analysis. These methods infer instance structure primarily from embedding similarity and grouping rules. Our method instead supervises cross-root identity associations with mask-derived evidence, reducing dependence on globally consistent embeddings.

Codebook-based Methods. Codebook-based methods organize scene semantics with globally shared entries that encode object identities, semantic prototypes, or language-feature bases. GOI (Qu et al., 2024) learns a feature-clustering codebook with an optimizable semantic hyperplane, while Unified-Lift (Zhu et al., 2025c) jointly optimizes Gaussian features and an object-level codebook. LangSplatV2 (Li et al., 2025b) represents language features with a global dictionary, whereas GALA (Alegret et al., 2026) and OpenGaFF (Li et al., 2026a) employ attention-based codebooks to improve semantic consistency. Despite their different entry semantics, these approaches retain scene-wide codebooks: identity codebooks must cover more objects as scenes grow, while language dictionaries must span increasingly diverse CLIP semantics. Because each view exposes only a local subset of either space, codebook activation and supervision become sparse and uneven, complicating global optimization. Our method distributes capacity across compact local codebooks and recovers scene-level identities through learned cross-root association.

3 METHOD

Given posed aerial–ground images $\{I^v\}_{v\in\mathbb{V}}$, where $\mathbb{V}$ indexes the views, our objective is to distill semantic cues from 2D foundation models (Kirillov et al., 2023; Radford et al., 2021) into a Gaussian scene reconstructed by unified aerial–ground methods such as Urban-GS (Wang et al., 2026a), enabling view-consistent 3D instance segmentation for interactive selection and open-vocabulary retrieval. Figure 3 outlines our Spatially Structured Codebook Learning (SSCL) framework. We first partition the scene into local regions, each equipped with a compact codebook of leaves (Sec. 3.1). Granularity-aware codebook learning mitigates supervision conflicts from multi-view mask fragmentation and merging (Sec. 3.2). Concurrently, we learn cross-root connectivity to associate corresponding leaves across regions and recover globally consistent 3D instances (Sec. 3.3).

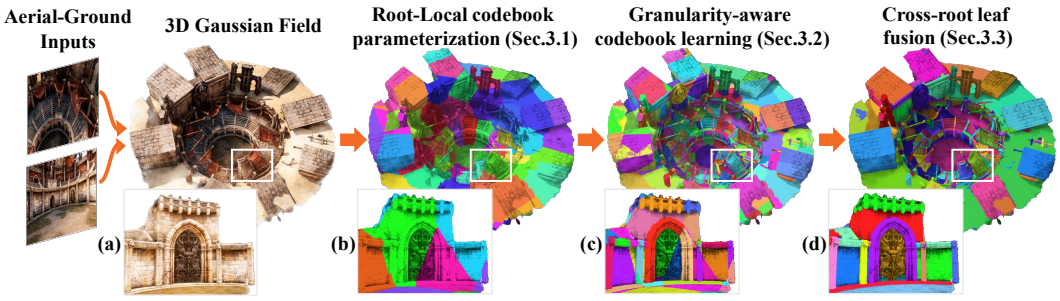

Figure 3: **Overview of the Spatially Structured Codebook Learning (SSCL) pipeline.** Starting from a reconstructed 3D Gaussian scene (a), we build a *root-local instance representation* by partitioning scene anchors into spatial roots with compact local codebooks (b). *Granularity-aware codebook learning* then reconciles inconsistent multi-view mask granularities to obtain reliable leaf assignments (c). Finally, *cross-root leaf fusion* combines learned connectivity with spatial adjacency to fuse leaves into coherent scene-level instances (d). The insets highlight local changes across processing stages.

3.1 ROOT-LOCAL INSTANCE REPRESENTATION

SSCL builds a root-local instance representation on the reconstructed 3D Gaussian scene. We first specify the anchor-based backbone and then introduce the root-local parameterization and root-aware rendering used by the subsequent mask-supervision and cross-root association stages.

Neural 3DGS backbone. Following previous work (Jiang et al., 2025; Wang et al., 2026a), we adopt the anchor-based neural Gaussian representation of Scaffold-GS (Lu et al., 2024) as our 3DGS backbone. Each anchor i stores a 3D position $\boldsymbol{x}_i$, a feature vector $\boldsymbol{f}_i$, a scaling vector $\boldsymbol{s}_i$, and K learnable offsets $\{\boldsymbol{o}_{i,k}\}_{k=1}^{K}$ that define its local structure. It generates the centers of its spawned neural Gaussians as

$$\boldsymbol{\mu}_{i,k} = \boldsymbol{x}_i + \boldsymbol{s}_i \odot \boldsymbol{o}_{i,k}, \qquad k = 1, \ldots, K, \tag{1}$$

where $\odot$ denotes element-wise multiplication. Other attribute parameters—such as opacities, scales, rotations, and view-dependent colors—are decoded from the anchor feature using attribute-specific MLPs. In particular, the view-dependent color of the k-th Gaussian is decoded as $\boldsymbol{c}_{i,k}^{v} = F_{\text{color}}(\boldsymbol{f}_i, \boldsymbol{o}_{i,k}, \boldsymbol{d}_i^{v})$, where $\boldsymbol{d}_i^{v}$ is the viewing direction from anchor i to the camera in view v. During rendering, all spawned Gaussians across anchors are flattened into a single sequence and sorted by depth. We compute the rendered RGB color $\widehat{I}^{v}(\boldsymbol{u})$ at pixel $\boldsymbol{u}$ via standard alpha compositing:

$$\widehat{I}^{v}(\boldsymbol{u}) = \sum_{j \in \mathbb{J}^{v}(\boldsymbol{u})} T_j^{v}(\boldsymbol{u}) \alpha_j^{v}(\boldsymbol{u}) \boldsymbol{c}_j^{v}, \qquad T_j^{v}(\boldsymbol{u}) = \prod_{q=1}^{j-1} \left(1 - \alpha_q^{v}(\boldsymbol{u})\right), \tag{2}$$

Here, $\mathbb{J}^{v}(\boldsymbol{u})$ indexes the overlapping Gaussians in depth order. For Gaussian j, $\boldsymbol{c}_j^{v}$ is its RGB vector, $\alpha_j^{v}(\boldsymbol{u})$ its opacity at $\boldsymbol{u}$, and $T_j^{v}(\boldsymbol{u})$ the accumulated transmittance.

Root-local codebook parameterization. We partition scene anchors into R spatial roots using MiniBatch K-means, assigning each anchor i a root index $r_i \in \{0, \ldots, R-1\}$. Each root maintains a codebook of L leaves representing local instance identities. Anchor i has a learnable logit vector $\boldsymbol{z}_i \in \mathbb{R}^{L}$ and probability vector $\boldsymbol{p}_i$, with scalar entries $z_{i,l}$ and $p_{i,l}$, respectively. These define a discrete leaf assignment $\widehat{l}_i$:

$$\boldsymbol{p}_i = \mathrm{softmax}(\boldsymbol{z}_i), \qquad \widehat{l}_i = \arg\max_{l} z_{i,l}. \tag{3}$$

Each spawned neural Gaussian inherits its parent anchor's root assignment $r_{\eta(j)}$ and probability distribution $\boldsymbol{p}_{\eta(j)}$, where $\eta(j)$ denotes the source anchor of Gaussian j. Although leaf indices $l \in \{0, \ldots, L-1\}$ are local to each root, we define $g(r, l) = rL + l$ to uniquely map them into a global index space for cross-root leaf association.

Root-aware feature rendering. Directly compositing leaf distributions via equation 2 causes boundary ambiguity as leaf indices l are reused locally. To eliminate cross-root index collisions, we restrict rendering at pixel $\boldsymbol{u}$ to its dominant root $\rho^{v}(\boldsymbol{u})$. Let $\mathbb{J}_r^{v}(\boldsymbol{u}) = \{j \in \mathbb{J}^{v}(\boldsymbol{u}) \mid r_{\eta(j)} = r\}$ index the Gaussians originating from root r. The dominant root is identified by cumulative opacity:

$$V_r^{v}(\boldsymbol{u}) = \sum_{j \in \mathbb{J}_r^{v}(\boldsymbol{u})} T_j^{v}(\boldsymbol{u}) \alpha_j^{v}(\boldsymbol{u}), \qquad \rho^{v}(\boldsymbol{u}) = \arg\max_{r} V_r^{v}(\boldsymbol{u}). \tag{4}$$

For global leaf ID $a = g(r, l)$, its rendered leaf response is

$$\widehat{P}_a^{v}(\boldsymbol{u}) = \mathbf{1}[r = \rho^{v}(\boldsymbol{u})] \sum_{j \in \mathbb{J}_r^{v}(\boldsymbol{u})} T_j^{v}(\boldsymbol{u}) \alpha_j^{v}(\boldsymbol{u}) p_{\eta(j),l}. \tag{5}$$

The indicator $\mathbf{1}[\cdot]$ retains only channels of the dominant root. These unnormalized responses satisfy $\sum_a \widehat{P}_a^{v}(\boldsymbol{u}) = V_{\rho^{v}(\boldsymbol{u})}^{v}(\boldsymbol{u})$. During validation and testing, we use top-k sparse leaf probabilities with $k = 4$ for rendering, following (Li et al., 2025b).

3.2 GRANULARITY-AWARE CODEBOOK LEARNING

Learning root-local codebooks requires associating 2D mask components with leaves through linear assignment (Siddiqui et al., 2023; Zhu et al., 2025c; Kuhn, 1955). In aerial–ground scenes, changes

in projected scale, visibility, and detail cause masks to merge instances or fragment a single instance across views. Our heuristic granularity-aware strategy mitigates the resulting supervision conflicts during assignment and learning.

Granularity-aware mask-to-leaf assignment. Let $\mathcal{M}^v$ be the collection of 2D instance masks in view v, each represented by its foreground pixel set m. All mask-dependent scores below refer to this view. Conventional matching averages each identity's rendered response within a mask (Siddiqui et al., 2023). Unified-Lift (Zhu et al., 2025c) removes area normalization to reduce small-segment sensitivity, accumulating responses instead:

$$\pi^*_{\mathrm{UL}} = \arg\max_{\pi} \sum_{m \in \mathcal{M}^v} \sum_{\boldsymbol{u} \in m} P^v_{\pi(m)}(\boldsymbol{u}), \tag{6}$$

where $\pi(m)$ denotes the instance identity assigned to mask m, and $P^v_{\pi(m)}(\boldsymbol{u})$ is its rendered scalar response at pixel $\boldsymbol{u}$. This shifts the bias from fragments to large masks. We instead balance purity and coverage for root-local mask–leaf matching. For each root r, we intersect the 2D masks $\mathcal{M}^v$ with its pixel support $\mathbb{U}^v_r = \{\boldsymbol{u} \mid \rho^v(\boldsymbol{u}) = r\}$ to obtain the non-empty root-local components $\mathcal{M}^v_r$. For a component $m_r \in \mathcal{M}^v_r$ and leaf l within root r, we define

$$s_{\mathrm{purity}}(m_r, l) = \frac{\sum_{\boldsymbol{u} \in m_r} \widehat{P}^v_{g(r,l)}(\boldsymbol{u})}{|m_r|}, \qquad s_{\mathrm{coverage}}(m_r, l) = \frac{\sum_{\boldsymbol{u} \in m_r} \widehat{P}^v_{g(r,l)}(\boldsymbol{u})}{\sum_{\boldsymbol{u} \in \mathbb{U}^v_r} \widehat{P}^v_{g(r,l)}(\boldsymbol{u}) + \epsilon}. \tag{7}$$

Here, $\epsilon > 0$ stabilizes denominators and logarithms. The score s_{purity} measures the mean leaf response within a component, while s_{coverage} measures the fraction of the leaf's rendered mass captured by that component. High s_{purity} suggests that the component predominantly corresponds to leaf l, while high s_{coverage} suggests that it captures that leaf's visible region more completely. We combine them into a matching score and define the final assignment as

$$S_{(m_r,l)} = \frac{2 s_{\mathrm{purity}}(m_r, l) s_{\mathrm{coverage}}(m_r, l)}{s_{\mathrm{purity}}(m_r, l) + s_{\mathrm{coverage}}(m_r, l) + \epsilon}, \qquad \mathbb{H}^{v,*}_r = \arg\max_{\mathbb{H}_r} \sum_{(m_r,l) \in \mathbb{H}_r} S_{(m_r,l)}. \tag{8}$$

Here, $\mathbb{H}_r$ is a set of component–leaf pairs, ranging over exclusive assignments between components not selected for fragment attachment below and the L local leaves. Each component and each leaf appears at most once in a Hungarian assignment.

Fragment attachment and purity-weighted supervision. Although the combined score reduces bias in mask-to-leaf correspondence evaluation, over-segmented 2D masks of the same instance introduce many-to-one correspondences that conflict with the one-to-one assignment. We introduce fragment attachment to associate their root-local components with a shared leaf before matching the remaining components. Specifically, components with high s_{purity} but low s_{coverage} are treated as candidate fragments. We directly attach m_r to leaf l when

$$s_{\mathrm{purity}}(m_r, l) \geq \tau_{\mathrm{high}} \quad \text{and} \quad s_{\mathrm{coverage}}(m_r, l) \leq \tau_{\mathrm{low}}. \tag{9}$$

The attached pairs form $\mathbb{F}^v_r$. Hungarian matching on the remaining components yields $\mathbb{H}^{v,*}_r$. The local supervision set is $\mathbb{A}^v_r = \mathbb{H}^{v,*}_r \cup \mathbb{F}^v_r$. Under-segmented masks typically exhibit low s_{purity}. Although discarding them removes ambiguous assignments, we empirically observe that this deprives the corresponding regions of necessary supervision. We instead retain merged masks and modulate their loss contribution via s_{purity}:

$$\mathcal{L}_{\mathrm{leaf}} = -\frac{1}{N_{\mathrm{asg}}} \sum_r \sum_{(m_r,l) \in \mathbb{A}^v_r} \mathrm{sg}[s_{\mathrm{purity}}(m_r, l)] \sum_{\boldsymbol{u} \in m_r} \log\Big(\widehat{P}^v_{g(r,l)}(\boldsymbol{u}) + \epsilon\Big). \tag{10}$$

Here, N_{asg} counts supervised pixels and sg stops gradients through the weight. Fragment attachment allows multiple components to supervise a common leaf, while purity weighting reduces the influence of low-response assignments.

3.3 Cross-Root Leaf Fusion

Recovering scene-level identities across roots requires a learnable cross-root connectivity matrix $\boldsymbol{C} \in [0,1]^{RL \times RL}$. For leaves $a = g(r,l)$ and $b = g(r',l')$ in different roots, the entry $C_{a,b} =$

$\mathrm{sigmoid}(\ell_{a,b})$ is their association score, with learnable scalar logit $\ell_{a,b}$. Connectivity supervision is constructed only from co-observed Hungarian matches in $\mathbb{H}_r^{v,*}$. For a matched leaf a, let $M_a^v \in \mathcal{M}^v$ denote the original 2D mask from which its matched component was obtained. Equality of M_a^v and M_b^v refers to source-mask identity in the current view. We use source-mask agreement as an association target, with a reliability weight reflecting local matching confidence and visibility:

$$c_{a,b}^v = \mathbf{1}[M_a^v = M_b^v], \qquad w_{a,b}^v = S_{(m_r,l)} S_{(m_{r'},l')} \min(\gamma_a^v, \gamma_b^v), \tag{11}$$

where $\gamma_a^v \in [0,1]$ denotes the visibility ratio of leaf a in view v, defined as the fraction of a leaf's anchors that are visible in the current view. Let $\mathbb{Q}_v$ contain these co-observed cross-root leaf pairs. Using binary cross-entropy BCE, we optimize

$$\mathcal{L}_{\mathrm{conn}} = \sum_{(a,b)\in\mathbb{Q}_v} \mathrm{sg}[w_{a,b}^v]\, \mathrm{BCE}(C_{a,b}, c_{a,b}^v). \tag{12}$$

With connectivity-loss weight λ_{conn}, the joint objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{leaf}} + \lambda_{\mathrm{conn}} \mathcal{L}_{\mathrm{conn}}. \tag{13}$$

After training, the learned connectivity links fuse local leaves into coherent scene-level instances.

Connectivity-guided leaf fusion. During inference, we progressively aggregate local leaves into scene-level instances using learned connectivity and the spatial adjacency matrix $\boldsymbol{P}$. Initializing each leaf as an isolated group, we define the set of physically adjacent cross-root leaf pairs between two groups $\mathbb{G}_i$ and $\mathbb{G}_j$ as:

$$\mathbb{Q}(\mathbb{G}_i, \mathbb{G}_j) = \{(a,b) \mid a \in \mathbb{G}_i,\ b \in \mathbb{G}_j,\ r(a) \neq r(b),\ P_{a,b} = 1\}, \tag{14}$$

where $r(a)$ is the root of leaf a, and entry $P_{a,b} \in \{0,1\}$ indicates physical adjacency in 3D. For nonempty $\mathbb{Q}(\mathbb{G}_i, \mathbb{G}_j)$, the scalar group-level connectivity is

$$\bar{C}(\mathbb{G}_i, \mathbb{G}_j) = \frac{1}{|\mathbb{Q}(\mathbb{G}_i, \mathbb{G}_j)|} \sum_{(a,b)\in\mathbb{Q}(\mathbb{G}_i,\mathbb{G}_j)} C_{a,b}. \tag{15}$$

We greedily merge the group pair with the highest $\bar{C}$ provided $\mathbb{Q}(\mathbb{G}_i, \mathbb{G}_j) \neq \emptyset$ and $\bar{C} > \tau_{\mathrm{conn}}$, repeating until convergence.

4 Experiments

4.1 Benchmark and Evaluation Protocol

AG-Seg benchmark. AG-Seg adds instance-level annotations and Click/Text evaluation protocols to three synthetic and two real scenes from Horizon-GS and two synthetic scenes from UC-GS. The seven scenes cover cross-altitude appearance changes and diverse semantic categories in aerial–ground environments. For open-vocabulary evaluation, we follow the annotation procedure of LERF (Kerr et al., 2023). The query text prompts used in each scene, annotation counts, and detailed evaluation protocols are provided in Appendix B (Annotation and Evaluation Details).

Query protocols and metrics. Following AG^{2}aussian, we evaluate Click and Text queries using mean intersection-over-union (mIoU). We use *best-of-three Click evaluation*: three deterministic positive points independently select method-native regions for each annotated instance. Both IoU and boundary IoU (BIoU) use the prediction with the highest ground-truth IoU. For *Text*, we average IoUs over positive text–image pairs within each scene, using the union of visible matching instances as ground truth. *Average* in Table 1 is the equal-weight mean over the seven AG-Seg scenes, reported separately for each track. Click sampling, query prompts, and annotation statistics are detailed in Appendix B.

Baselines. We compare against OpenGaussian Wu et al. (2024), InstanceGaussian (Li et al., 2025a), Unified-Lift (Zhu et al., 2025c), AG^{2}aussian (Wang et al., 2025), LangSplatV2 (Li et al., 2025b), COS3D (Zhu et al., 2025b), GALA (Alegret et al., 2026), VALA (Wang et al., 2026c), and SCOUP (Budimir et al., 2026). Methods that do not support discrete region selection are evaluated only under the Text protocol.

Implementation Details. We adopt Urban-GS (Wang et al., 2026a) as the 3D-GS backbone and use a training schedule of 80K iterations for all scenes. We use 64 roots by default and 128 roots for the larger Citysample, Road, and Park scenes, with 24 leaves allocated to each root. For the remaining hyperparameters, we set $\lambda_{\mathrm{conn}} = 0.1$. The learning rates for the anchor leaf logits and connectivity logits are 1×10^{-3} and 1×10^{-2}, respectively. Additional details are provided in Appendix A.

Table 1: **Quantitative results on all seven AG-Seg scenes.** The best result in each column is **bold**, while the second-best result is underlined. All results are reported in mIoU.

Prompt	Method	Average	*Colosseum*	*Elvenruin*	*Citysample*	*Road*	*Park*	*NYC*	*SF*
Click	OpenGaussian	0.263	0.357	0.421	0.332	0.114	0.201	0.218	0.198
	InstanceGaussian	0.377	0.586	0.577	0.393	0.223	0.211	0.331	0.318
	Unified-Lift	0.579	0.714	0.672	0.582	0.558	0.562	0.434	0.527
	AG^{2}aussian	0.504	0.759	0.624	0.367	0.245	0.508	0.505	0.522
	Ours	**0.773**	**0.800**	**0.857**	**0.868**	**0.685**	**0.683**	**0.725**	**0.795**
Text	OpenGaussian	0.184	0.273	0.279	0.172	0.129	0.183	0.103	0.154
	InstanceGaussian	0.250	0.366	0.328	0.241	0.144	0.180	0.159	0.331
	Unified-Lift	0.367	0.572	0.376	0.293	0.323	0.411	0.201	0.397
	AG^{2}aussian	0.366	0.609	0.358	0.394	0.218	0.345	0.228	0.407
	LangSplatV2	0.324	0.326	0.323	0.379	0.336	0.350	0.175	0.382
	COS3D	0.244	0.437	0.423	0.115	0.155	0.191	0.224	0.160
	GALA	0.275	0.417	0.276	0.166	0.141	0.353	0.161	0.409
	VALA	0.418	0.558	0.484	0.334	0.416	0.433	0.256	0.448
	SCOUP	0.371	0.410	0.444	0.297	0.357	0.433	0.228	0.426
	Ours	**0.527**	**0.687**	**0.505**	**0.608**	**0.513**	**0.505**	**0.309**	**0.560**

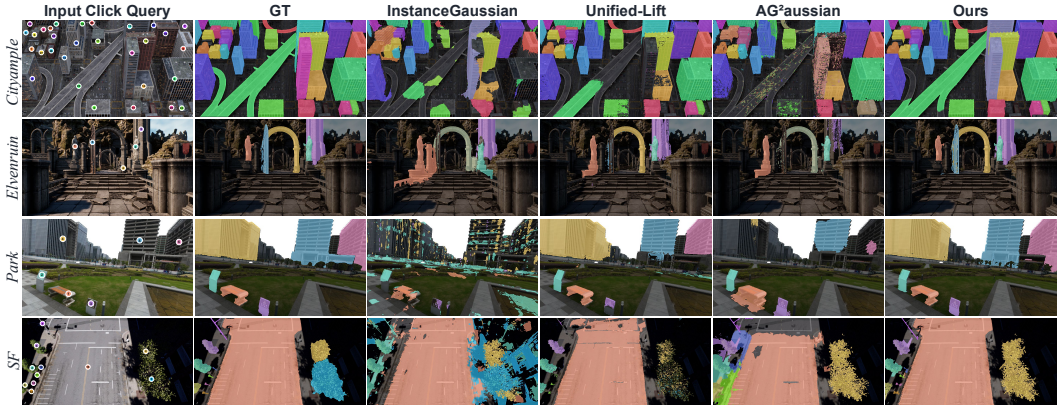

Figure 4: **Qualitative Click comparison on AG-Seg.** Our method yields more complete instance selections with clearer boundaries.

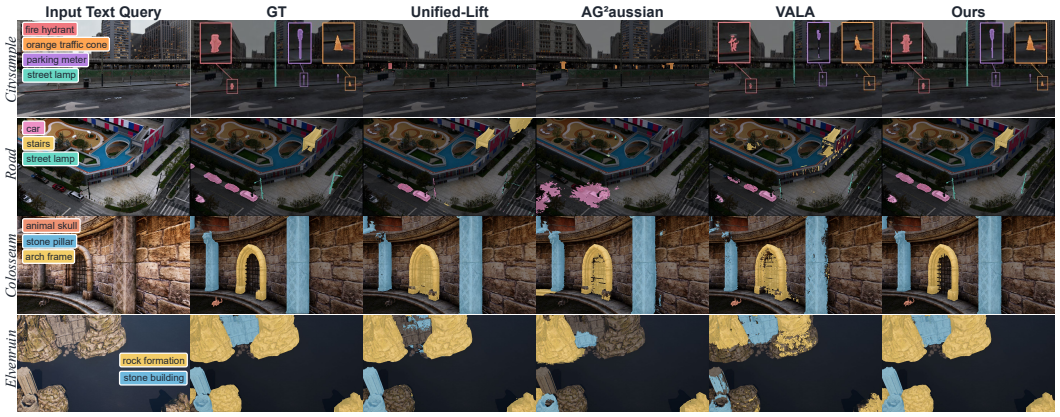

Figure 5: **Qualitative Text comparison on AG-Seg.** Our method localizes queried objects more accurately with less background leakage.

4.2 Experiment Results and Analysis

Table 1 shows that SSCL ranks first in all seven AG-Seg scenes under both query protocols, achieving average Click/Text mIoUs of 0.773/0.527. These exceed Unified-Lift and VALA, the strongest baselines in the respective tracks, by 19.4 and 10.9 percentage points. The largest gains occur in Citysample, where Click mIoU increases from 0.582 to 0.868 and Text mIoU from 0.394 to 0.608 relative to the best competing methods. The improvements also hold on the real-world Road and

Park scenes, where SSCL achieves Click mIoUs above 0.68 and Text mIoUs above 0.50. This consistent advantage across synthetic and captured scenes supports the effectiveness of the learned instance representation for both spatial selection and semantic retrieval. The qualitative comparisons further illustrate these improvements: Click queries recover more complete instance regions with clearer boundaries (Fig. 4), while Text queries localize matching objects with less background leakage (Fig. 5). Additional quantitative analyses, including semantic-field rendering efficiency and storage comparisons, are provided in Appendix C.3. Demonstrations of scene editing and physical simulation are presented in Appendix C.1, with further query visualizations in Appendix D.

4.3 Ablation Studies

We conduct ablation studies on the five-scene Horizon-GS subset of AG-Seg and report equal-weight averages. Detailed per-scene results are provided in Appendix C.2.

Table 2: **Ablation of codebook structure and granularity-aware learning.** Results are averaged over the five-scene Horizon-GS subset of AG-Seg. The "+" symbol denotes cumulative additions to our baseline.

	Click		Text	
Method	mIoU ↑	mBIoU ↑	mIoU ↑	mBIoU ↑
Global codebook baseline	0.618	0.572	0.395	0.336
Our baseline framework	0.731	0.696	0.522	0.456
+ Granularity-aware assignment	0.754	0.723	0.548	0.487
+ Fragment attachment	0.767	0.739	0.553	0.488
+ Purity-weighted supervision	**0.778**	**0.753**	**0.564**	**0.513**

Granularity-aware codebook learning. Table 2 compares the global-codebook baseline (Zhu et al., 2025c) with our root-local framework (Secs. 3.1 and 3.3). Starting from Unified-Lift matching without fragment attachment or purity weighting, we cumulatively add the three components in Sec. 3.2. Granularity-aware assignment uses purity–coverage scores for Hungarian matching (Eq. 8). Fragment attachment lets multiple over-segmented components supervise a shared leaf (Eq. 9). Purity-weighted supervision weights each assigned component's loss by s_{purity} (Eq. 10), reducing the influence of low-response assignments. Together, these components improve Click/Text mIoU from 0.731/0.522 to 0.778/0.564.

Table 3: **Ablation of confidence and visibility factors in cross-root edge weighting.**

$S_{(m_r,l)}S_{(m_{r'},l')}$	$\min(\gamma_a^v, \gamma_b^v)$	Click	
		mIoU ↑	mBIoU ↑
×	×	0.768	0.742
√	×	0.772	0.745
×	√	0.774	0.747
√	√	**0.778**	**0.753**

Table 4: **Sensitivity analysis on connectivity-fusion threshold τ_{conn}.**

τ_{conn}	Click	
	mIoU ↑	mBIoU ↑
0.45	0.763	0.737
0.50	**0.778**	**0.753**
0.55	0.776	0.750
0.60	0.776	0.749

Cross-root edge weighting. We ablate the two factors in equation 11, where $S_{(m_r,l)}$ and $S_{(m_{r'},l')}$ are the retained assignment scores defined in Sec. 3.2. Assignment confidence $S_{(m_r,l)}S_{(m_{r'},l')}$ downweights uncertain matches. The visibility factor $\min(\gamma_a^v, \gamma_b^v)$ downweights partial observations. Table 3 validates both components in enhancing cross-root connectivity learning.

Fusion threshold sensitivity. The default connectivity threshold τ_{conn} is set to 0.5, providing an intuitive decision boundary for group fusion. Additional evaluations with $\tau_{\text{conn}} \in \{0.45, 0.55, 0.60\}$ are conducted to assess parameter sensitivity. As shown in Table 4, performance remains stable across these settings, indicating the fusion strategy is not sensitive to the precise choice of τ_{conn}.

5 Conclusion

In this work, we present the Spatially Structured Codebook Learning (SSCL) framework for language-grounded 3D instance understanding in large aerial–ground environments. To handle sparse view coverage and cross-view granularity mismatches, SSCL factorizes the global identity space into local codebooks whose leaves are associated through learned cross-root connectivity, while incorporating a granularity-aware association mechanism to harmonize inconsistent masks. Extensive experiments on AG-Seg show that SSCL consistently outperforms existing baselines across all interaction modes, demonstrating its effectiveness on the evaluated aerial–ground scenes.

REFERENCES

Elena Alegret, Kunyi Li, Sen Wang, Siyun Liang, Michael Niemeyer, Stefano Gasperini, Nassir Navab, and Federico Tombari. GALA: Guided attention with language alignment for open vocabulary Gaussian splatting. In *Proceedings of the International Conference on 3D Vision (3DV)*, pp. 1717–1727, 2026.

Lovre Antonio Budimir, Yushi Guan, Steve Ryhner, Sven Lončarić, and Nandita Vijaykumar. Sparse code uplifting for efficient 3D language Gaussian splatting. *arXiv preprint arXiv:2605.13600*, 2026.

Jiazhong Cen, Jiemin Fang, Chen Yang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. Segment any 3D Gaussians. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 39, pp. 1971–1979, 2025.

Shijie Chen and Peixi Peng. FreeGen: Feed-forward reconstruction–generation co-training for free-viewpoint driving scene synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 682–699, 2026.

Jiahuan Cheng, Jan-Nico Zaech, Luc Van Gool, and Danda Pani Paudel. Occam's LGS: An efficient approach for language Gaussian splatting. In *Proceedings of the 36th British Machine Vision Conference (BMVC)*, pp. 1–14, 2025.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2432–2443, 2017.

Jiayu Ding, Xinpeng Liu, Zhiyi Pan, Shiqiang Long, and Ge Li. ExtrinSplat: Decoupling geometry and semantics for open-vocabulary understanding in 3D Gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31019–31028, 2026.

Zihan Gao, Lingling Li, Licheng Jiao, Fang Liu, Wenping Ma, Yuwei Guo, and Shuyuan Yang. StratoSplat: Taming layered regularities for sparse aerial 3D Gaussian splatting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2026.

Yuanming Hu, Yu Fang, Ziheng Ge, Ziyin Qu, Yixin Zhu, Andre Pradhana, and Chenfanfu Jiang. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Transactions on Graphics*, 37(4):146:1–146:14, 2018.

Lihan Jiang, Kerui Ren, Mulin Yu, Linning Xu, Junting Dong, Tao Lu, Feng Zhao, Dahua Lin, and Bo Dai. Horizon-GS: Unified 3D Gaussian splatting for large-scale aerial-to-ground scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26789–26799, 2025.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023.

Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. LERF: Language embedded radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19729–19739, 2023.

Jun-Seong Kim, GeonU Kim, Yu-Ji Kim, Yu-Chiang Frank Wang, Jaesung Choe, and Tae-Hyun Oh. Dr. Splat: Directly referring 3D Gaussian splatting via direct language embedding registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14137–14146, 2025.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4015–4026, 2023.

Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97, 1955.

Haijie Li, Yanmin Wu, Jiarui Meng, Qiankun Gao, Zhiyao Zhang, Ronggang Wang, and Jian Zhang. InstanceGaussian: Appearance-semantic joint Gaussian representation for 3D instance-level perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14078–14088, 2025a.

Kunyi Li, Michael Niemeyer, Sen Wang, Stefano Gasperini, Nassir Navab, and Federico Tombari. OpenGaFF: Open-vocabulary Gaussian feature field with codebook attention. *arXiv preprint arXiv:2605.06088*, 2026a.

Tingyun Li, Xinyi Liu, Yongjun Zhang, Yi Wan, Xiaoan Liu, Weiwei Fan, and Jiahao Liu. AeroGS: Scale-aware Gaussian splatting for pose-free dynamic UAV scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 40907–40917, 2026b.

Wanhua Li, Yujie Zhao, Minghan Qin, Yang Liu, Yuanhao Cai, Chuang Gan, and Hanspeter Pfister. LangSplatV2: High-dimensional 3D language Gaussian splatting with 450+ FPS. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 174306–174330, 2025b.

Zhida Li, Jianqiao Zhu, Hejin Huang, Yipeng Qin, Sibei Yang, and Guanbin Li. Direct language embedding enables Gaussian splatting for large scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 7231–7240, 2026c.

Siyun Liang, Sen Wang, Kunyi Li, Michael Niemeyer, Stefano Gasperini, Hendrik P. A. Lensch, Nassir Navab, and Federico Tombari. Supergseg: Open-vocabulary 3d segmentation with structured super-gaussians. In *Proceedings of the 2026 International Conference on 3D Vision (3DV)*, pp. 1650–1659, 2026.

Kunhao Liu, Fangneng Zhan, Jiahui Zhang, Muyu Xu, Yingchen Yu, Abdulmotaleb El Saddik, Christian Theobalt, Eric Xing, and Shijian Lu. Weakly supervised 3D open-vocabulary segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 53433–53456, 2023.

Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-GS: Structured 3D Gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20654–20664, 2024.

Yuning Peng, Haiping Wang, Yuan Liu, Chenglu Wen, Zhen Dong, and Bisheng Yang. GAGS: Granularity-aware feature distillation for language Gaussian splatting. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 40, pp. 8376–8384, 2026.

Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. LangSplat: 3D language Gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20051–20060, 2024.

Yansong Qu, Shaohui Dai, Xinyang Li, Jianghang Lin, Liujuan Cao, Shengchuan Zhang, and Rongrong Ji. GOI: Find 3D Gaussians of interest with an optimizable open-vocabulary semantic-space hyperplane. In *Proceedings of the 32nd ACM International Conference on Multimedia (ACM MM)*, pp. 5328–5337, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139, pp. 8748–8763, 2021.

Kerui Ren, Lihan Jiang, Tao Lu, Mulin Yu, Linning Xu, Zhangkai Ni, and Bo Dai. Octree-GS: Towards consistent real-time rendering with LOD-structured 3D Gaussians. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–15, 2025.

Kota Shimomura, Hidehisa Arai, Tsubasa Takahashi, Takayoshi Yamashita, and Hironobu Fujiyoshi. P2GS: Physical prior-guided Gaussian splatting for photometrically consistent urban reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11736–11745, 2026.

Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Norman Müller, Matthias Nießner, Angela Dai, and Peter Kontschieder. Panoptic lifting for 3D scene understanding with neural fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9043–9052, 2023.

László Szilágyi, Francis Engelmann, and Jeannette Bohg. SLAG: Scalable language-augmented Gaussian splatting. *IEEE Robotics and Automation Letters*, 10(7):6991–6998, 2025.

Meng Wang, Changqun Xia, Yuze Wang, Junyi Wang, Wantong Duan, Xinxiong Xie, and Yue Qi. Urban-GS: A unified 3D Gaussian splatting framework for compact and high-fidelity aerial-to-street reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 33207–33216, 2026a.

Renzhi Wang, Yuxiang Fu, Wuqi Wang, Haigen Min, Wei Feng, Lei Ma, and Qing Guo. Beyond visual reconstruction quality: Object perception-aware 3D Gaussian splatting for autonomous driving. In *International Conference on Learning Representations (ICLR)*, 2026b.

Sen Wang, Kunyi Li, Siyun Liang, Elena Alegret, Jing Ma, Nassir Navab, and Stefano Gasperini. Visibility-aware language aggregation for open-vocabulary segmentation in 3D Gaussian splatting. In *Proceedings of the International Conference on 3D Vision (3DV)*, pp. 1791–1800, 2026c.

Zhaonan Wang, Manyi Li, and Changhe Tu. AG^{2}aussian: Anchor-Graph structured Gaussian splatting for instance-level 3D scene understanding and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 26806–26816, 2025.

Yanmin Wu, Jiarui Meng, Haijie Li, Chenming Wu, Yahao Shi, Xinhua Cheng, Chen Zhao, Haocheng Feng, Errui Ding, Jingdong Wang, and Jian Zhang. OpenGaussian: Towards point-level 3D Gaussian-based open vocabulary understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 19114–19138, 2024.

Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. PhysGaussian: Physics-integrated 3D Gaussians for generative dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Weiyi Xue, Fan Lu, Chi Zhang, Tianhang Wang, Sanqing Qu, Zehan Zheng, Boyuan Zheng, Junqiao Zhao, and Guang Chen. Signal structure-aware Gaussian splatting for large-scale scene reconstruction. In *International Conference on Learning Representations (ICLR)*, 2026.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: A high-fidelity dataset of 3D indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12–22, 2023.

Saining Zhang, Baijun Ye, Xiaoxue Chen, Yuantao Chen, Zongzheng Zhang, Cheng Peng, Yongliang Shi, and Hao Zhao. Drone-assisted road Gaussian splatting with cross-view uncertainty. In *Proceedings of the 35th British Machine Vision Conference (BMVC)*, pp. 1–15, 2024.

Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3DGS: Supercharging 3D Gaussian splatting to enable distilled feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21676–21685, 2024.

Ruijie Zhu, Mulin Yu, Linning Xu, Lihan Jiang, Yixuan Li, Tianzhu Zhang, Jiangmiao Pang, and Bo Dai. ObjectGS: Object-aware scene reconstruction and scene understanding via Gaussian splatting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8350–8360, 2025a.

Runsong Zhu, Ka-Hei Hui, Zhengzhe Liu, Qianyi Wu, Weiliang Tang, Shi Qiu, Pheng-Ann Heng, and Chi-Wing Fu. COS3D: Collaborative open-vocabulary 3D segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 11351–11371, 2025b.

Runsong Zhu, Shi Qiu, Zhengzhe Liu, Ka-Hei Hui, Qianyi Wu, Pheng-Ann Heng, and Chi-Wing Fu. Rethinking end-to-end 2D to 3D scene segmentation in Gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3656–3665, 2025c.

Supplementary Material

A Detailed Implementation and Hyperparameters

We first reconstruct every scene with the anchor-based Urban-GS pipeline (Wang et al., 2026a). For the five Horizon-GS scenes in AG-Seg, we downsample the input images by a factor of two. Following previous works (Li et al., 2025b; Zhu et al., 2025b; Wang et al., 2026c), we then freeze all geometry and appearance parameters and optimize only the semantic variables. We use the same optimization schedule for all scenes. Semantic training runs for 80k iterations: 0–20k uses only $\mathcal{L}_{\text{leaf}}$ to optimize leaf logits. Granularity-aware learning starts at 20k and $\mathcal{L}_{\text{conn}}$ at 30k. Leaf optimization continues throughout, with both losses active from 30k to 80k. The maximum local capacity is $L = 24$ leaves per root. We use $R = 64$ roots as default setting, while larger scenes, including Citysample, Road, and Park, are partitioned into $R = 128$ roots. The anchor-local leaf logits are optimized with Adam at a learning rate of 10^{-3}, while the connectivity logits are optimized with SparseAdam at 10^{-2}. In equation 13, the leaf and connectivity loss weights are 1.0 and 0.1, respectively. For all scenes, the cross-root connectivity threshold is $\tau_{\text{conn}} = 0.5$. We construct the spatial adjacency matrix $\boldsymbol{P}$ from a KNN graph on frozen 3D anchors by counting distinct cross-leaf anchor pairs: entry $P_{a,b} = 1$ iff the count exceeds 20, and zero otherwise. Fragment attachment uses hard-ownership statistics with $\tau_{\text{high}} = 0.75$, $\tau_{\text{low}} = 0.5$. Given the large scene scale and instance count, we increase the codebook size of instance-based baselines, including Unified-Lift and GALA, to 1024. For other methods whose codebook entries represent CLIP feature prototypes, we increase the codebook size to 128 for evaluation. Following prior work (Zhu et al., 2025c; Wang et al., 2025; Wang et al., 2026c), we use only single-level SAM masks for supervision. Specifically, we adopt the 2D mask–3D instance association strategy from AG^{2}aussian (Wang et al., 2025) to associate each 3D instance with a 512-dimensional CLIP image feature. Since Unified-Lift (Zhu et al., 2025c) does not provide an official protocol for CLIP feature association, its evaluation follows the same setting as AG^{2}aussian for a consistent comparison.

B AG-Seg Annotation and Evaluation Details

AG-Seg extends seven scenes from Horizon-GS and UC-GS with instance-level annotations for Click and Text evaluation. The annotations follow the query-based labeling protocols of LERF (Kerr et al., 2023) and LangSplat (Qin et al., 2024). Candidate regions whose valid pixel region contain fewer than 50 pixels are excluded from annotation. For Text evaluation, a single query may match multiple instances in the same image. Because some evaluated methods do not maintain explicit instance identities, we evaluate their complete pixel-level responses: the set of all pixels returned for a query is compared with the union of the annotated ground-truth pixels for all matching instances. Click evaluation, in contrast, retains an instance-level protocol and evaluates each annotated instance separately. Figure 6 presents representative Text and Click annotations for all seven AG-Seg scenes, covering both aerial and ground views with diverse object categories and observation scales. The semantic concepts and relations in these scenes are too numerous and complex for exhaustive annotation. AG-Seg therefore covers a representative subset of instances rather than every possible semantic concept and its associated instances. Nevertheless, the per-scene statistics in Table 5 demonstrate that the current annotations are sufficiently broad and substantial to support a diverse and challenging evaluation.

Best-of-three Click evaluation. All methods use the same three points per ground-truth mask at evaluation resolution. We compute the boundary-distance function $D(\boldsymbol{u})$ using OpenCV's L2 distance transform (mask size 5) with a one-pixel zero border. Over foreground pixels, the first point is $\boldsymbol{u}_1 = \arg\max_{\boldsymbol{u}} D(\boldsymbol{u})$. The candidate set $\mathbb{S}$ contains foreground pixels satisfying $D(\boldsymbol{u}) \geq 0.25 D(\boldsymbol{u}_1)$. We retain candidates within $\|\boldsymbol{u} - \boldsymbol{u}_1\|_2 \leq 0.30d$, where $d = \max(1, \sqrt{(x_{\max} - x_{\min})^2 + (y_{\max} - y_{\min})^2})$ uses mask-pixel extrema, provided at least $\min(3, |\mathbb{S}|)$ candidates remain. Otherwise, we use all of $\mathbb{S}$. The next two points greedily maximize distance to the nearest selected point. Sampling ties use row-major order. Exhausted candidates are repeated cyclically in selection order. Each click independently returns one prediction. We select the first prediction attaining maximum ground-truth IoU and compute both IoU and BIoU from that mask, without maximizing BIoU separately.

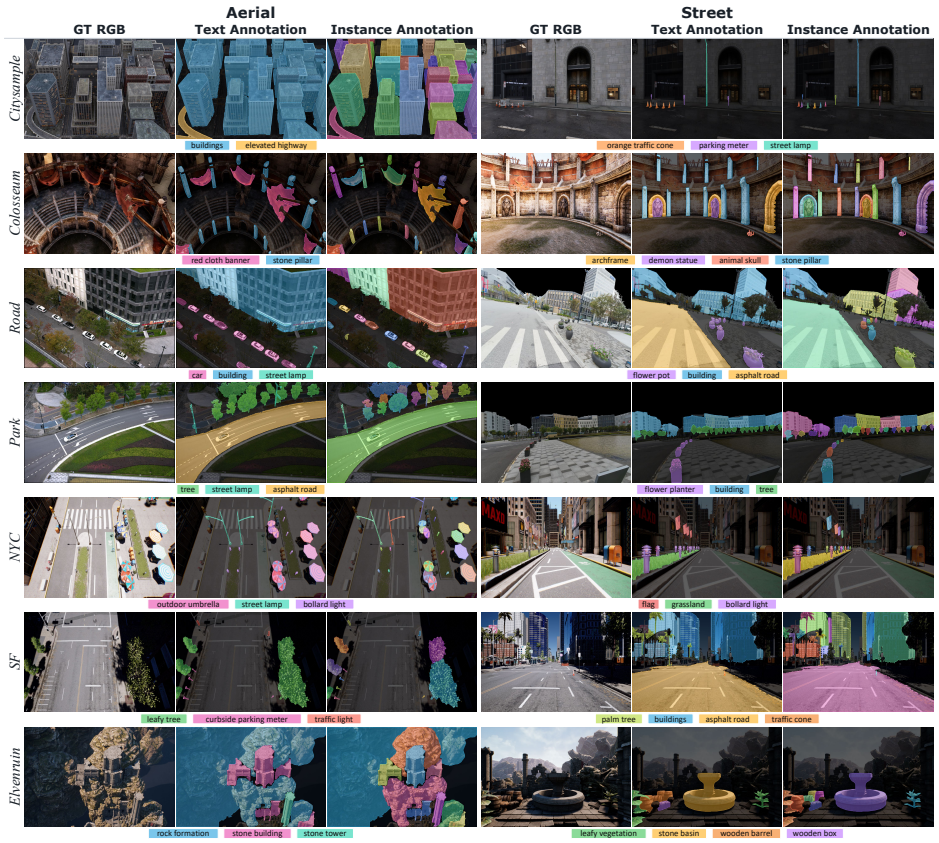

Figure 6: **Examples of AG-Seg annotations.**

Table 5: **AG-Seg annotation and evaluation inventory.** Type indicates whether the scene is constructed from real-world captured data (**R**) or generated in a synthetic virtual environment (**S**).

Scene	Type	Query text	Text categories	Text queries	Click queries
Colosseum	S	`animal skull`, `archframe`, `balustrade`, `carved panel`, `demon statue`, `human statue`, `red cloth banner`, `soil ground`, `stone pillar`	9	113	369
Elvenruin	S	`bench`, `leafy vegetation`, `ornate metal gate`, `robed stone statue`, `rock formation`, `stone building`, `arch`, `stone basin`, `stone tower`, `winged stone statue`, `wooden barrel`, `wooden box`, `wooden crate`	13	145	446
Citysample	S	`black trash can`, `maroon buildings`, `blue mailbox`, `buildings`, `car`, `elevated highway`, `fire hydrant`, `green construction fence`, `metal traffic barrier`, `orange traffic cone`, `parking meter`, `road surface`, `stop sign`, `street clock`, `street lamp`, `traffic light`, `tree`	17	336	1,247
Road	R	`air conditioning unit`, `bicycle`, `blue traffic sign`, `building`, `car`, `crane truck`, `electric scooter`, `flower pot`, `grassland`, `hedge`, `metal fence`, `metallic lattice ring`, `parking sign`, `asphalt road`, `stairs`, `street lamp`, `traffic light`, `tree`, `wooden louvered enclosure`	19	385	1,274
Park	R	`Construction vehicle`, `bench`, `blue road sign`, `building`, `car`, `flower bed`, `flower planter`, `hedge`, `metal lattice ring sculpture`, `red ornamental plants`, `asphalt road`, `plaque`, `street lamp`, `traffic light`, `tree`, `yellow ornamental plants`	16	230	988
NYC	S	`A-frame barricade`, `building`, `flag`, `grassland`, `metal crowd control barrier`, `outdoor umbrella`, `asphalt road`, `bollard light`, `street lamp`, `traffic light`, `trash bag`	11	155	564
SF	S	`USPS mailbox`, `buildings`, `temporary construction fence`, `fire hydrant`, `palm tree`, `curbside parking meter`, `asphalt road`, `streetlight`, `traffic cone`, `traffic light`, `waste bin`, `leafy tree`	12	99	343
Total			**97**	**1,463**	**5,231**

C Additional Results and Analysis

C.1 Application

Scene editing. Our scene-level instance identities allow selected objects to be manipulated as groups of Gaussians. Figure 7 demonstrates translation and removal of traffic cones in Citysample, as well as duplication and scaling of benches in Park. The instance representation isolates the target objects from nearby structures, enabling localized edits within the reconstructed scene.



Figure 7: **Instance-level scene editing.** Left: translating and removing traffic cones. Right: duplicating a bench and scaling another to $0.5\times$ its original size.

Physical simulation. We use PhysGaussian (Xie et al., 2024), based on the moving least squares material point method (MLS-MPM) (Hu et al., 2018), to animate selected instances. Figure 8 shows a barrier interacting with traffic cones and a ring sculpture undergoing different motions and deformations. Our representation provides the object selection, while the simulator governs the subsequent dynamics. These examples suggest potential uses in interactive VR/AR environments and scene variation for embodied-agent training.

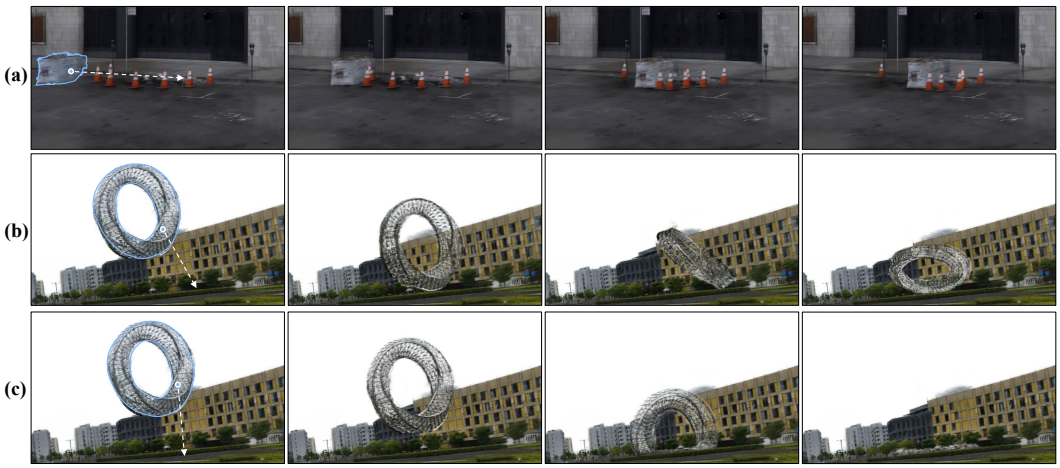

Figure 8: **Physical simulation of selected 3D instances.** Frames progress from left to right, showing (a) barrier–cone interaction in Citysample and (b, c) motion and deformation of a ring sculpture in Park under different simulation settings.

C.2 PER-SCENE ABLATIONS ON THE HORIZON-GS SUBSET OF AG-SEG

The main paper reports equal-weight averages over the five-scene Horizon-GS subset of AG-Seg. Table 6 expands the cumulative configurations in Table 2 with per-scene Click/Text mIoU and mBIoU (Sec. 3.2). Tables 7 and 8 provide per-scene Click results corresponding to Tables 3 and 4, respectively (Sec. 3.3): confidence and visibility factors in cross-root edge weighting, and sensitivity to connectivity-fusion threshold τ_{conn}.

Table 6: **Per-scene ablation of granularity-aware learning.** Click and Text denote best-of-three point-prompt selection and open-vocabulary text retrieval, respectively.

Configuration	Query	Metric	Colosseum	Elvenruin	Citysample	Road	Park
Our baseline framework	Click	mIoU	0.777	0.819	0.793	0.604	0.662
		mBIoU	0.745	0.785	0.758	0.560	0.632
	Text	mIoU	0.660	0.446	0.602	0.433	0.469
		mBIoU	0.588	0.399	0.527	0.362	0.404
+ Granularity-aware assignment	Click	mIoU	0.800	0.813	0.817	0.658	0.682
		mBIoU	0.772	0.781	0.790	0.616	0.655
	Text	mIoU	0.659	0.464	0.593	0.514	0.508
		mBIoU	0.596	0.417	0.541	0.434	0.445
+ Fragment attachment	Click	mIoU	0.801	0.829	0.843	0.676	0.687
		mBIoU	0.778	0.802	0.819	0.635	0.662
	Text	mIoU	0.694	0.470	0.566	0.518	0.517
		mBIoU	0.641	0.423	0.470	0.442	0.462
+ Purity-weighted supervision	Click	mIoU	0.800	0.857	0.868	0.685	0.683
		mBIoU	0.777	0.833	0.847	0.647	0.659
	Text	mIoU	0.687	0.505	0.608	0.513	0.505
		mBIoU	0.638	0.458	0.573	0.443	0.452

Table 7: **Per-scene ablation of confidence and visibility factors in cross-root edge weighting.**

$S_{(m_r,l)}S_{(m_{r'},l')}$	$\min(\gamma_a^v, \gamma_b^v)$	Metric	Colosseum	Elvenruin	Citysample	Road	Park
×	×	mIoU	0.793	0.837	0.852	0.677	0.684
		mBIoU	0.770	0.811	0.831	0.639	0.659
√	×	mIoU	0.799	0.839	0.854	0.682	0.686
		mBIoU	0.776	0.813	0.832	0.644	0.662
×	√	mIoU	0.795	0.852	0.855	0.680	0.688
		mBIoU	0.771	0.827	0.834	0.642	0.663
√	√	mIoU	0.800	0.857	0.868	0.685	0.683
		mBIoU	0.777	0.833	0.847	0.647	0.659

Table 8: **Per-scene sensitivity analysis on connectivity-fusion threshold τ_{conn}.**

τ_{conn}	Metric	Colosseum	Elvenruin	Citysample	Road	Park	Average
0.45	mIoU	0.799	0.819	0.843	0.680	0.676	0.763
	mBIoU	0.776	0.793	0.822	0.643	0.652	0.737
0.50	mIoU	**0.800**	**0.857**	**0.868**	**0.685**	0.683	**0.778**
	mBIoU	**0.777**	**0.833**	**0.847**	**0.647**	0.659	**0.753**
0.55	mIoU	0.799	0.848	0.863	0.682	**0.689**	0.776
	mBIoU	0.775	0.823	0.842	0.644	**0.664**	0.750
0.60	mIoU	0.799	0.854	0.859	0.679	0.688	0.776
	mBIoU	0.776	0.829	0.836	0.641	0.663	0.749

C.3 EFFICIENCY CONSIDERATIONS

Tables 9 and 10 report semantic-field rendering throughput and semantic tensor storage subtotals across all seven AG-Seg scenes, using the same $k = 4$ inference setting as Table 1. This setting is used throughout validation and testing.

Rendering protocol. All methods use the same registered training and test views per scene. Rendering resolutions are 960×540 for Colosseum, Elvenruin, and Citysample and 960×480 for NYC and SF. For Road and Park, we use 800×533 for aerial views and 800×450 for street-level views. On a single NVIDIA RTX 4090, we measure CUDA-event time over three complete passes with independently shuffled view orders. Per-scene FPS is total rendered frames divided by total execution time. The reported mean equally weights the seven scenes. The timed region includes visibility filtering, neural Gaussian generation, projection, depth sorting, rasterization, and each method's native feature normalization. Model loading, warm-up, disk I/O, semantic-code packing, language decoding, and text matching are excluded. Our method and LangSplatV2 both use indexed top-4 weights for sparse rendering. Our method additionally retains root identities for root-aware compositing. The root-table capacity is fixed before timing and is set sufficiently large to avoid overflow for all evaluated views.

Table 9: **Semantic-field rendering FPS on all seven AG-Seg scenes.**

Scene	Unified-Lift	AG^{2}aussian	LangSplatV2	VALA	Ours
Colosseum	134	148	118	32	104
Elvenruin	111	120	95	28	114
Citysample	55	59	50	17	56
Road	100	112	88	29	105
Park	104	116	89	27	108
NYC	92	113	75	35	111
SF	78	96	67	27	100
Mean	96	109	83	28	100

Semantic storage. Both LangSplatV2 and SSCL use top-4 sparse codebook storage. LangSplatV2 uses a shared language codebook, while SSCL organizes instance-level entries within spatial roots. Table 10 reports the semantic tensor storage subtotals. The efficiency comparison is limited to semantic-field rendering and these storage subtotals on the seven AG-Seg scenes.

Table 10: **Semantic tensor storage subtotals on AG-Seg** (MiB).

Scene	Unified-Lift	AG^{2}aussian	LangSplatV2	VALA	Ours
Colosseum	105	57	32	3,308	38
Elvenruin	108	239	33	3,392	41
Citysample	210	673	65	6,611	79
Road	162	711	50	5,061	63
Park	133	261	41	4,192	52
NYC	45	47	14	1,381	19
SF	58	73	18	1,780	25
Mean	117	294	36	3,675	45

C.4 DEPTH-ADAPTIVE SAM PROMPTING UNDER AERIAL–GROUND SCALE VARIATION

We compare ordinary SAM with a GAGS-style distance-adaptive prompting implementation (Peng et al., 2026) on Citysample and Road. Both mask sets are used in the same baseline framework, with granularity-aware assignment, fragment attachment, and purity-weighted supervision disabled. All other downstream components remain unchanged. Ours denotes the full method.

Under this implementation and setting, Table 11 shows lower Click mIoU on Citysample and similar performance on Road. Figure 9 reveals missing regions in the retained GAGS-style masks. These coverage gaps can weaken 3D supervision alongside granularity mismatches, so the performance differences cannot be attributed solely to the relation between distance and granularity. This two-scene comparison illustrates how the evaluated prompting implementation can omit valid regions under aerial–ground viewing conditions.

Table 11: **Comparison with GAGS-style SAM prompting.** Best-of-three Click mIoU (↑) on Citysample and Road.

Scene	Baseline	GAGS-style	Ours
Citysample	0.793	0.696	**0.868**
Road	0.604	0.601	**0.685**

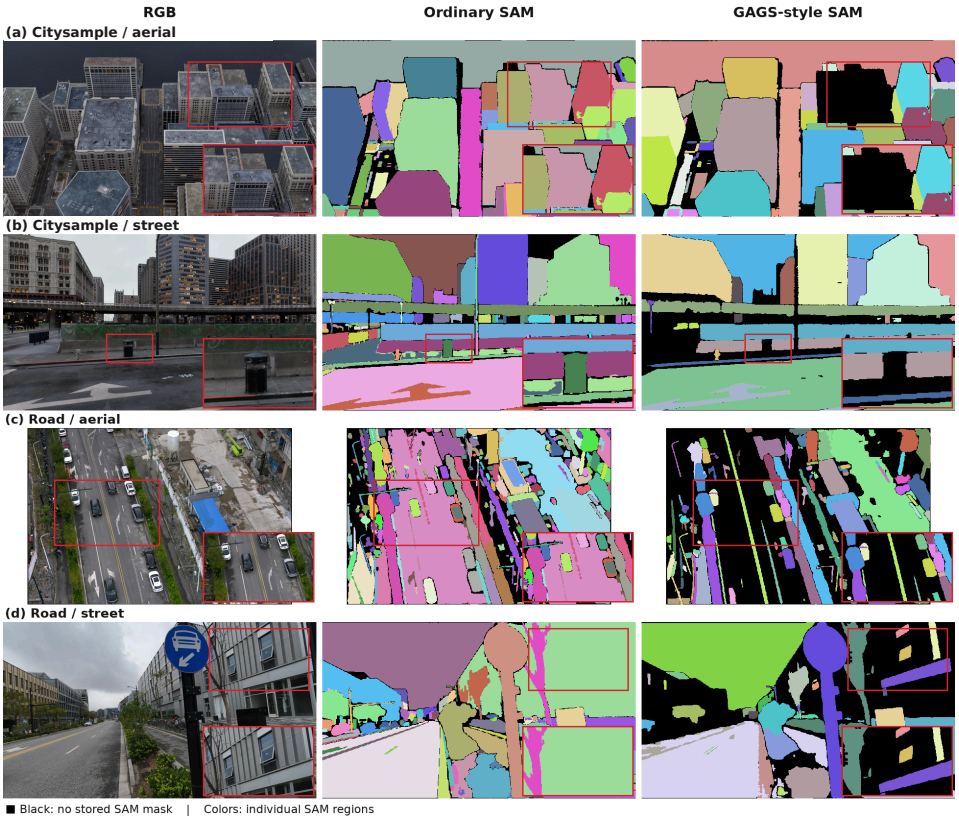

Figure 9: **Missing regions in the SAM inputs to reconstruction.** We visualize all stored regions in the ordinary SAM input layer and the GAGS-style SAM input layer. Colors distinguish regions within each panel. Black denotes pixels with no retained SAM mask. Red boxes and insets show corresponding image regions. GAGS-style inputs omit building regions beside retained neighboring buildings in (a), a street-side trash can in (b), much of the asphalt road in (c), and a building facade in (d), even when narrow markings or facade parts are retained. These examples show coverage gaps in the evaluated GAGS-style implementation before 3D learning.

D ADDITIONAL VISUALIZATIONS

Figure 10 provides additional Text-query comparisons across all seven AG-Seg scenes, covering both aerial and ground views. The examples range from large regions, such as buildings and roads, to small objects and thin structures, such as flower pots, street lamps, and barriers. Our method generally recovers more complete target regions with less background leakage and fragmentation, complementing the quantitative results in the main paper. Additional visualizations are provided in the supplementary video *demo.mp4*.



Figure 10: **Additional open-vocabulary selection results on AG-Seg.** Colored overlays indicate the regions selected by the corresponding text queries. Insets highlight local details.

E PERFORMANCE LIMITATIONS

Our performance remains bounded by the frozen 2D foundation models used to construct supervision and interpret text queries. Missing, fragmented, or over-merged SAM masks can propagate into the learned 3D identities, especially for small, thin, heavily occluded, or weakly observed objects. A merged SAM mask can yield pure root-local components for distinct objects, creating an incorrect positive connection even with high confidence and visibility. Patch co-occurrence and 3D adjacency do not certify identity, so errors may persist when views separating the objects provide insufficient negative evidence. CLIP-based matching can also be unreliable when an isolated object lacks sufficient visual context or appears very differently in aerial and ground views. These issues are particularly visible for ambiguous queries and objects observed at extreme scales. Figure 11 provides crop-level diagnostics of two limitations inherited from the region-level CLIP matcher. In Fig. 11(a), preserving the same umbrella foreground and crop extent while masking its surroundings lowers the score for "outdoor umbrella," showing that an otherwise valid object crop can lose essential scene context. In Fig. 11(b), the same bench is sharply resolved in View 1 but occupies only a few blurred pixels in the more distant View 2, and its "bench" relevance decreases accordingly. Because the bench pair also changes viewpoint, it illustrates combined scale-and-viewpoint variation rather than an isolated scale effect.

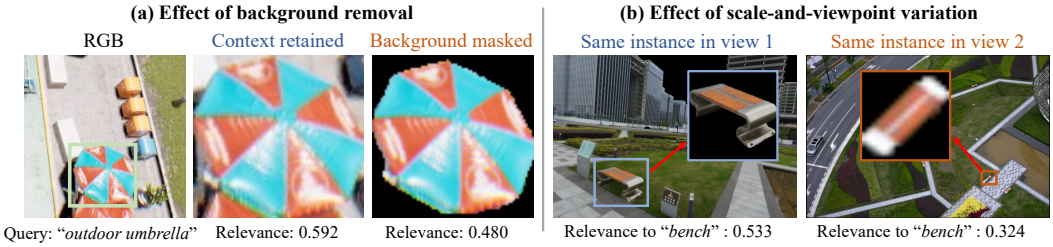

Figure 11: **Crop-level CLIP diagnostics under (a) background removal and (b) scale-and-viewpoint variation.**

F AI USE STATEMENT

For tasks requiring disclosure, we used generative AI tools to help implement our method and translation for paper writing. We did not use AI for synthetic dataset creation, theoretical modeling or conceptual framework development, mathematical claim formulation, hypothesis generation or refinement, research methodology or experimental design and feedback, dataset cleaning or reformatting, qualitative or thematic analysis of data, or interpretation of results. Supplying key arguments for mathematical proofs and drafting proofs are not applicable, as this work contains no formal proofs.

For tasks with recommended disclosure, we used AI to create and edit implementation code, draft manuscript text, improve readability, and adjust figure text and layout. The authors reviewed and revised the AI-assisted text, checked the translations, and inspected and tested the AI-assisted code for correctness. The authors take full responsibility for the final manuscript and accompanying artifacts, including any AI-assisted content.