ACTIVELANG: ACTIVE OPEN-VOCABULARY 3D MAPPING WITH SEMANTIC-UNCERTAINTY-GUIDED EXPLORATION

Anonymous authors
Paper under double-blind review

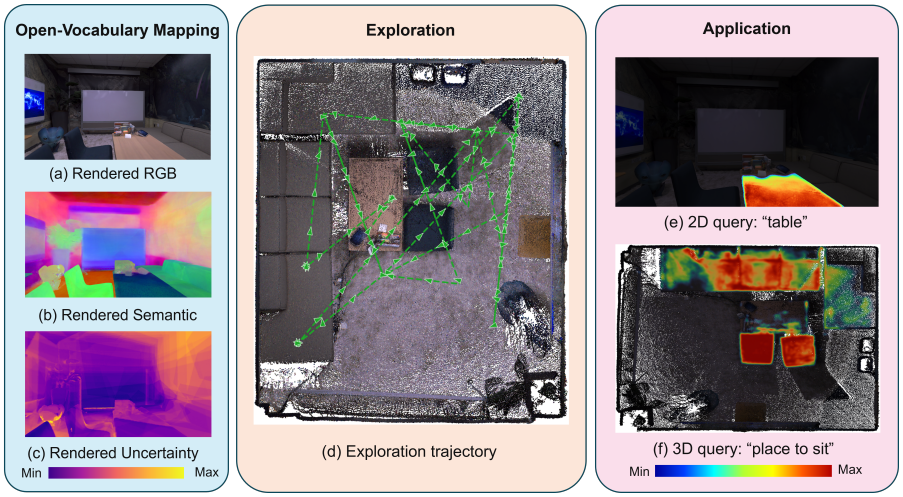


Figure 1: **ActiveLang integrates open-vocabulary 3D mapping with autonomous exploration. Left:** ActiveLang reconstructs semantic RGB-D Gaussian maps. The rendered semantic maps show coherent regions that largely follow true semantic boundaries, while learned uncertainty guides exploration. **Middle:** The autonomous trajectory overlaid on the reconstructed 3D map illustrates that exploration achieves scene coverage. **Right:** The semantic Gaussians support open-vocabulary queries in individual novel views ("table") and directly on the 3D point cloud ("place to sit").

ABSTRACT

As robots increasingly assist humans with diverse tasks, they need both geometric and semantic understanding of their surroundings. Moreover, robots often operate in unfamiliar environments and take on new tasks without knowing the relevant concepts ahead of time. This motivates language-annotated 3D maps that support open-vocabulary scene understanding and human–robot interaction. We introduce ActiveLang, an autonomous system for active open-vocabulary 3D mapping with semantic-uncertainty-guided exploration. ActiveLang performs online language-feature adaptation on a compact dual-Gaussian representation to jointly reconstruct scene geometry, appearance, and open-vocabulary semantics with modest memory overhead. Its planner efficiently selects informative viewpoints, enabling effective mapping with fewer observations and lower computational cost. Experiments on Replica and ScanNet++ demonstrate substantial improvements in 2D and 3D open-vocabulary segmentation over both online and offline baselines, highlighting that actively exploring scenes builds language-annotated 3D maps more efficiently.

1 INTRODUCTION

As embodied AI advances, robots are increasingly deployed in diverse environments and tasked with a growing range of activities. Operating effectively in these settings requires an accurate understanding of both the geometry and semantics of their surroundings. However, conventional closed-vocabulary semantic mapping methods (Li et al., 2024b), rely on *predefined sets of labels*, which may not cover the concepts needed for unfamiliar environments or new tasks, or align with the terms used by humans. To facilitate natural human–robot interaction and support robots operating in unfamiliar environments,

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.

semantic maps that are directly queryable through natural language are needed. Language-annotated 3D maps meet this requirement by associating spatial scene representations with language features (Qin et al., 2024).

Creating such maps requires multiple observations from the robot's sensors. Reconstruction can be performed offline in batch mode (Qin et al., 2024) or online on the data stream (Katragadda et al., 2025). We refer to mapping methods as *passive* when the trajectory of the sensor is imposed externally, e.g. when the robot is driven by a human operator, and as *active* when the robot determines the trajectory *autonomously* (Ahmed et al., 2023). Active mapping, therefore, must operate on streams of data that are dynamically determined as information about the environments is revealed (Lluvia et al., 2021; Placed et al., 2023). Determining the trajectory autonomously relieves human operators from the time and effort needed to drive the robot, and possibly also from the effort to familiarize themselves with characteristics of the environment and the robot. If the exploration strategy is effective, active mapping is capable of generating a map of the same quality while limiting acquisition effort compared to passive mapping.

In this paper, we address **active open-vocabulary 3D mapping** at the level of pixels and 3D primitives; see Fig. 1. This is a much more challenging than tasks addressed by prior works, none of which tackle active mapping of all objects and surfaces with an open vocabulary. Our method aims for dense semantic representations that are not restricted to foreground objects, without knowing a priori the number or type of visual categories that are present in the environment. To accomplish this goal, we must overcome the following challenges.

First, open-vocabulary mapping suffers from embedding misalignment across modalities. Unlike in closed-vocabulary settings, where the mapping between language and visual semantic embeddings is given, open-vocabulary semantics are characterized by ambiguity due to a plethora of causes, including ambiguous granularity, e.g. *table* and *desk* may be treated as the same or different categories, and multi-modality of the embeddings, e.g. the embeddings of legs, wheels and backs of different *chairs* may be similar in language space and different in visual space. High-dimensional language features also impose substantial storage, optimization, and rendering costs, motivating feature compression (Qin et al., 2024; Katragadda et al., 2025). During active exploration, however, new concepts appear progressively and observations are unevenly distributed across the scene. Learning a compact representation from this evolving stream can lead to bias in favor of frequently observed content and may blur semantic distinctions, making efficient compression and semantic preservation difficult to achieve jointly

Second, selecting observations that jointly improve geometric and semantic map quality is challenging. A viewpoint that reveals previously unseen surfaces may provide weak or inconsistent language features, while revisiting a geometrically accurate region may help resolve semantic ambiguity without expanding coverage. Designing an exploration strategy that strikes a good balance between these two criteria is hard. Several recent approaches for open-vocabulary exploration and navigation rely on frontiers (Alama et al., 2025; Padilla-Cerdio et al., 2026; Schperberg et al., 2026), possibly to bypass the formulation of an uncertainty-driven exploration criterion.

To address these challenges, we introduce ActiveLang, a closed-loop system for active open-vocabulary 3D mapping. Our main contributions are:

- **A closed-loop framework for dense language mapping and exploration.** ActiveLang incrementally reconstructs geometry, appearance, and language features while autonomously selecting its next observations. By coupling map updates with viewpoint selection, it builds language-queryable maps of the scene.
- **Compact, semantic-preserving representation.** We adapt a language-feature autoencoder as observations arrive to reduce the cost of dense semantic representation. We introduce a prototype-based adaptation objective that balances supervision across feature modes while preserving their separation during compression, reducing bias towards frequently observed content and semantic confusion.
- **Exploration guided by geometric coverage and semantic uncertainty.** Our semantic loss jointly considers the semantics rendered by the Gaussians and their predicted uncertainty, and uses the latter as a proxy for the potential informativeness of additional observations. Our next-best-view criterion combines this signal with geometric coverage and travel cost to balance exploration of unseen surfaces with refinement of uncertain semantics.

2 Related Work

Most relevant to our work are methods that address active and semantic mapping, separately or jointly, using Neural Radiance Fields (NeRF) (Mildenhall et al., 2020; Yao et al., 2026) and Gaussian Splatting (GS) (Kerbl et al., 2023; Chen & Wang, 2026) to represent the map. Surveys on semantic mapping and on using NeRF for it were published by Raychaudhuri & Chang (2025) and Nguyen et al. (2024b), respectively. We classify prior work along two axes: *mapping mode*, which may be offline when all data are available simultaneously, online when data are processed sequentially, and active when the robot determines the next data to be acquired; and type of *semantic vocabulary*, which may be foreground objects only, closed or open.

Table 1 shows representative methods from each category. For example, GSNeRF (Chou et al., 2024) is an offline, closed-vocabulary approach. Representative semantic SLAM approaches with GS mapping backbones include SGS-SLAM (Li et al., 2024b), NIDS-SLAM (Haghighi et al., 2023), SNI-SLAM (Zhu et al., 2024b), DNS-SLAM (Li et al., 2024a), SemGauss-SLAM (Zhu et al., 2024a) and Hier-SLAM (Li et al., 2025a).

Table 1: Comparison with representative passive and active mapping methods. All methods address appearance and geometric mapping. Semantic vocabulary is none (–), object-level, closed-set, or open-set.

Method	Mapping Mode	Semantic Vocabulary
GSNeRF	Offline	Closed
LangSplat	Offline	Open
OpenGaussian	Offline	Open
SplaTAM	Online	–
SGS-SLAM	Online	Closed
OVI-MAP	Online	Object, Open
Online Language Splatting	Online	Open
ActiveGAMER	Active	–
AutoNeRF	Active	Object
ActiveSGM	Active	Closed
ActiveLang (Ours)	Active	Open

Active Mapping. Lluvia et al. (2021) and Placed et al. (2023) surveyed the literature. Recently, NeRF-based representations have been used for path planning (Adamkiewicz et al., 2022) and next-best-view selection (Jin et al., 2023; Lee et al., 2022; Pan et al., 2022; Ran et al., 2023; Xue et al., 2024; Yan et al., 2023), while hybrid implicit-explicit representations have also been proposed (Zhan et al., 2022; Feng et al., 2024; Kuang et al., 2024) to reduce computational cost. GS-based uncertainty predictors are faster and more scalable. They are often combined with additional representations in approaches such as AG-SLAM (Jiang et al., 2024), RT-GuIDE (Tao et al., 2025), ActiveSplat (Li et al., 2025c), ActiveGS (Jin et al., 2025), ActiveGAMER (Chen et al., 2025a) and NextBestPath (Li et al., 2025b). AREA3D (Xu et al., 2026) relies on semantic uncertainty for exploration, but it only maps RGB-D.

Active Closed-vocabulary Mapping. These methods determine sensor trajectory autonomously, but do not face all challenges of Section 1 since labels do not need to be mapped across modalities (from language to images). Li et al. (2026) select among discrete views for semantic mapping, neglecting travel cost. Other methods focus on objects, but not surfaces (Gervet et al., 2023; Georgakis et al., 2022; Ramakrishnan et al., 2022). Zhang et al. (2024) guide exploration via semantic mutual information and properties of the SLAM pose graph, but their approach is limited to 8 classes. AutoNeRF(Marza et al., 2024) added semantics to the NeRF backbone, but is limited to 2D motion and 15 classes. ActiveSGM (Chen et al., 2025b) labels both the foreground and background, handling over 100 labels while exploring the scene guided by geometric and semantic uncertainty.

Open-vocabulary Mapping. Early methods were offline, starting with LERF Kerr et al. (2023), which endowed the volumetric rendering of NeRF with semantic features, supervised by CLIP. OpenNeRF Engelmann et al. (2024) uses pixel-wise VLM features, instead of LERF's global CLIP features, resulting in a lighter architecture. The most representative offline open-vocabulary semantic mapping method is LangSplat (Qin et al., 2024). Similar to our work, LangSplat trains a scene-specific autoencoder to capture distinctions between the categories that are present. LangSplat, however, trains the autoencoder offline on all frames, while we have to adapt it online. It also relies on SAM Kirillov et al. (2023), which we do not use. OpenGaussian (Wu et al., 2024) advocates for reasoning on the 3D splats rather than on renderings generated by them as LangSplat does. Other methods focus on detail-preserving instance segmentation either by operating on the 3D map (Alegret et al., 2026) or by consolidating 2D masks from multiple images (Lu et al., 2025; Nguyen et al., 2024a; Yan et al., 2024; Yin et al., 2026). Given semantic maps, Gao et al. (2026) support vision-language navigation, while Kim et al. (2026) generate novel views to facilitate queries.

Online formulations include LatentBKI (Wilson et al., 2025) which is based on a moving average scheme to update semantics in latent space. OpenGS-SLAM (Yang et al., 2025a) and OpenGS-Fusion (Yang et al., 2025b) infer consistent labels for the objects via the multi-view consensus of 2D

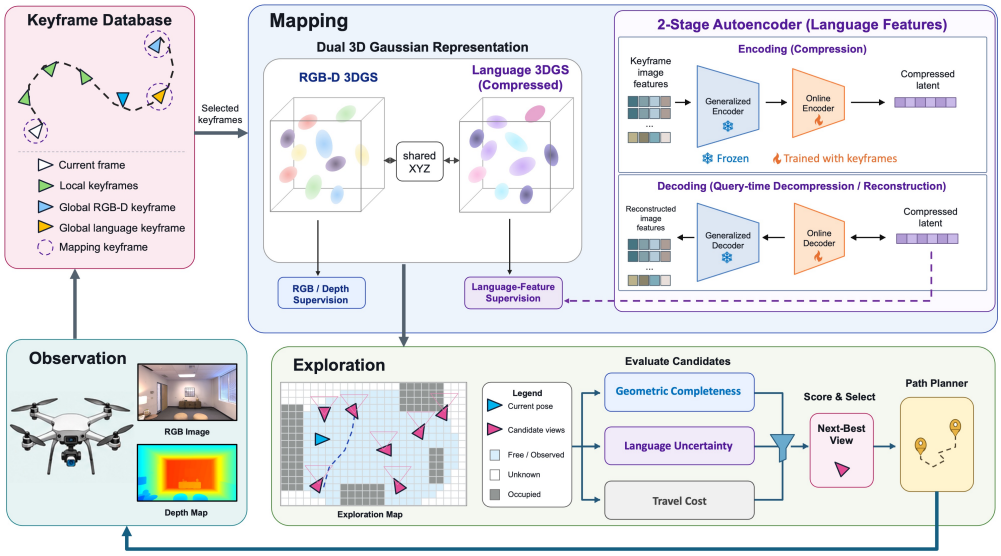


Figure 2: **Overview of the ActiveLang System.**

foundational models. OpenMonoGS-SLAM (Yoo et al., 2025) relies on MASt3R-SLAM (Murai et al., 2025) for point maps, SAM for instance masks, and CLIP to generate features for each mask, which are integrated on the Gaussians. More recently, OVI-MAP (Deng et al., 2026) build a class-agnostic 3D instance map without using SAM. Online Language Splatting (Katragadda et al., 2025) models both objects and background, aided by a trainable online autoencoder that compresses CLIP features to supervise language Gaussians that are disentangled from the ones modeling appearance.

Active Open-vocabulary Mapping. The most relevant prior work differs from ActiveLang by only modeling objects or by resorting to frontier-based exploration without formulating a semantic uncertainty criterion. OpenFrontier (Padilla-Cerdio et al., 2026) is a zero-shot navigation framework that detects frontiers in the image enabling language-driven semantic tasks without dense 3D reconstruction. RayFronts (Alama et al., 2025) distinguishes between regions of the map that are within and beyond depth sensing range, relies on frontiers for exploration, but ignores background categories. RoboAtlas (Schperberg et al., 2026) uses a mixture of experts comprising frontier-based exploration, semantic mapping and egocentric VLM modules. RoboAtlas facilitates language-driven navigation but only considers object instances.

3 METHOD

Task Definition. We consider active open-vocabulary 3D mapping in an unknown environment, where a robot must autonomously acquire observations to incrementally reconstruct scene geometry, appearance, and dense open-vocabulary semantics. Whereas passive mapping follows a predefined trajectory, our system actively determines where to observe next based on the current map. Mapping and exploration are therefore coupled: newly acquired observations update the geometric and semantic map, while the evolving map guides subsequent viewpoint selection.

System Overview. The ActiveLang system is illustrated in Fig. 2. ActiveLang forms a closed loop between two core processes: online 3D mapping and active exploration. Given a newly acquired RGB-D frame and its pose, the mapping module incrementally updates a dual-Gaussian representation of scene geometry and appearance in one branch, and semantic features and their uncertainty in the other. A two-stage autoencoder, which is finetuned online, adapts the semantic representation to the sequentially observed scene. The keyframe database retains informative RGB-D frames and provides complementary supervision for geometric, appearance, and semantic map refinement. Using the updated map, the exploration module evaluates candidate viewpoints according to geometric coverage, semantic uncertainty, and travel cost. The selected viewpoint produces a new observation that is incorporated into the map and closes the mapping–exploration loop.

3.1 3DGS Representation

Dual Gaussian Representation. We represent the scene using paired RGB-D and semantic Gaussian primitives that jointly encode geometry, appearance, and open-vocabulary semantics, as in Online Language Splatting (Katragadda et al., 2025). For each primitive i, the two branches share a common 3D center $\boldsymbol{\mu}_i$ while maintaining branch-specific scale, rotation, and opacity parameters. Specifically, following Kerbl et al. (2023), the RGB-D Gaussian is parameterized as $\mathcal{G}_i^{\text{rgb}} = \{\boldsymbol{\mu}_i, \mathbf{s}_i^{\text{rgb}}, \mathbf{q}_i^{\text{rgb}}, \alpha_i^{\text{rgb}}, \mathbf{c}_i\}$, where $\mathbf{c}_i \in \mathbb{R}^3$ denotes its color. The corresponding semantic Gaussian is represented as $\mathcal{G}_i^{\text{sem}} = \{\boldsymbol{\mu}_i, \mathbf{s}_i^{\text{sem}}, \mathbf{q}_i^{\text{sem}}, \alpha_i^{\text{sem}}, \mathbf{f}_i, u_i\}$, where $\mathbf{f}_i \in \mathbb{R}^{15}$ is a compact semantic feature and $u_i \in \mathbb{R}$ represents semantic uncertainty. Sharing only the Gaussian centers keeps the two representations geometrically aligned while allowing appearance and semantic information to use different spatial support.

RGB-D Map Optimization. The RGB-D branch provides the geometric scaffold of the map and is optimized using incoming color and depth observations by minimizing RGB L_1 and DSSIM losses together with a depth L_1 loss over valid pixels. The shared Gaussian centers are updated through RGB-D reconstruction, whereas semantic supervision updates only the semantic parameters. Our RGB-D optimization follows SplaTAM (Keetha et al., 2024) for online Gaussian mapping.

Semantic Features and Uncertainty. Each semantic Gaussian stores both a compact semantic feature and an uncertainty estimate. The online semantic mapping module described in Sec. 3.2 produces a target feature $\mathbf{f}_p^\star$ for each supervised pixel p. Rendering the semantic Gaussians produces a semantic feature $\hat{\mathbf{f}}_p$ together with a log-variance $\hat{u}_p = \log \sigma_p^2$. We jointly optimize semantic feature reconstruction and heteroscedastic uncertainty using

$$\mathcal{L}_{\text{sem}} = \frac{1}{|\Omega_S|} \sum_{p \in \Omega_S} \left[\frac{1}{2d} \exp(-\hat{u}_p) \left\| \mathbf{f}_p^\star - \hat{\mathbf{f}}_p \right\|_2^2 + \frac{1}{2} \hat{u}_p \right], \tag{1}$$

where Ω_S denotes the supervised semantic pixels and $d = 15$ is the semantic feature dimension. The first term weighs semantic feature reconstruction according to the predicted uncertainty, while the second prevents the model from trivially increasing the uncertainty to reduce loss (Kendall & Gal, 2017). The resulting semantic uncertainty is maintained as part of the online 3D map and is subsequently used for viewpoint selection in Sec. 3.3.

Rendering across Observation Scales. During exploration, the same surfaces may be viewed from substantially different distances, resulting in varying effective sampling rates. Standard 3DGS rendering may therefore exhibit scale-dependent aliasing, dilation, and erosion artifacts as the viewpoint changes. To make both appearance and semantic rendering robust to scale variations, we apply 3D smoothing and 2D Mip filtering to both Gaussian branches. The 3D filter is updated online as new observations are incorporated into the map as in Mip-Splatting (Yu et al., 2024).

3.2 Online Open-Vocabulary Semantic Mapping

Active exploration involves a temporally variable and semantically imbalanced stream of observations. As the camera moves through different regions, large homogeneous surfaces may be repeatedly observed while less frequent semantic content causes fewer updates to the autoencoder. This makes indiscriminate online adaptation undesirable, since the compact semantic representation can become biased toward dominant feature modes or lose distinctions between previously separable concepts. We therefore combine online semantic feature adaptation with feature-mode-balanced compression and semantic-aware observation selection.

Online Semantic Feature Adaptation. To maintain a compact semantic field while adapting to newly observed content, we use an online scene-adaptive feature compression module. For each observation, dense 768D semantic features are first compressed by a frozen general-purpose encoder into 32D codes. A trainable online autoencoder (OLAE) further maps these codes into the 15D features stored by the semantic Gaussians. For an intermediate feature $\mathbf{z} \in \mathbb{R}^{32}$, the online encoder and decoder produce $\mathbf{f} = E_\theta(\mathbf{z}) \in \mathbb{R}^{15}$ and $\hat{\mathbf{z}} = D_\phi(\mathbf{f})$. We optimize the reconstruction objective

$$\ell_{\text{rec}}(\mathbf{z}) = \frac{1}{32} \|\hat{\mathbf{z}} - \mathbf{z}\|_1 + \lambda_{\cos} \left[1 - \cos(\hat{\mathbf{z}}, \mathbf{z})\right], \tag{2}$$

which encourages both feature fidelity and directional consistency in the frozen feature space. We adopt the two-stage compression architecture of Online Language Splatting (Katragadda et al., 2025), while adapting its online training to the actively acquired observation stream as described below.

Prototype-Balanced and Separation-Preserving Adaptation. We address two failure modes of online semantic compression: feature-mode imbalance and compression-induced mode collapse. To reduce feature-mode imbalance, we maintain a bounded, class-agnostic feature-mode prototype bank $\mathcal{P}_t = \{\mathbf{b}_k\}_{k=1}^{K_t}$ in the frozen 32D feature space. Each prototype represents a recurring semantic feature mode rather than a predefined semantic class. Incoming features are matched to existing prototypes by cosine similarity and merged through exponential moving averages. Unmatched feature modes are inserted into the bank, replacing the least recently used prototype when the bank is full.

At each adaptation step, we assign equal weights to the retained prototypes and combine their reconstruction loss with the expected loss of samples from the current observation:

$$\mathcal{L}_{\text{balanced}} = \frac{\eta}{K_t} \sum_{k=1}^{K_t} \ell_{\text{rec}}(\mathbf{b}_k) + (1-\eta)\mathbb{E}_{\mathbf{z}\sim\mathcal{Q}_t}[\ell_{\text{rec}}(\mathbf{z})], \tag{3}$$

where $\mathcal{Q}_t$ denotes the pixel sampling distribution and η controls the balance between retained feature modes and the current samples. Equal prototype weighting reduces the dominance of large image regions, while pixel samples preserve within-mode variation.

Balancing feature modes does not by itself prevent distinct semantics from becoming overly similar after compression. We therefore regularize prototype pairs that are sufficiently dissimilar in the frozen feature space. Let $\mathcal{A}_t = \{(i,j) : i < j, \cos(\mathbf{b}_i, \mathbf{b}_j) < \tau_{\text{in}}\}$. For these pairs, we penalize excessive similarity between their compressed latent features:

$$\mathcal{L}_{\text{sep}} = \frac{1}{|\mathcal{A}_t|} \sum_{(i,j)\in\mathcal{A}_t} \left[\max(0,\ \cos\left(E_\theta(\mathbf{b}_i), E_\theta(\mathbf{b}_j)\right) - m)\right], \tag{4}$$

where m is a similarity margin. The OLAE is optimized using $\mathcal{L}_{\text{OLAE}} = \mathcal{L}_{\text{balanced}} + \lambda_{\text{sep}}\mathcal{L}_{\text{sep}}$. The encoded semantic features are detached before supervising the semantic Gaussians, separating feature adaptation from Gaussian map optimization.

Keyframe Management and Semantic Supervision Gating. We maintain a keyframe database with three complementary types of observations. Local keyframes support optimization of recently observed regions, global RGB-D keyframes enable re-optimizing views with poor geometric or appearance reconstruction, and global semantic keyframes play the same role for views with large semantic reconstruction errors. Both types of global keyframes are kept in a shared pool.

Not every actively acquired observation provides suitable supervision for online semantic adaptation. We therefore apply OLAE and semantic-Gaussian updates only to views with sufficient feature diversity and observation context, rejecting semantically uninformative or extreme close-up views while still using them for RGB-D reconstruction.

3.3 Semantic-Uncertainty-Guided Exploration

We sample candidate viewpoints in observed free space and score them using travel cost, geometric coverage, and semantic uncertainty; candidate-generation details are provided in the Appendix A.5.

- **Travel cost.** $l_{\text{dist}}(v) = \|\mathbf{t}_v - \mathbf{t}_t\|_2$ measures the straight-line distance between the candidate and current camera centers.
- **Geometric coverage.** We threshold the rendered silhouette to obtain a valid-pixel mask $M_v(p)$. The novel-pixel count $n(v) = \sum_{p\in\Omega}[1 - M_v(p)]$ favors views of incompletely mapped regions, as in SplaTAM (Keetha et al., 2024).
- **Semantic uncertainty.** $l_u(v)$ is the mean rendered semantic log-variance over valid pixels, favoring views whose semantic representation remains uncertain.

We prioritize geometric coverage before incorporating semantic uncertainty, which requires sufficient map support. Let $c(v) = |\Omega|^{-1}\sum_{p\in\Omega} M_v(p)$ denote view coverage. When its mean over the candidate set is below 0.5, selection uses only travel cost and novel-pixel count. Once sufficient map coverage is established, semantic uncertainty is additionally incorporated into viewpoint selection. This condition is re-evaluated at every planning step.

Table 2: **Open-vocabulary segmentation and localization in 2D and 3D on Replica and ScanNet++.** Results are averaged over 8 scenes for each dataset, evaluating held-out SLAM views, zero-shot novel views (NVS), and 3D point clouds, noting that *Trained on same frames* highlights an inherent baseline advantage where autoencoders were fitted directly to the SLAM trajectory. Complete results are shown in Appendix Tables A2 and A4.

Method	AE Settings		2D / SLAM		2D / NVS		3D	
	Trained on same frames	Type	mIoU ↑	Loc ↑	mIoU ↑	Loc ↑	mIoU ↑	mAcc ↑
Replica								
LangSplat	✓	pretrained	32.36	63.29	27.65	61.52	22.89	35.54
OnlineLangSplat	✓	online	40.55	69.94	37.81	63.85	26.56	35.76
ActiveLang	×	online	**42.05**	**73.64**	**43.66**	**70.71**	**30.91**	**41.67**
ScanNet++								
LangSplat	✓	pretrained	9.06	24.26	9.06	23.85	12.22	23.24
OnlineLangSplat	✓	online	8.43	33.35	9.22	30.18	16.24	23.71
ActiveLang	×	online	**21.01**	**38.29**	**20.13**	**33.89**	**21.48**	**30.49**

Table 3: **Reconstruction quality on Replica.** We report novel-view rendering quality and 3D geometric reconstruction.

Method	Novel-View Rendering				3D Geometry		
	PSNR ↑	SSIM ↑	LPIPS ↓	L1 (cm) ↓	Acc (cm) ↓	Comp (cm) ↓	Comp-ratio (%) ↑
LangSplat	27.55	0.95	0.15	7.08	2.48	2.58	96.35
OnlineLangSplat	26.13	0.93	0.21	7.24	2.48	6.37	80.54
ActiveSGM	30.61	0.96	0.14	1.36	**1.19**	1.59	**96.68**
ActiveGAMER	**32.02**	**0.97**	**0.11**	**1.12**	**1.19**	**1.56**	96.50
ActiveLang	31.77	**0.97**	0.13	1.17	1.84	2.50	92.92

To balance score magnitudes, we normalize the criteria across candidates: $\tilde{l}_{\text{dist}} = \text{softmax}(l_{\text{dist}})$ and $\tilde{n} = \text{softmax}(\log \max(n, 10^{-6}))$. We use $\tilde{l}_u = \text{softmax}(l_u)$ when the coverage condition is met, and $\tilde{l}_u(v) = 1$ otherwise. The next view is selected as

$$v^* = \arg\max_{v \in \mathcal{V}_t} \left(1 - \tilde{l}_{\text{dist}}(v)\right) \tilde{n}(v)\, \tilde{l}_u(v). \tag{5}$$

Candidates without valid pixels are assigned the minimum available semantic uncertainty. If no candidate contains valid pixels, the semantic term is omitted. An RRT-based planner then computes a path to the selected viewpoint.

Observation Orientation Constraints. The language-aligned visual features used in our map do not explicitly account for gravity alignment or 3D geometric consistency. Semantic predictions can therefore be sensitive to camera orientation. Tilted observations may introduce inconsistent semantic supervision for orientation-dependent categories such as floor and ceiling. We restrict the camera to 4-DoF motion, allowing 3D translation and yaw while fixing roll and pitch to maintain upright observations. Our ablation study assesses the effect of this constraint.

4 Experiments

Datasets and Evaluation. We use Habitat (Savva et al., 2019) for RGB-D rendering and a dense OpenCLIP extractor (Cherti et al., 2023) to obtain pixel-aligned semantic features. We evaluate on eight scenes from Replica (Straub et al., 2019) and eight scenes from ScanNet++ (Yeshwanth et al., 2023). For both datasets, all methods are evaluated on held-out views from the predefined capture trajectory (*SLAM*), zero-shot novel views (*NVS*), and directly on the reconstructed 3D map. For open-vocabulary semantic mapping, we use the ten most frequent semantic categories in each scene as text queries. We report 2D mean Intersection over Union (mIoU) and localization accuracy (Loc), together with 3D mIoU and mean class accuracy (mAcc). For reconstruction quality, we report geometric accuracy, completeness, and completeness ratio, as well as PSNR, SSIM, LPIPS, and depth L_1 for novel-view rendering. Details on all aspects are provided in the Appendix.

Baselines. We compare primarily with LangSplat (Qin et al., 2024) and Online Language Splatting (Katragadda et al., 2025) as representative offline and online, passive, open-vocabulary mapping

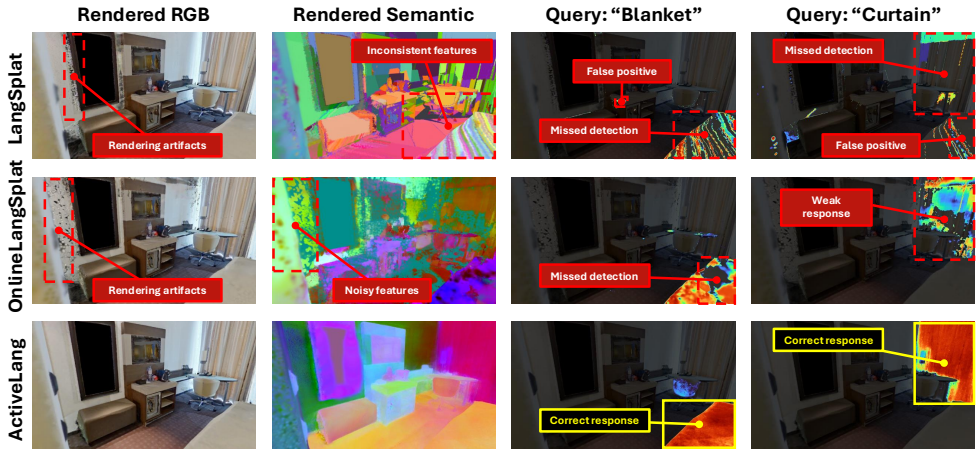


Figure 3: **Qualitative results for ScanNet++ on scene 6115eddb86.** Rows show LangSplat, OnlineLangSplat, and ActiveLang (ours), from top to bottom. Columns show rendered RGB, semantic maps, and responses to the language queries "blanket" and "curtain." Red dashed boxes highlight rendering artifacts, semantic errors, and missed, spurious, or weak query responses; yellow boxes indicate correct responses. Our method generates more complete queried object.

baselines. *Note that there are no active open-vocabulary mapping baselines to compare against.* For geometric reconstruction and novel-view synthesis, we additionally compare with ActiveSGM (Chen et al., 2025b) and ActiveGAMER (Chen et al., 2025a).

4.1 Active Open-Vocabulary Mapping

Table 2 compares ActiveLang with LangSplat and OnlineLangSplat on Replica and ScanNet++ across held-out SLAM views, zero-shot novel views, and the reconstructed 3D maps.

2D Open-Vocabulary Mapping. Across both datasets, ActiveLang consistently improves open-vocabulary segmentation and localization over the passive mapping baselines. On Replica, the advantage becomes particularly clear on strictly novel views, where ActiveLang surpasses OnlineLangSplat by $+5.9\%$ in mIoU, indicating stronger generalization beyond the observations used during mapping. The gain is substantially larger on ScanNet++, where ActiveLang improves 2D mIoU by a wide margin over both baselines, showing that the proposed mapping and exploration strategy remains effective under more complex real-world scene geometry and viewpoint variation. Qualitative comparisons in Figs 3 and A2 further show cleaner and more spatially coherent semantic fields, with fewer missed detections or spurious query responses.

3D Open-Vocabulary Mapping. The same overall trend holds when semantic quality is evaluated directly on the reconstructed 3D maps. ActiveLang achieves the highest 3D mIoU across both datasets and improves over OnlineLangSplat in semantic accuracy, demonstrating that the gains are not limited to rendered 2D views. The improvement is generally smaller in 3D than in 2D, suggesting that part of the advantage also comes from the quality of the rendered semantic field rather than only from per-Gaussian semantic assignments. Overall, the results in Table 2 show consistent gains across datasets, viewpoints, and both 2D and 3D evaluation settings.

4.2 Reconstruction Quality on Replica

We evaluate reconstruction quality primarily on Replica, whose high-quality geometry and controlled rendering setup provide reliable ground truth for both novel-view synthesis and 3D reconstruction.

Novel-View Synthesis. As shown in Table 3, ActiveLang remains competitive with more geometry-focused mapping systems while substantially outperforming the open-vocabulary baselines. Compared with LangSplat and OnlineLangSplat, ActiveLang improves PSNR by more than 4 dB while reducing depth error by approximately 6 cm. Compared with geometry-focused methods, ActiveLang surpasses ActiveSGM and performs comparably to ActiveGAMER. These results show that incor-

Table 4: **Component ablation on Replica.** Averaged across four scenes with the best score in **bold** and the second best underlined. The results show that while global keyframe variants offer minor metric trade-offs, ablating language supervision or the 4-DoF orientation constraint leads to performance drops.

	Keyframes			Lang. supervision		Cam.	Reconstruction				2D / NVS		3D	
Variant	Local	RGB-D	Lang.	Gate	Proto.	DoF	PSNR ↑	SSIM ↑	LPIPS ↓	L1 (cm) ↓	mIoU ↑	Loc ↑	mIoU ↑	mAcc ↑
w/o prototype bank	✓	✓	✓	✓	×	4	**33.16**	**0.97**	**0.13**	0.86	38.44	65.05	21.12	30.13
w/o supervision gate	✓	✓	✓	×	✓	4	32.86	0.96	0.14	0.94	38.13	64.18	21.13	29.74
w/o orientation constraint	✓	✓	✓	✓	✓	6	30.71	0.95	0.16	1.19	32.33	61.34	22.39	36.08
w/o semantic uncertainty	✓	✓	✓	✓	✓	4	32.97	**0.97**	**0.13**	0.89	39.88	67.11	27.09	**39.27**
local keyframes only	✓	×	×	✓	✓	4	32.19	0.96	0.15	1.18	**41.04**	**70.01**	**27.46**	38.93
ActiveLang (full)	✓	✓	✓	✓	✓	4	33.07	**0.97**	**0.13**	**0.85**	40.91	67.74	27.34	38.81

porating dense open-vocabulary semantics and semantic-aware exploration does not substantially compromise appearance or depth reconstruction quality.

3D Geometric Reconstruction. As shown in Table 3, ActiveLang achieves the best geometric accuracy among the open-vocabulary mapping baselines and substantially improves completeness over OnlineLangSplat. Geometry-focused active mappers achieve higher completeness, partly because they can exploit unconstrained 6-DoF viewpoints to inspect non-horizontal surfaces. In contrast, ActiveLang restricts camera motion to 4 DoF to maintain more consistent semantic observations. Ablating this constraint to 6 DoF yields modest geometric gains but substantially degrades open-vocabulary segmentation by 8.6% mIoU in 2D and 5.0% mIoU in 3D. Additional novel-view synthesis results on ScanNet++ are reported in Table A4 in the Appendix, where ActiveLang also remains competitive.

4.3 Ablation Study

Table 4 shows our ablations of the main components of ActiveLang across four Replica scenes.

Online Semantic Adaptation. Removing the prototype-based adaptation mechanism reduces 2D novel-view mIoU from 40.91% to 38.44% and 3D mIoU from 27.34% to 21.12%. Removing semantic supervision gating causes similar reductions in 2D and 3D mIoU. These results show that balancing persistent feature modes and filtering unsuitable semantic observations are important for maintaining a reliable representation during online adaptation. Since the separation objective operates on the prototype bank, the prototype ablation evaluates the prototype-based adaptation mechanism as a whole without isolating the separation term.

Semantic-Aware Exploration. Removing semantic uncertainty from viewpoint selection causes a modest decrease in semantic performance, with 2D mIoU dropping from 40.91% to 39.88% and 3D mIoU from 27.34% to 27.09%. This indicates that semantic uncertainty provides complementary guidance beyond geometric coverage when deciding where to observe. A substantially larger degradation occurs when the 4-DoF orientation constraint is removed: 2D mIoU decreases by 8.6% and PSNR drops by approximately 2.4 dB. This result highlights the importance of controlling how the scene is observed, since upright viewpoints provide more consistent semantic supervision.

Keyframe Management. Using only local keyframes primarily affects geometric and appearance reconstruction rather than semantic accuracy. Depth error increases from 0.85 to 1.18 cm and PSNR decreases from 33.07 to 32.19 dB, while semantic metrics remain nearly unchanged. This suggests that global keyframes mainly support continual geometric and appearance refinement.

5 Conclusion

We have presented ActiveLang, an active open-vocabulary 3D mapping system combining a dual-Gaussian structure, prototype-balanced online language adaptation, and semantic-uncertainty-guided exploration. Evaluated on Replica and ScanNet++, ActiveLang outperforms existing online and offline *passive* mapping baselines in 2D and 3D open-vocabulary segmentation while matching geometry-focused mappers in visual quality at lower computational cost. Current limitations include assuming ground-truth poses are available and minor surface detail trade-offs imposed by camera orientation constraints. Future work will focus on deploying with unknown camera poses and closing the geometric gap while preserving multi-modal stability.

AI USE STATEMENT

We used generative AI tools to assist with manuscript preparation, including correcting grammatical errors and improving wording and clarity. We also used these tools to assist with figure annotation and presentation, such as adding panel labels and scene names to the visualizations in Fig. 3. These changes were limited to presentation and did not alter the underlying experimental outputs. We did not use generative AI to generate or modify any experimental results reported in this paper.

We reviewed all AI-assisted text and figure annotations for accuracy and consistency with the underlying methods and results. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, the beginning of Sec. 4 describes the simulator, datasets, baselines, and evaluation protocol used in our experiments. Appendix A.4 documents the hardware and software resources, while Appendix A.5 provides further implementation details, including the hyperparameters for each module. Together, these details specify the experimental setup and procedures needed to reproduce our results. We will release our code upon acceptance.

REFERENCES

Michal Adamkiewicz, Timothy Chen, Adam Caccavale, Rachel Gardner, Preston Culbertson, Jeannette Bohg, and Mac Schwager. Vision-only Robot Navigation in a Neural Radiance World. *IEEE Robotics and Automation Letters*, 7(2):4606–4613, 2022. 3

Muhammad Farhan Ahmed, Khayyam Masood, Vincent Fremont, and Isabelle Fantoni. Active SLAM: A review on last decade. *Sensors*, 23(19):8097, 2023. doi: 10.3390/s23198097. 2

Omar Alama, Avigyan Bhattacharya, Haoyang He, Seungchan Kim, Yuheng Qiu, Wenshan Wang, Cherie Ho, Nikhil Keetha, and Sebastian Scherer. RayFronts: Open-Set Semantic Ray Frontiers for Online Scene Understanding and Exploration. In *IROS*, pp. 5930–5937, 2025. 2, 4

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 *International Conference on 3D Vision (3DV)*, pp. 1717–1727, 2026. 3

Guikun Chen and Wenguan Wang. A Survey on 3D Gaussian Splatting. *ACM Computing Surveys*, 58(12):1–39, 2026. 3

Liyan Chen, Huangying Zhan, Kevin Chen, Xiangyu Xu, Qingan Yan, Changjiang Cai, and Yi Xu. ActiveGAMER: Active GAussian Mapping through Efficient Rendering. In *CVPR*, 2025a. 3, 8, 23, 24

Liyan Chen, Huangying Zhan, Hairong Yin, Yi Xu, and Philippos Mordohai. Understanding while Exploring: Semantics-driven Active Mapping. *NeurIPS*, 38:33134–33161, 2025b. 3, 8

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible Scaling Laws for Contrastive Language-Image Learning. In *CVPR*, pp. 2818–2829, 2023. 7, 23, 25

Zi-Ting Chou, Sheng-Yu Huang, I Liu, and Yu-Chiang Frank Wang. GSNeRF: Generalizable Semantic Neural Radiance Fields with Enhanced 3D Scene Understanding. In *CVPR*, pp. 20806–20815, 2024. 3

Zilong Deng, Federico Tombari, Marc Pollefeys, Johanna Wald, and Daniel Barath. OVI-MAP:Open-Vocabulary Instance-Semantic Mapping. In *CVPR*, pp. 12606–12616, 2026. 4

Francis Engelmann, Fabian Manhardt, Michael Niemeyer, Keisuke Tateno, and Federico Tombari. OpenNeRF: Open Set 3D Neural Scene Segmentation with Pixel-Wise Features and Rendered Novel Views. In *ICLR*, pp. 8396–8407, 2024. 3

Ziyue Feng, Huangying Zhan, Zheng Chen, Qingan Yan, Xiangyu Xu, Changjiang Cai, Bing Li, Qilun Zhu, and Yi Xu. NARUTO: Neural Active Reconstruction from Uncertain Target Observations. In *CVPR*, pp. 21572–21583, 2024. 3

Jianzhe Gao, Rui Liu, Yuxuan Xu, Tongtong Cao, Yingxue Zhang, Zhanguang Zhang, Sida Peng, Yi Yang, and Wenguan Wang. Uncertainty-Aware Gaussian Map for Vision-Language Navigation. In *ICLR*, 2026. 3

Georgios Georgakis, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, and Kostas Daniilidis. Learning to Map for Active Semantic Goal Navigation. In *ICLR*, 2022. 3

Theophile Gervet, Soumith Chintala, Dhruv Batra, Jitendra Malik, and Devendra Singh Chaplot. Navigating to Objects in the Real World. *Science Robotics*, 8(79), 2023. 3

Yasaman Haghighi, Suryansh Kumar, Jean-Philippe Thiran, and Luc Van Gool. Neural Implicit Dense Semantic SLAM. *arXiv preprint arXiv:2304.14560*, 2023. 3

Wen Jiang, Boshu Lei, Katrina Ashton, and Kostas Daniilidis. AG-SLAM: Active Gaussian Splatting SLAM. *arXiv preprint arXiv:2410.17422*, 2024. 3

Liren Jin, Xieyuanli Chen, Julius Rückin, and Marija Popović. NeU-NBV: Next Best View Planning Using Uncertainty Estimation in Image-Based Neural Rendering. In *IROS*, 2023. 3

Liren Jin, Xingguang Zhong, Yue Pan, Jens Behley, Cyrill Stachniss, and Marija Popović. ActiveGS: Active Scene Reconstruction using Gaussian Splatting. *IEEE Robotics and Automation Letters*, 2025. 3

Saimouli Katragadda, Cho-Ying Wu, Yuliang Guo, Xinyu Huang, Guoquan Huang, and Liu Ren. Online Language Splatting. In *ICCV*, 2025. 2, 4, 5, 6, 7, 23, 25

Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. SplaTAM: Splat Track & Map 3D Gaussians for Dense RGB-D SLAM. In *CVPR*, pp. 21357–21366, 2024. 5, 6, 23

Alex Kendall and Yarin Gal. What Uncertainties do we need in Bayesian Deep Learning for Computer Vision? In *NeurIPS*, 2017. 5

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 3, 5, 23

Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. LERF: Language Embedded Radiance Fields. In *ICCV*, pp. 19729–19739, 2023. 3, 25

Yu-Ji Kim, Dahye Lee, Kim Jun-Seong, Nam Hyeon-Woo, GeonU Kim, Yongjin Kwon, Yu-Chiang Frank Wang, Jaesung Choe, and Tae-Hyun Oh. SplatReasoner: Enhancing Embodied Reasoning and Grounding by Novel View Synthesis. In *ECCV*, pp. 133–151, 2026. 3

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 *ICCV*, pp. 3992–4003, 2023. 3

Zijia Kuang, Zike Yan, Hao Zhao, Guyue Zhou, and Hongbin Zha. Active Neural Mapping at Scale. In *IROS*, 2024. 3

Soomin Lee, Le Chen, Jiahao Wang, Alexander Liniger, Suryansh Kumar, and Fisher Yu. Uncertainty Guided Policy for Active Robotic 3D Reconstruction Using Neural Radiance Fields. *IEEE Robotics and Automation Letters*, 2022. 3

Boying Li, Zhixi Cai, Yuan-Fang Li, Ian Reid, and Hamid Rezatofighi. Hier-SLAM: Scaling-up Semantics in SLAM with a Hierarchically Categorical Gaussian Splatting. In *ICRA*, 2025a. 3

Kunyi Li, Michael Niemeyer, Nassir Navab, and Federico Tombari. DNS-SLAM: Dense Neural Semantic-Informed SLAM. In *IROS*, pp. 7839–7846, 2024a. 3

Mingrui Li, Shuhong Liu, Heng Zhou, Guohao Zhu, Na Cheng, Tianchen Deng, and Hongyu Wang. SGS-SLAM: Semantic Gaussian Splatting for Neural Dense SLAM. In *ECCV*, pp. 163–179, 2024b. 1, 3

Shiyao Li, Antoine Guédon, Clémentin Boittiaux, Shizhe Chen, and Vincent Lepetit. NextBestPath: Efficient 3D Mapping of Unseen Environments. In *ICLR*, 2025b. 3

Yiqian Li, Wen Jiang, and Kostas Daniilidis. Next Best View Selections for Semantic and Dynamic 3D Gaussian Splatting. In *International Conference on 3D Vision (3DV)*, pp. 1916–1926, 2026. 3

Yuetao Li, Zijia Kuang, Ting Li, Guyue Zhou, Shaohui Zhang, and Zike Yan. ActiveSplat: High-Fidelity Scene Reconstruction through Active Gaussian Splatting. *IEEE Robotics and Automation Letters*, 2025c. 3

Iker Lluvia, Elena Lazkano, and Ander Ansuategi. Active Mapping and Robot Exploration: A Survey. *Sensors*, 21(7):2445, 2021. 2, 3

Yiren Lu, Yunlai Zhou, Yiran Qiao, Chaoda Song, Tuo Liang, Jing Ma, Huan Wang, and Yu Yin. Segment then Splat: Unified 3D Open-Vocabulary Segmentation via Gaussian Splatting. *NeurIPS*, pp. 165975–165994, 2025. 3

Pierre Marza, Laetitia Matignon, Olivier Simonin, Dhruv Batra, Christian Wolf, and Devendra S Chaplot. AutoNeRF: Training Implicit Scene Representations with Autonomous Agents. In *IROS*, pp. 13442–13449, 2024. 3

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV*, pp. 405–421, 2020. 3

Riku Murai, Eric Dexheimer, and Andrew J Davison. MASt3R-SLAM: Real-time dense SLAM with 3D reconstruction priors. In *CVPR*, pp. 16695–16705, 2025. 4

Phuc Nguyen, Tuan Duc Ngo, Evangelos Kalogerakis, Chuang Gan, Anh Tran, Cuong Pham, and Khoi Nguyen. Open3DIS: Open-Vocabulary 3D Instance Segmentation with 2D Mask Guidance. In *CVPR*, pp. 4018–4028, 2024a. 3

Thang-Anh-Quan Nguyen, Amine Bourki, Mátyás Macudzinski, Anthony Brunel, and Mohammed Bennamoun. Semantically-aware Neural Radiance Fields for Visual Scene Understanding: A Comprehensive Review. *arXiv preprint arXiv:2402.11141*, 2024b. 3

Esteban Padilla-Cerdio, Boyang Sun, Marc Pollefeys, and Hermann Blum. OpenFrontier: General Navigation with Visual-Language Grounded Frontiers. In *RSS*, 2026. 2, 4

Xuran Pan, Zihang Lai, Shiji Song, and Gao Huang. ActiveNeRF: Learning where to See with Uncertainty Estimation. In *ECCV*, pp. 230–246. Springer, 2022. 3

Julio A Placed, Jared Strader, Henry Carrillo, Nikolay Atanasov, Vadim Indelman, Luca Carlone, and José A Castellanos. A Survey on Active Simultaneous Localization and Mapping: State of the Art and New Frontiers. *IEEE Transactions on Robotics*, 39(3):1686–1705, 2023. 2, 3

Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. LangSplat: 3D Language Gaussian Splatting. In *CVPR*, 2024. 2, 3, 7, 23, 25

Santhosh Kumar Ramakrishnan, Devendra Singh Chaplot, Ziad Al-Halah, Jitendra Malik, and Kristen Grauman. PONI: Potential Functions for ObjectGoal Navigation With Interaction-Free Learning. In *CVPR*, pp. 18890–18900, 2022. 3

Yunlong Ran, Jing Zeng, Shibo He, Jiming Chen, Lincheng Li, Yingfeng Chen, Gimhee Lee, and Qi Ye. NeurAR: Neural Uncertainty for Autonomous 3D Reconstruction With Implicit Neural Representations. *IEEE Robotics and Automation Letters*, 8(2):1125–1132, 2023. 3

Sonia Raychaudhuri and Angel X Chang. Semantic Mapping in Indoor Embodied AI–A Comprehensive Survey and Future Directions. *arXiv preprint arXiv:2501.05750*, 2025. 3

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A Platform for Embodied AI Research. In *ICCV*, pp. 9339–9347, 2019. 7, 23

Alexander Schperberg, Shivam K Panda, Abraham P Vinod, MK Jawed, and Stefano Di Cairano. RoboAtlas: Contextual Active SLAM. *arXiv preprint arXiv:2606.26046*, 2026. 2, 4

Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The Replica Dataset: A Digital Replica of Indoor Spaces. *arXiv preprint arXiv:1906.05797*, 2019. 7, 22

Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J Davison. iMAP: Implicit mapping and positioning in real-time. In *ICCV*, pp. 6229–6238, 2021. 24

Yuezhan Tao, Dexter Ong, Varun Murali, Igor Spasojevic, Pratik Chaudhari, and Vijay Kumar. RT-GuIDE: Real-Time Gaussian splatting for Information-Driven Exploration. *IEEE Robotics and Automation Letters*, 2025. 3

Joey Wilson, Ruihan Xu, Yile Sun, Parker Ewen, Minghan Zhu, Kira Barton, and Maani Ghaffari. LatentBKI: Open-dictionary Continuous Mapping in Visual-Language Latent Spaces with Quantifiable Uncertainty. *IEEE Robotics and Automation Letters*, 10(4):3102–3109, 2025. 3

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 *NeurIPS*, 2024. 3

Tianling Xu, Shengzhe Gan, Leslie Gu, Yuelei Li, Fangneng Zhan, and Hanspeter Pfister. AREA3D: Active Reconstruction Agent with Unified Feed-Forward 3D Perception and Vision-Language Guidance. In *CVPR*, pp. 37133–37142, 2026. 3

Shangjie Xue, Jesse Dill, Pranay Mathur, Frank Dellaert, Panagiotis Tsiotra, and Danfei Xu. Neural Visibility Field for Uncertainty-Driven Active Mapping. In *CVPR*, pp. 18122–18132, 2024. 3

Dongyu Yan, Jianheng Liu, Fengyu Quan, Haoyao Chen, and Mengmeng Fu. Active Implicit Object Reconstruction using Uncertainty-guided Next-Best-View Optimziation. *arXiv preprint arXiv:2303.16739*, 2023. 3

Mi Yan, Jiazhao Zhang, Yan Zhu, and He Wang. MaskClustering: View Consensus based Mask Graph Clustering for Open-Vocabulary 3D Instance Segmentation. In *CVPR*, pp. 28274–28284, 2024. 3

Dianyi Yang, Yu Gao, Xihan Wang, Yufeng Yue, Yi Yang, and Mengyin Fu. OpenGS-SLAM: Open-Set Dense Semantic SLAM with 3D Gaussian Splatting for Object-Level Scene Understanding. In *ICRA*, 2025a. 3

Dianyi Yang, Xihan Wang, Yu Gao, Shiyang Liu, Bohan Ren, Yufeng Yue, and Yi Yang. OpenGS-Fusion: Open-Vocabulary Dense Mapping with Hybrid 3D Gaussian Splatting for Refined Object-Level Understanding. In *IROS*, pp. 21135–21142, 2025b. 3

Mingyuan Yao, Yukang Huo, Yang Ran, Qingbin Tian, Ruifeng Wang, and Haihua Wang. Neural Radiance Field-based Visual Rendering: A Comprehensive Review. *IEEE TVCG*, 2026. 3

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: A High-Fidelity Dataset of 3D Indoor Scenes. In *ICCV*, pp. 12–22, 2023. 7, 23

Hairong Yin, Huangying Zhan, Yi Xu, and Raymond A Yeh. Semantic and 3D-Consistent Language Gaussian Splatting. In *ICRA*, 2026. 3

Jisang Yoo, Gyeongjin Kang, Hyun-kyu Ko, Hyeonwoo Yu, and Eunbyung Park. OpenMonoGS-SLAM: Monocular Gaussian Splatting SLAM with Open-set Semantics. *arXiv preprint arXiv:2512.08625*, 2025. 4

Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-Splatting: Alias-free 3D Gaussian Splatting. In *CVPR*, 2024. 5, 23

Huangying Zhan, Jiyang Zheng, Yi Xu, Ian Reid, and Hamid Rezatofighi. ActiveRMAP: Radiance Field for Active Mapping And Planning. *arXiv preprint arXiv:2211.12656*, 2022. 3

Rongge Zhang, Haechan Mark Bong, and Giovanni Beltrame. Active Semantic Mapping and Pose Graph Spectral Analysis for Robot Exploration. In *IROS*, pp. 13787–13794, 2024. 3

Siting Zhu, Renjie Qin, Guangming Wang, Jiuming Liu, and Hesheng Wang. SemGauss-SLAM: Dense Semantic Gaussian Splatting Slam. *arXiv preprint arXiv:2403.07494*, 2024a. 3

Siting Zhu, Guangming Wang, Hermann Blum, Jiuming Liu, Liang Song, Marc Pollefeys, and Hesheng Wang. SNI-SLAM: Semantic Neural Implicit Slam. In *CVPR*, pp. 21167–21177, 2024b. 3

A Appendix

This appendix provides additional experimental results, analyses, and implementation details. Sec. A.1 supplements the experiments in the main paper with additional results. Secs. A.2 and A.3 further investigate image–text feature misalignment and feature-mode imbalance with compression-induced mode collapse, respectively, providing empirical evidence for the two key challenges in active open-vocabulary mapping discussed in the introduction. Sec. A.4 describes the datasets, hardware, and open-source software used to reproduce our experiments. Finally, Sec. A.5 provides further implementation details, including the hyperparameters for each module.

A.1 Additional Results

We present additional results for both datasets in this section, including exploration trajectories of ActiveLang, per-scene rendering quality on novel views, and 2D and 3D open-vocabulary semantic evaluation. For each scene, we select the ten most frequent semantic categories in the ground-truth mesh as text queries and use the same query text for both 2D and 3D semantic evaluation.

Replica. The exploration trajectories of ActiveLang are shown as green lines in Fig. A3(a), and the text queries are listed in Table A1. Table A2 reports per-scene results for open-vocabulary semantic mapping and novel-view rendering. Our method outperforms the baselines in most scenes, particularly in the 2D semantic metrics.

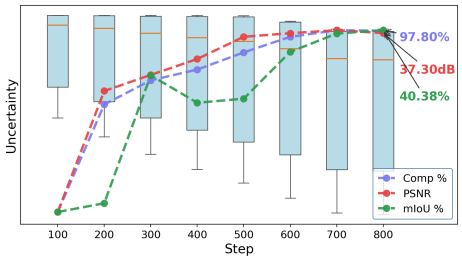


Figure A1: **Exploration progress on Replica `office1`.** Boxplots show mean rendered language uncertainty; curves show PSNR, reconstruction completeness, and 2D mIoU. Uncertainty decays shows an overall downward trend as rendering, geometry, and semantics improve during exploration.

We analyze how map quality and predicted language uncertainty evolve during exploration in Replica `office1` (Fig. A1). We save a checkpoint every 100 exploration steps and evaluate reconstruction progress using a fixed set of 20 views uniformly sampled along the SLAM trajectory. At each checkpoint, we render RGB images, semantic maps, and language uncertainty maps at these viewpoints. We evaluate RGB reconstruction using PSNR and open-vocabulary segmentation using 2D mIoU with the LERF evaluation protocol. The boxplots summarize the distribution of mean rendered language uncertainty across the evaluation views. To additionally measure geometric coverage, we compute the fraction of points sampled from the reconstructed map that lie within 5 cm of the ground-truth mesh. The three performance curves are independently normalized for visualization; endpoint annotations report their original values.

As exploration proceeds, PSNR and geometric completeness increase and gradually plateau, reaching 37.30 dB and 97.80%, respectively. Semantic quality improves overall to 40.38% mIoU, although it temporarily decreases at intermediate checkpoints, indicating that semantic refinement is less uniform than geometric coverage. Meanwhile, the median rendered language uncertainty decreases, and the lower quartile shifts downward substantially. The distribution remains broad, suggesting that

Table A1: **Scene-specific text queries for Replica.** We use the following 10 category labels in each scene for 2D and 3D open-vocabulary semantic evaluation.

Scene	Query categories
`office0`	wall, ceiling, rug, sofa, blinds, table, chair, indoor-plant, tv-screen, door
`office1`	wall, ceiling, floor, pillow, blinds, pillar, desk, door, chair, blanket
`office2`	wall, floor, ceiling, sofa, table, chair, lamp, tv-screen, door, stool
`office3`	wall, floor, ceiling, chair, sofa, table, window, tv-stand, blinds, lamp
`office4`	wall, floor, ceiling, chair, window, bench, lamp, tv-screen, door, bin
`room0`	wall, ceiling, floor, blinds, sofa, rug, chair, table, lamp, window
`room1`	wall, ceiling, floor, blinds, rug, lamp, bed, pillow, comforter, nightstand
`room2`	wall, ceiling, floor, chair, blinds, rug, table, lamp, shelf, door

Table A2: **Per-scene open-vocabulary semantic mapping and rendering quality on Replica.** Semantic metrics are evaluated on held-out SLAM views, zero-shot novel views (NVS), and 3D point clouds using the ten scene-specific query categories in Table A1. Rendering metrics (PSNR, SSIM, and LPIPS) are evaluated on novel views (NVS). Results provide per-scene breakdowns of Tables 2 and 3. Best result per scene and metric in **bold.**

		2D / SLAM		2D / NVS		3D		Rendering		
Scene	Method	mIoU ↑	Loc ↑	mIoU ↑	Loc ↑	mIoU ↑	mAcc ↑	PSNR ↑	SSIM ↑	LPIPS ↓
office0	LangSplat	30.97	63.37	27.33	52.83	**27.03**	**40.38**	29.96	0.952	0.123
	OnlineLangSplat	**42.43**	**73.80**	**36.03**	**53.66**	23.04	26.12	27.54	0.921	0.228
	ActiveLang	32.78	71.77	30.08	52.60	19.43	26.37	**35.13**	**0.979**	**0.092**
office1	LangSplat	**30.14**	**64.96**	25.68	59.90	22.65	35.39	31.34	0.963	0.137
	OnlineLangSplat	21.47	48.09	32.06	**62.48**	12.62	22.76	29.09	0.934	0.251
	ActiveLang	27.00	60.63	**39.14**	61.93	**25.73**	**45.92**	**37.23**	**0.978**	**0.115**
office2	LangSplat	38.87	**75.31**	30.07	**76.72**	21.79	**33.47**	23.92	0.930	0.194
	OnlineLangSplat	44.27	71.28	31.31	50.79	22.54	28.41	22.26	0.897	0.288
	ActiveLang	**49.99**	71.95	**45.96**	72.72	**30.17**	32.72	**29.10**	**0.962**	**0.154**
office3	LangSplat	29.43	63.36	22.25	48.04	13.53	20.35	25.87	0.932	0.224
	OnlineLangSplat	50.00	85.47	51.03	**92.12**	**35.62**	**39.99**	23.61	0.889	0.314
	ActiveLang	**51.41**	**86.95**	**52.83**	80.04	32.04	39.85	**30.46**	**0.973**	**0.134**
office4	LangSplat	28.30	58.42	19.93	58.60	16.70	32.95	30.31	0.975	**0.101**
	OnlineLangSplat	40.14	67.54	35.51	51.24	24.59	42.76	28.48	0.961	0.187
	ActiveLang	**44.71**	**75.67**	**44.78**	**74.36**	**34.32**	**50.91**	**32.76**	**0.981**	0.103
room0	LangSplat	32.45	59.13	34.86	64.25	30.99	42.88	**30.01**	**0.960**	**0.123**
	OnlineLangSplat	44.99	73.46	44.50	79.92	36.22	51.08	28.76	0.947	0.167
	ActiveLang	**47.43**	**79.53**	**51.47**	**82.99**	**36.95**	**53.29**	29.60	0.953	0.162
room1	LangSplat	35.75	60.07	34.91	63.98	24.57	**36.67**	28.18	**0.957**	**0.112**
	OnlineLangSplat	34.81	59.13	37.03	60.62	24.24	31.24	28.18	0.955	0.114
	ActiveLang	**41.80**	**70.43**	**42.96**	**73.45**	**27.24**	29.67	**30.31**	0.956	0.139
room2	LangSplat	32.94	61.71	26.19	**67.83**	25.87	42.22	20.81	0.921	0.162
	OnlineLangSplat	39.66	59.21	34.98	59.97	33.61	43.76	21.09	0.921	0.162
	ActiveLang	**41.26**	**72.22**	**42.05**	67.60	**41.38**	**54.62**	**29.55**	**0.953**	**0.134**

uncertainty is resolved unevenly across the evaluation views. This overall reduction is desirable: as the agent acquires additional informative observations, the map should become better constrained and its semantic predictions less uncertain. The concurrent improvements in reconstruction and segmentation are consistent with this interpretation.

Figure A2 presents a qualitative comparison with LangSplat and OnlineLangSplat from a viewpoint in Replica `room0`, showing rendered RGB images, PCA visualizations of rendered language features, and relevancy maps for the queries "sofa" and "window." In the highlighted regions, our method reduces spurious responses to the "sofa" query and produces more consistent responses across the "window" regions. For the PCA visualizations, we project the 3D features of LangSplat and the 32D features of OnlineLangSplat and our method onto their respective three principal components. PCA is fitted independently for each method, so the resulting colors are not directly comparable.

ScanNet++. The exploration trajectories of ActiveLang are shown as green lines in Fig. A3(b), and the text queries are listed in Table A3. Table A2 reports per-scene results for open-vocabulary

Table A3: **Scene-specific text queries for ScanNet++.** We use the following 10 category labels in each scene for open-vocabulary semantic evaluation.

Scene	Query categories
`3f15a9266d`	wall, blinds, floor, ceiling, cabinet, bookshelf, jacket, ceiling lamp, window, monitor
`5eb31827b7`	wall, ceiling, floor, monitor, door, window, whiteboard, table, ceiling lamp, office chair
`6115eddb86`	curtain, wall, floor, bed, ceiling, shower wall, ceiling lamp, door, blanket, plant
`acd95847c5`	floor, cabinet, ceiling, wall, ceiling lamp, shelf, bookshelf, table, pipe, window
`bcd2436daf`	wall, ceiling, shower wall, floor, blanket, window, heater, ceiling lamp, bookshelf, tv
`c49a8c6cff`	wall, floor, shoes, chair, blinds, monitor, door, curtain, table, ceiling
`c50d2d1d42`	wall, floor, ceiling, ceiling lamp, refrigerator, shelf, chair, monitor, blinds, whiteboard
`fb5a96b1a2`	ceiling, floor, wall, bookshelf, whiteboard, table, cabinet, doorframe, ceiling lamp, window frame

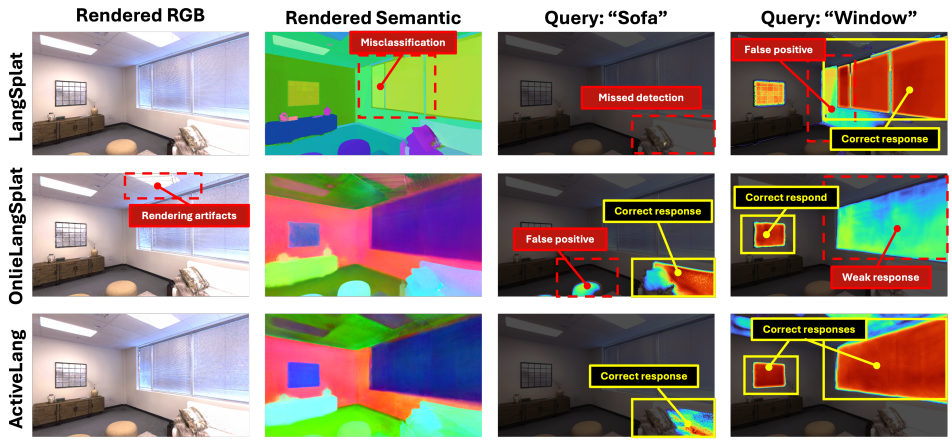
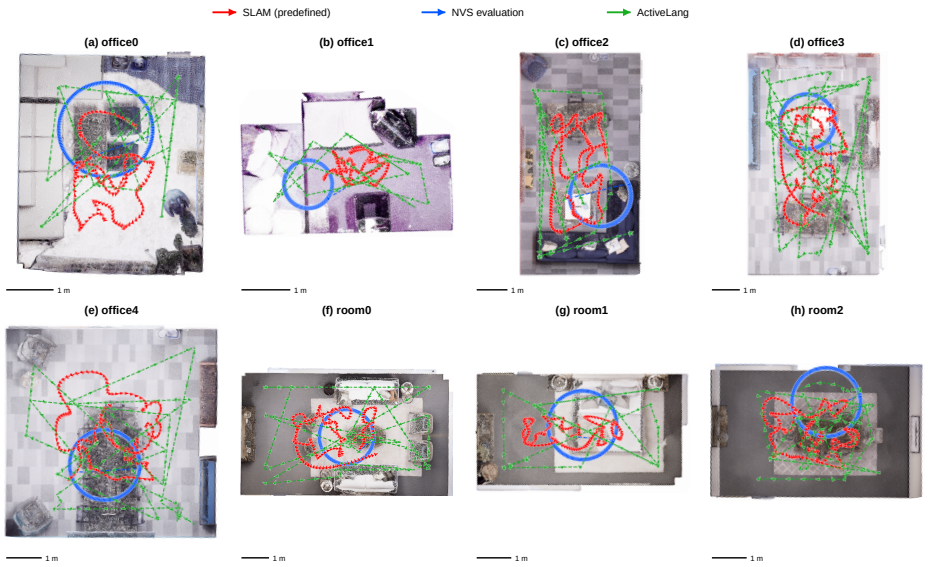


Figure A2: **Qualitative comparison on a view from Replica room0.** Rows show LangSplat, OnlineLangSplat, and ActiveLang (ours), from top to bottom. Columns show rendered RGB, semantic maps, and responses to the language queries "sofa" and "window." Red dashed boxes highlight rendering artifacts, semantic errors, and missed, spurious, or weak query responses; yellow boxes indicate correct responses. In the highlighted regions, our method reduces spurious responses to "sofa" and responds consistently to both regions queried as "window."

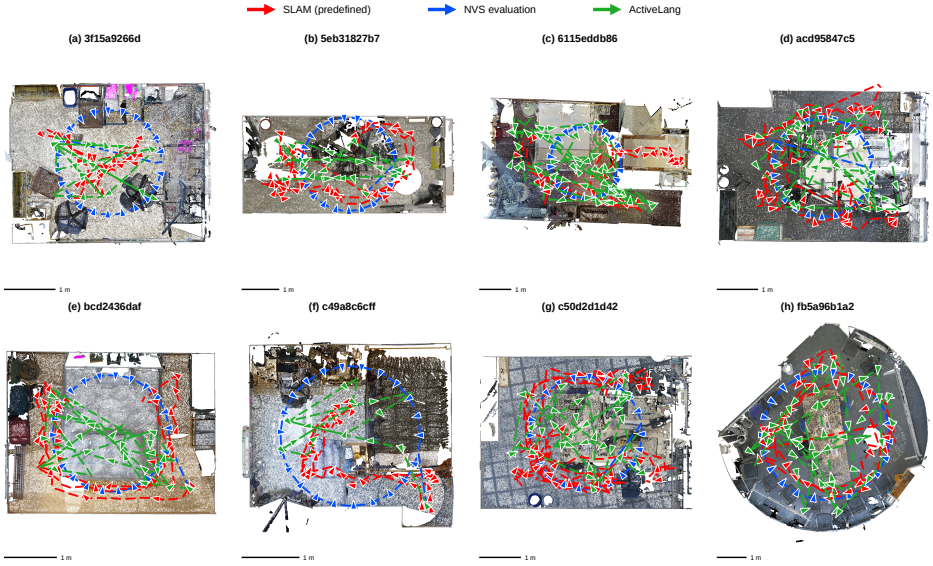
Table A4: **Per-scene open-vocabulary semantic mapping and rendering quality on ScanNet++.** Semantic results provide a per-scene breakdown of Table 2. 2D metrics are computed on 50 held-out SLAM views and a 180-view novel-view orbit (NVS), and 3D metrics against the ground-truth semantic mesh, using the ten scene-specific query categories in Table A3. Rendering metrics (PSNR, SSIM, and LPIPS) are evaluated on the 180-view novel-view orbit (NVS). Best result per scene and metric in **bold**.

		2D / SLAM		2D / NVS		3D		Rendering		
Scene	Method	mIoU ↑	Loc ↑	mIoU ↑	Loc ↑	mIoU ↑	mAcc ↑	PSNR ↑	SSIM ↑	LPIPS ↓
3f15a9266d	LangSplat	10.25	25.82	11.36	24.74	9.85	18.61	26.23	**0.961**	**0.109**
	OnlineLangSplat	16.38	48.68	15.31	49.14	19.07	25.86	24.30	0.921	0.209
	ActiveLang	**33.47**	**54.03**	**35.78**	**53.66**	**24.80**	**36.55**	**26.57**	0.954	0.121
5eb31827b7	LangSplat	16.28	25.68	10.09	28.20	10.93	24.29	17.47	0.782	0.355
	OnlineLangSplat	16.74	57.27	12.34	40.02	17.37	**25.80**	16.47	0.686	0.514
	ActiveLang	**35.11**	**64.27**	**25.21**	**55.36**	**18.86**	22.60	**19.61**	**0.828**	**0.277**
6115eddb86	LangSplat	8.67	32.50	8.49	30.05	9.55	18.54	27.53	0.938	0.174
	OnlineLangSplat	17.50	56.75	14.84	41.11	18.49	28.20	22.26	0.814	0.385
	ActiveLang	**37.08**	**58.30**	**33.34**	**53.77**	**25.98**	**38.27**	**30.21**	**0.969**	**0.114**
acd95847c5	LangSplat	7.42	19.34	8.54	19.67	9.47	**18.96**	**30.13**	**0.973**	**0.107**
	OnlineLangSplat	7.88	30.89	6.75	32.87	9.83	14.84	24.49	0.825	0.407
	ActiveLang	**20.14**	**39.47**	**22.82**	**35.91**	**12.15**	15.68	28.09	0.963	0.132
bcd2436daf	LangSplat	11.34	32.83	18.00	44.38	12.11	21.02	31.81	0.960	0.116
	OnlineLangSplat	5.94	46.19	21.34	51.80	20.20	27.53	25.99	0.862	0.316
	ActiveLang	**35.28**	**60.64**	**39.68**	**63.40**	**25.68**	**32.14**	**32.58**	**0.967**	**0.113**
c49a8c6cff	LangSplat	14.85	38.43	13.91	35.24	15.54	**27.15**	**28.09**	**0.960**	**0.118**
	OnlineLangSplat	11.48	32.23	11.14	**37.73**	16.66	22.51	24.93	0.913	0.240
	ActiveLang	**24.55**	**45.52**	**20.23**	33.01	**19.17**	26.39	26.91	0.951	0.141
c50d2d1d42	LangSplat	12.61	44.48	15.42	**53.05**	17.49	34.93	**30.69**	**0.973**	**0.108**
	OnlineLangSplat	12.54	56.48	9.42	40.68	13.51	21.89	23.83	0.818	0.406
	ActiveLang	**27.83**	**61.23**	**32.24**	51.50	**22.68**	**37.03**	28.73	0.961	0.127
fb5a96b1a2	LangSplat	17.68	44.45	20.84	54.13	12.83	22.45	**26.91**	**0.958**	**0.145**
	OnlineLangSplat	10.16	46.81	5.17	55.22	14.77	23.09	22.78	0.864	0.325
	ActiveLang	**33.46**	**59.10**	**36.07**	**64.34**	**22.54**	**35.29**	25.83	0.945	0.169

semantic mapping and novel-view rendering. Our method achieves the highest 2D and 3D mIoU in all eight scenes and the highest localization accuracy on the SLAM views, demonstrating consistent semantic improvements across diverse real-world scenes. Rendering performance varies across scenes, with our method achieving the best PSNR, SSIM, and LPIPS in three of the eight scenes.



(a) Trajectory comparison in the eight Replica scenes



(b) Trajectory comparison in the eight Scannet++ scenes

Figure A3: Red, blue, and green denote the predefined SLAM trajectory, the novel-view evaluation trajectory, and the trajectory generated by ActiveLang, respectively. Our active trajectories extend beyond the regions covered by the predefined trajectories, exploring a broader range of viewpoints throughout each room. Each panel includes a 1 m scale bar.

A.2 Analysis of Image–Text Alignment in Semantic Features

Experimental setup. To further investigate the embedding misalignment challenge identified in the introduction, we examine whether image features align with the text embeddings of their corresponding semantic categories. We analyze 768D OpenCLIP features in the `office0` scene of the Replica dataset. We use the 2,000 RGB frames along the SLAM trajectory and their ground-truth semantic labels. For each frame, we extract semantic image regions using these labels and group the resulting sub-images by category. We retain the ten most frequent categories for this analysis. For each category, we compute a category-level image embedding by averaging the OpenCLIP image embeddings of its sub-images. We also encode the corresponding ground-truth category name using the OpenCLIP text encoder.

Similarity matrices. Let $\mathcal{R}_c$ denote the collection of sub-images assigned to category c, and let f_{img} and f_{text} denote the OpenCLIP image and text encoders, respectively. The category-level image and text embeddings are

$$\boldsymbol{\mu}_c = \frac{1}{|\mathcal{R}_c|} \sum_{r \in \mathcal{R}_c} f_{\text{img}}(r), \qquad \mathbf{t}_c = f_{\text{text}}(\ell_c), \tag{A6}$$

where ℓ_c is the category name. We construct two similarity matrices:

$$S^{\text{II}}_{ij} = \cos(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j), \qquad S^{\text{IT}}_{ij} = \cos(\boldsymbol{\mu}_i, \mathbf{t}_j). \tag{A7}$$

The first measures similarity between category-level image embeddings, while the second measures cross-modal similarity between image embeddings and category names.

Fig. A4(a) shows substantial similarity between the average image embeddings of different categories. The smallest off-diagonal elements are greater than 0.7. In the image–text matrix (Fig. A4(b)), the corresponding category name does not always attain the highest similarity. In particular, the average image embeddings of *rug*, *blinds*, *table*, and *floor* are more similar to the text embedding of *wall* than to their corresponding category-name embeddings.

This analysis illustrates that category-averaged visual features can remain ambiguous despite being embedded in a language-aligned space. Even when semantic regions are identified using ground-truth labels, nearest-text matching does not consistently recover their category names in the scene. These results motivate careful treatment of semantic distinctions when constructing language-queryable maps. The analysis evaluates category-averaged features rather than individual regions and uses features before compression; it therefore does not account for additional effects introduced by feature compression or online adaptation.

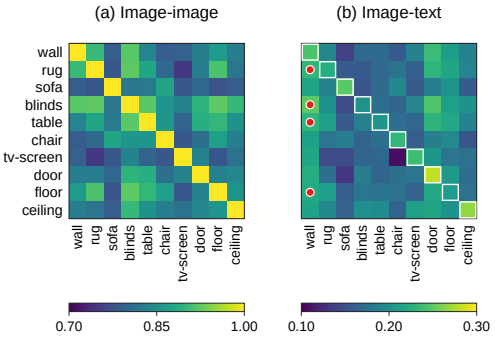


Figure A4: **Category-level OpenCLIP similarity in Replica `office0`.** We aggregate image embeddings from ground-truth semantic regions across 2,000 frames for the ten most frequent categories. (a) Cosine similarity between category-averaged image embeddings. (b) Cosine similarity between category-averaged image embeddings (rows) and category-name text embeddings (columns). White boxes indicate corresponding image–text category pairs; red dots mark row maxima attained by an incorrect text category.

A.3 Semantic Preservation in Online Open-Vocabulary Mapping

Active exploration exposes an online open-vocabulary mapper to an evolving and uneven distribution of observations. Newly encountered semantic content may initially occupy few pixels or appear in only a few views, while previously observed surfaces continue to dominate the training data. We investigate how feature compression can weaken semantic distinctions under these conditions, and how this motivates ActiveLang's prototype-balanced and separation-preserving adaptation, together with semantic supervision gating (Sec. 3.2). We combine a stage-wise compression diagnostic with a comparison of final-map semantic purity on Replica room0.

Definition and evaluation protocol. We define *purity* as point-wise semantic classification accuracy on the reconstructed Gaussian map. Ground truth supplies a reference label through nearest-neighbor matching; predictions come exclusively from the decoded language features stored in the map. For this experiment, we retain Gaussians with opacity at least 0.1 and subsample their centers using a 1 cm voxel grid. We construct the reference point cloud by back-projecting labeled RGB-D observations, sampling every fifth frame with up to 4,096 pixels per frame, and applying the same voxel size.

For a retained center $\mathbf{x}_i$, let $\mathbf{z}_i$ denote its decoded, normalized language feature, and let $\mathbf{t}_c$ be the normalized text embedding of category c. The prediction for $\mathbf{x}_i$ is

$$\hat{y}_i = \arg\max_{c \in \mathcal{C}} \mathbf{z}_i^{\mathsf{T}} \mathbf{t}_c. \tag{A8}$$

The candidate set $\mathcal{C}$ contains categories represented by at least 50 points in the sampled reference cloud. The ActiveLang and OnlineLangSplat runs both end up with 23 candidate categories, which we adopt instead of restricting this diagnostic to the ten query categories used in the main evaluation. Let $j(i)$ index the nearest reference point, with position $\mathbf{x}_{j(i)}^{\mathrm{GT}}$ and label $y_{j(i)}^{\mathrm{GT}}$. We evaluate only spatially matched centers, $\mathcal{M} = \{i : \|\mathbf{x}_i - \mathbf{x}_{j(i)}^{\mathrm{GT}}\|_2 \leq 0.05\,\mathrm{m}\}$, and compute

$$\mathrm{Purity} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \mathbb{1}\left[\hat{y}_i = y_{j(i)}^{\mathrm{GT}}\right]. \tag{A9}$$

Thus, purity measures semantic agreement on the spatially matched portion of the reconstructed map. It does not measure scene coverage, and centers farther than 5 cm from the reference cloud are excluded instead of being counted as semantic errors.

Feature-mode imbalance and compression-induced collapse. Pixel-sampled reconstruction losses weigh feature modes according to their frequency in the training observations. A small image region or a newly observed category with few supporting views therefore contributes relatively little to the adaptation objective. With a compact shared latent representation, reducing average reconstruction error need not preserve the distinctions needed to classify every category correctly. For an underrepresented mode, compression can alter feature directions enough that another category's text embedding becomes the nearest match. At the prediction level, this appears as *semantic absorption*: content that was distinguishable before compression is decoded as another category. This observation motivates investigating compression-induced mode collapse, although classification errors alone do not establish that the latent codes have collapsed. Here, underrepresentation concerns exposure in the online training distribution, rather than physical object size or final-map point count.

Table A5 compares OnlineLangSplat and ActiveLang using compression diagnostics for the two-stage AE on Replica room0. For each method, sampled image features are classified before compression (Raw), after being encoded-decoded by the frozen AE (AE), and after being encoded-decoded by the full AE+OLAE (Full), all in the 768D decoded feature space. Samples and candidate labels are held fixed across stages within each run. The two runs report identical per-class sample counts and identical Raw and AE accuracies.

Compression changes semantic decisions unevenly. The frozen AE reduces lamp accuracy from 55.3% to 0.7%, table from 84.9% to 21.5%, and picture from 96.7% to 27.3%. Other categories survive this stage but are misclassified during online compression: ceiling falls from 87.5% after AE to 0.0% in both methods, with floor the most frequent prediction. OnlineLangSplat's sofa accuracy falls from 87.4% to 24.0%, with wall becoming its most frequent prediction, whereas floor and wall retain 99.9% and 96.3%. Thus, a reconstruction-based compression objective can preserve some categories while redirecting others toward competing text labels.

ActiveLang improves Full-stage accuracy for table from 27.3% to 55.2% and picture from 63.8% to 76.7% relative to OnlineLangSplat. For table, the most frequent prediction changes from floor to the correct label. Window also improves from 31.7% to 36.2%, while cushion remains close at 89.1% versus 89.4%. These gains are selective: sofa falls to 0.0% in ActiveLang, and lamp and ceiling remain at zero in both methods. Macro accuracy increases from 44.36% to 45.74%, a gain of 1.38 percentage points across the ten reported categories.

These results identify compression as a source of classification errors and show partial recovery of particular semantic distinctions with ActiveLang. They are consistent with the feature-mode imbalance mechanism above, but do not record exposure when a category first appears. Diagnostic sample counts are not training frequencies, and the comparison of complete systems does not isolate the effects of balancing, separation, or gating.

Connection to prototype-balanced and separation-preserving adaptation. The two-stage architecture in Sec. 3.2 compresses 768D semantic features into frozen 32D codes and then into the 15D codes stored by the semantic Gaussians. Our contribution concerns how this online compression module learns from actively acquired observations. In Sec. 3.2, we address feature-mode imbalance with a bounded, class-agnostic prototype bank in the frozen 32D space. The balanced objective assigns equal weight to retained prototypes within its bank term, reducing the dependence of a mode's contribution on image-region area. Current-observation pixels complement this term by preserving

Table A5: **Compression-chain classification accuracy on Replica room0.** Raw, AE, and Full denote classification in the original 768D feature space, after being encoded-decoded by the frozen AE, and after being encoded-decoded by the full AE+OLAE, respectively. Each method evaluates the same samples and candidate labels across stages. Both runs report identical sample counts and Raw/AE accuracies. Scores are percentages; N is the diagnostic sample count, and Mode is the most frequent Full-stage prediction. Bold marks the higher Full accuracy for each class across methods (ties are unbolded). Macro averages assign equal weight to the ten reported classes. These image-feature measurements are distinct from 3D map purity.

(a) OnlineLangSplat

Class	N	Raw	AE	Full	Mode
cushion	1,000	93.7	83.5	89.1	cushion
floor	1,000	94.0	99.3	**99.9**	floor
lamp	1,000	55.3	0.7	0.0	wall
sofa	1,000	91.9	87.4	**24.0**	wall
wall	1,000	93.9	95.7	**96.3**	wall
window	1,000	10.2	6.8	31.7	wall
ceiling	800	94.9	87.5	0.0	floor
table	776	84.9	21.5	27.3	floor
rug	686	67.5	46.6	**11.5**	floor
picture	600	96.7	27.3	63.8	picture
Macro	–	78.3	55.6	44.4	–

(b) ActiveLang

Class	N	Raw	AE	Full	Mode
cushion	1,000	93.7	83.5	**89.4**	cushion
floor	1,000	94.0	99.3	96.8	floor
lamp	1,000	55.3	0.7	0.0	wall
sofa	1,000	91.9	87.4	0.0	floor
wall	1,000	93.9	95.7	93.6	wall
window	1,000	10.2	6.8	**36.2**	wall
ceiling	800	94.9	87.5	0.0	floor
table	776	84.9	21.5	**55.2**	table
rug	686	67.5	46.6	9.5	floor
picture	600	96.7	27.3	**76.7**	picture
Macro	–	78.3	55.6	45.7	–

within-mode variation. A prototype represents a feature mode, not a ground-truth category, so this mechanism does not require category labels during mapping.

Equal weighting alone does not constrain what happens to distinct modes after compression. The separation objective therefore penalizes excessive latent cosine similarity between prototypes that are sufficiently dissimilar in the frozen space. It complements balancing by explicitly discouraging compression-induced mode collapse. Both objectives operate on the OLAE; its encoded targets are detached when supervising the semantic Gaussians. This separates feature adaptation from map optimization and makes preservation of the semantic targets an explicit part of the mapping pipeline.

Semantic supervision gating. Active exploration can also produce observations dominated by a few feature modes or close-up views with insufficient semantic context. These observations can provide poor supervision even when the loss balances the retained prototypes. The gates in Sec. 3.2 therefore control which observations update the OLAE and supervise the semantic Gaussians. Feature diversity governs eligibility for semantic updates, while the median-depth check excludes extreme close-up views. The forced OLAE update after repeated diversity-based rejections avoids prolonged adaptation pauses without bypassing the semantic-map gate (see Sec. A.5).

Purity by Gaussian creation time. At the evaluated checkpoint, we group sampled Gaussians using the deciles of their stored creation timestamps and compute purity within each group. Decile 1 contains the oldest Gaussians and decile 10 the newest. All groups are evaluated using the map and decoder from the same checkpoint: the curve is therefore a comparison of creation-time cohorts, not a record of accuracy measured throughout training. The decile boundaries are computed separately for each run. In particular, active and predefined trajectories can observe different regions at the same relative stage, so corresponding deciles do not contain the same surfaces or semantic classes.

Comparison with online compression. OnlineLangSplat exhibits a decline toward later-created cohorts, despite a temporary increase in cohort 6. Purity is 76.9% and 78.3% in the first two cohorts, but only 12.9%, 26.5%, 12.8%, and 19.5% in cohorts 7–10. This pattern reveals poor semantic agreement among many later-created Gaussians in the evaluated map. It does not, on its own, establish that previously learned features deteriorated over time.

ActiveLang's profile remains non-monotonic: purity falls to 18.1% and 18.9% in cohorts 3 and 4, then rises to 48.3% and 59.7% in cohorts 5 and 6. It exceeds OnlineLangSplat in every cohort from 5 through 10. The newest cohort reaches 55.6%, compared with 19.5% for OnlineLangSplat. This supports improved semantic agreement in later-created portions of the map under the complete ActiveLang system.

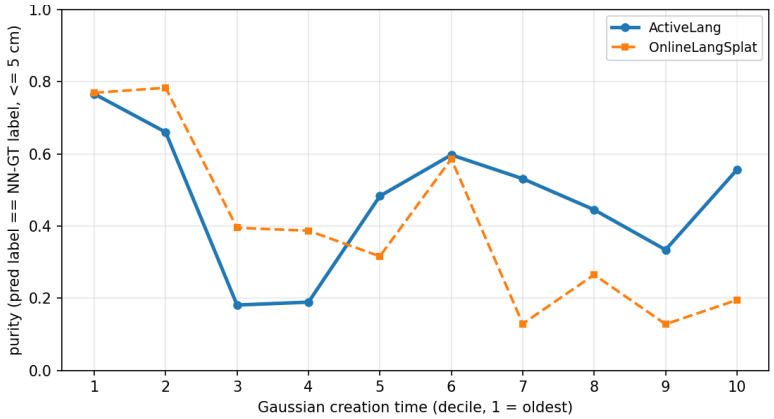


Figure A5: **Final-map semantic purity by Gaussian creation time on Replica room0.** Decile 1 contains the oldest Gaussians and decile 10 the newest. All cohorts are evaluated at the same checkpoint using nearest-neighbor ground-truth labels within 5 cm. ActiveLang shows higher purity than OnlineLangSplat in cohorts 5–10, and its newest cohort exceeds both baselines. Cohorts are defined independently for each run and do not represent the same scene content. The curves describe cohort differences, not longitudinal accuracy trajectories.

These results suggest that compression can exacerbate semantic ambiguity, with sparsely observed categories becoming confused with background categories such as walls. ActiveLang partially mitigates this effect, consistent with the intended role of its prototype bank in preserving underrepresented feature modes.

A.4 Assets Used and Reproducibility

A.4.1 Experimental result reproducibility

We describe the full system in Sec. 3 and the experimental protocol in Sec. 4; per-scene results and the scene-specific text queries for both datasets are reported in Sec. A.1 ActiveLang is built upon open-source software listed in Sec. A.4.3. More implementation details are listed in Sec. A.5. Our code will be released upon acceptance.

A.4.2 Compute resources for Experiments

All experiments were performed on two NVIDIA RTX A6000 GPUs (48 GB each). The first GPU hosts the dual 3DGS backbone: RGB-D Gaussian optimization, the gradient-based per-Gaussian language-field training (which shares the mapping loss), and the planner's rendering queries (geometric information gain and semantic uncertainty). The second GPU hosts the language-supervision pipeline that produces the per-pixel training targets: the dense CLIP encoder with its high-resolution refinement network, the general autoencoder, and the online adaptation of the scene-level autoencoder. The two memory footprints are strongly asymmetric: the language pipeline stays constant at $\approx$6 GB throughout a run, while the mapping GPU grows with the map, from $\approx$11 GB at initialization to $\approx$25 GB after the exploration stages on Replica and peaking at $\approx$35 GB depending on the scene size. This asymmetry motivates the two-GPU partition.

A.4.3 Licenses for existing assets

Datasets. We conduct experiments on the following publicly available datasets. We list the URL, license, and citation for each dataset below.

1. **Replica Dataset** (Straub et al., 2019)
   - URL: https://github.com/facebookresearch/Replica-Dataset
   - License: Research or Education only (https://github.com/facebookresearch/Replica-Dataset/blob/main/LICENSE)

2. **ScanNet++ Dataset** (Yeshwanth et al., 2023)
   - URL: https://kaldir.vc.in.tum.de/scannetpp/
   - License: Non-commercial research and education only (ScanNet++ Terms of Use)

Software. We use Habitat-Sim to render RGB-D observations from the dataset meshes. For mapping, we adopt SplaTAM, built on 3D Gaussian Splatting, as the backbone, and use a dense OpenCLIP extractor to produce pixel-aligned language features. We compare against LangSplat and Online Language Splatting using their released implementations. The source code for these components is available at:

1. **Habitat-Sim** (Savva et al., 2019)
   - URL: https://github.com/facebookresearch/habitat-sim
   - License: MIT
2. **3D Gaussian Splatting (3DGS)** (Kerbl et al., 2023)
   - URL: https://github.com/graphdeco-inria/gaussian-splatting
   - License: Custom, non-commercial (https://github.com/graphdeco-inria/gaussian-splatting/blob/main/LICENSE.md)
3. **SplaTAM** (Keetha et al., 2024)
   - URL: https://github.com/spla-tam/SplaTAM
   - License: BSD-3-Clause
4. **OpenCLIP** (Cherti et al., 2023)
   - URL: https://github.com/mlfoundations/open_clip
   - License: MIT
5. **LangSplat** (Qin et al., 2024)
   - URL: https://github.com/minghanqin/LangSplat
   - License: Gaussian-Splatting License, non-commercial research use (https://github.com/minghanqin/LangSplat/blob/main/LICENSE.md)
6. **Online Language Splatting** (Katragadda et al., 2025)
   - URL: https://github.com/rpng/online_lang_splatting
   - License: No repository-level license; the bundled 3DGS component is under the Gaussian-Splatting License
7. **Mip-Splatting** (Yu et al., 2024)
   - URL: https://github.com/autonomousvision/mip-splatting.git
   - License: Gaussian-Splatting License, non-commercial research use (https://github.com/autonomousvision/mip-splatting/blob/main/LICENSE.md)
8. **ActiveGAMER** (Chen et al., 2025a)
   - URL: https://github.com/oppo-us-research/ActiveGAMER.git
   - License: MIT

A.5 Implementation Details

A.5.1 Simulation and Data

Rendering. All observations are rendered in Habitat-Sim (Savva et al., 2019) at 680×1200 resolution with a pinhole camera of focal length 600 px, giving a 59.0° vertical and 90.0° horizontal field of view. Depth is stored at $1/6553.5$ m per unit, and depth readings below 0.2 m are treated as invalid.

Replica. We use 8 scenes: `office0–office4` and `room0–room2`. The passive baselines follow the standard 2,000-frame Replica trajectory (Sucar et al., 2021). We evaluate on (i) a 180-view novel-view orbit, with 36 azimuths $\times$ 5 heights on a 1.0 m-radius circle facing inward, and (ii) every 10th frame of the SLAM trajectory (200 views). Fig. A3 (a) shows the trajectories under different settings overlaid on top-down views of the scenes.

ScanNet++. We use 8 scenes: `3f15a9266d`, `5eb31827b7`, `6115eddb86`, `acd95847c5`, `bcd2436daf`, `c49a8c6cff`, `c50d2d1d42`, and `fb5a96b1a2`. We evaluate on (i) a 180-view inward-facing novel-view orbit whose radius is tuned per scene so that no view clips geometry, and (ii) every 20th frame real iPhone capture path (50 views). Trajectories under different settings are shown in Fig. A3 (b).

A.5.2 Semantic Supervision Gates

Not every actively acquired observation provides suitable supervision for online semantic adaptation. We therefore gate both OLAE updates and semantic-Gaussian supervision based on feature diversity and observation context.

For each frame, we greedily group up to 1,024 sampled and normalized 32D codes. A code forms a new feature mode if its cosine similarity to all existing mode centers is below 0.6. After initialization, a frame is used to train the OLAE only if it contains at least eight distinct feature modes; on the Replica `room0` trajectory, typical observations contain 9–18 modes (median 12). To avoid prolonged pauses in online adaptation, one OLAE update is forced after ten consecutive diversity-based rejections.

We additionally reject extreme close-up observations whose median valid depth is below 0.5 m, since such views may provide insufficient semantic context. Semantic-Gaussian supervision also requires the feature-diversity criterion to be satisfied. These gates affect only semantic learning, while RGB and depth reconstruction remain unaffected.

A.5.3 Keyframe Selection

We maintain three complementary types of observations for map refinement: local keyframes, global RGB-D keyframes, and global semantic keyframes.

Local Keyframes. Local keyframes form the 32-frame mapping window and support optimization of recently observed regions.

Global RGB-D Keyframes. Global RGB-D keyframes enable re-optimizing observations with poor geometric or appearance reconstruction. Every 100 frames, excluding the five most recent observations, we add the lowest 30% of previously unseen keyframes according to rendered-color PSNR.

Global Semantic Keyframes. Global semantic keyframes enable re-optimizing observations with high semantic reconstruction error. Every 100 frames, we additionally add the lowest 30% of previously unseen keyframes according to cosine similarity between rendered and extracted semantic features, computed at $1/4$ resolution.

The RGB-D and semantic criteria contribute to a shared global keyframe pool. A keyframe's scores are recomputed only when its rendered content is likely to have changed, namely when at least 50 new Gaussians fall inside its frustum or when a newly mapped view overlaps its surface points by more than 5%.

A.5.4 Exploration Planner

Candidate view sampling. Following ActiveGAMER (Chen et al., 2025a), we adopt a coarse-to-fine exploration strategy. We first use a coarse occupancy grid as exploration map to guide initial exploration, then densify the grid and resample unexplored voxels to guide the second exploration stage. The exploration map is a 5 cm voxel grid inside the bounding box of the scene. Candidates are sampled from newly observed free space, with viewing directions drawn from a Fibonacci lattice on the sphere. Candidates closer than 0.5 m to the nearest occupied voxel are discarded.

Scoring and termination. Candidates are scored with Eq. equation 5, where the semantic-uncertainty term is enabled once the mean view coverage reaches 0.5. A candidate leaves the pool when its novel-pixel count falls below 0.5% of the image. An exploration stage ends when the pool is empty, and a run ends after the last stage or at 2,000 steps. On ScanNet++, runs took 452–1,458 steps (889 on average), fewer than the 1,000 poses mapped by the passive baselines.

Motion and orientation. An RRT local planner advances up to 0.05 m and 10° per step. Every pose has zero roll relative to gravity. Transit poses face the next waypoint projected onto the horizontal plane and therefore also have zero pitch, while goal poses keep their sampled azimuth and pitch with roll removed.

A.5.5 Baselines

All methods share the same RGB-D Guassian mapping backbone, ground-truth poses, Mip-Splatting filters, per-scene settings, and evaluation code. They differ only in their semantic pipelines and in whether a planner chooses the views.

OnlineLangSplat. We re-implementated Online Language Splatting (Katragadda et al., 2025) on the shared backbone. It follows the passive trajectory and uses the same feature pipeline and OLAE schedule as ours, together with the dual-Gaussian representation and heteroscedastic loss without robust weighting. In place of the prototype bank, it uses a pixel replay buffer of 16 frames with 2 sampled per step. It uses neither supervision gates nor global keyframes.

LangSplat. A LangSplat (Qin et al., 2024) pipeline on the shared backbone, following the passive trajectory. Masks are produced by SAM2 (`hiera-large`, 32×32 point grid, predicted-IoU threshold 0.7, stability threshold 0.85, minimum mask area 100 px). Background-zeroed square crops are encoded by OpenCLIP (Cherti et al., 2023) ConvNeXt-L (`laion2b_s29b_b131k_ft_soup`, 768-d, the same text space as ours). A scene-specific autoencoder compresses these features to 3 dimensions ($\ell_2 + 0.001\cos$ loss, Adam with learning rate 7×10^{-4}, 100 epochs) and is trained before mapping on every 10th frame of the passive trajectory. Each Gaussian stores a 3D feature supervised with an ℓ_1 loss and shares the RGB-D geometry, without a separate semantic branch or uncertainty.

A.5.6 Evaluation Protocol

Each query is embedded with the OpenCLIP ConvNeXt-L text encoder, and its relevancy against the canonical negatives is $r = \min_{\text{neg}} \operatorname{softmax}(10\,[s_{\text{pos}}, s_{\text{neg}}])_0$ (Kerr et al., 2023).

2D segmentation and localization. Rendered features are decoded to 768 dimensions and scored per query. The relevancy map is smoothed with a 30×30 mean filter and averaged with the unsmoothed map, then min–max normalized and thresholded at 0.5 after mapping to $[-1, 1]$ and clipping. A 7×7 majority filter follows, and IoU is computed against the ground-truth mask. Localization counts a hit when the argmax of the smoothed relevancy falls inside a ground-truth bounding box.

3D segmentation. The prediction consists of all semantic Gaussians with opacity ≥ 0.1, voxel-sampled at 1 cm and labeled by relevancy > 0.5. The ground truth is the full semantic mesh, restricted to the top-10 classes with at least 50 points each. Each ground-truth point receives the label of its nearest Gaussian, from which we compute mIoU and mAcc.

Geometry and rendering. RGB-D Gaussians with opacity ≥ 0.1, voxel-sampled at 1 cm, are compared with uniform samples of the ground-truth mesh. We report accuracy and completion in cm as mean one-directional distances, and the completion ratio at a 5 cm threshold. Rendering is evaluated with PSNR, SSIM, and LPIPS on RGB, and depth ℓ_1 in cm over valid ground-truth depth.