CityLangSplat: Integrating CityGML Semantics into 3D Language Gaussian Splatting for Urban Scene Understanding

Qilin Zhang[1,2,3], Jinyu Zhu[1], Olaf Wysocki[4], Boris Jutzi[1,3]

[1] Technical University of Munich (TUM), [2] Munich Center for Machine Learning (MCML),
[3] Karlsruhe Institute of Technology (KIT), [4] University of Cambridge

Keywords: Gaussian Splatting, Open-vocabulary 3D Understanding, CityGML, Semantic Fusion, Urban Scene Understanding

Abstract

Combining visual semantics with language representations has made 3D interpretation more flexible and intuitive. Recent advances in Gaussian Splatting extend this to efficient 3D language fields supporting open-vocabulary queries. However, existing approaches show limited generalization in large urban scenes, especially for detailed building segmentation. Semantic 3D city models such as CityGML, by contrast, provide hierarchical and geometry-aligned structural semantics that complement appearance-driven visual cues. We introduce CityLangSplat, which integrates CityGML semantics into 3D Language Gaussian Splatting for urban environments. CityLangSplat rasterizes CityGML into pixel-aligned semantic maps, extracts vision-language features from SAM-derived segments and CityGML regions, and compresses both sources into a shared latent space via a lightweight autoencoder. 3D Gaussians are then optimized with a coverage-aware loss that balances accurate, building-focused CityGML supervision with broader SAM supervision, enabling geometry-aligned open-vocabulary reasoning in urban scenes. Experiments on TUM2TWIN and ZAHA datasets show consistent gains over LangSplat, with relative improvements of 22.9% in 2D and 15.1% in 3D evaluation while preserving real-time rendering. CityLangSplat provides a practical framework for combining semantic city models with language-embedded 3D Gaussian Splatting for geometry-aligned urban scene interpretation. Code will be released at `https://github.com/zqlin0521/CityLangSplat`.

1. Introduction

Recent progress in neural rendering has established 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) as an efficient explicit representation for real-time photorealistic view synthesis. Unlike implicit volumetric methods such as Neural Radiance Fields (NeRFs) (Mildenhall et al., 2021), which integrate radiance along continuous rays, 3DGS represents a scene as a set of anisotropic Gaussian primitives rendered through differentiable rasterization. Beyond novel view synthesis, 3DGS has become an effective foundation for higher-level vision tasks such as segmentation, editing, and open-vocabulary 3D understanding (Wu et al., 2024). Compared with NeRF-based methods, its explicit representation allows direct manipulation of scene elements, efficient optimization, and more straightforward semantic supervision (Ye et al., 2024). However, standard 3DGS mainly models on radiometry and geometry, without explicit modeling of semantic entities or object-level structures.

To address this limitation, recent research has increasingly integrated 2D foundation models, such as the Segment Anything Model (SAM) (Kirillov et al., 2023), and vision-language models, such as Contrastive Language Image Pretraining (CLIP) (Radford et al., 2021), into the 3DGS framework. These models transfer open-world segmentation and language-grounded reasoning into the 3D domain by distilling 2D semantic features into 3D Gaussian representations (Qin et al., 2024; Zhou et al., 2024). By associating Gaussians with language features, natural language queries can directly localize and segment objects or architectural components in reconstructed scenes. Despite this progress, achieving accurate and structured semantic segmentation of buildings in urban environments remains challenging. Complex visual phenomena such as reflections and refractions on glass surfaces, together with highly repetitive façade elements, often lead to ambiguous or inconsistent semantics in purely vision-based approaches. Concurrently, the development of smart cities has led to the widespread availability of semantic 3D building models. Among them, models in the CityGML standard provide geometrically aligned and hierarchically organized information that purely vision-based approaches cannot reliably extract. These models encode architectural semantics such as walls and windows, and are typically stored in lightweight boundary-representation (B-Rep) form, making them effective geometric and semantic priors for urban reconstruction (Zhang et al., 2025). While LoD3 CityGML models provide detailed geometry, integrating them with 3D language fields enables geometry-aligned open-vocabulary querying in photorealistic reconstructed scenes and

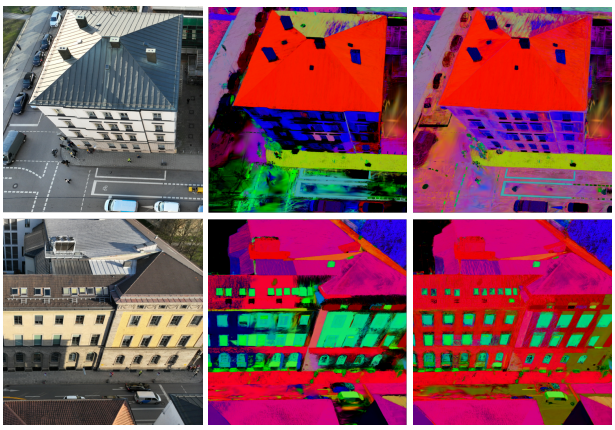

Figure 1. Comparison of rendered language features on urban façades from trained 3D Gaussian language fields. Our method CityLangSplat yields more geometrically consistent façade features by leveraging CityGML-based supervision.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-411-2026 | © Author(s) 2026. CC BY 4.0 License.

helps complement city models that may be incomplete or outdated. Consequently, this cross-modal fusion offers a highly practical direction for achieving accurate and interpretable semantic Gaussian Splatting in urban environments.

In this paper, we propose CityLangSplat, a geometry-grounded language Gaussian Splatting method that integrates CityGML-based supervision into the vision-language Gaussian Splatting paradigm. Our method first projects the hierarchical CityGML semantics into the image domain through differentiable raycasting. For each projected semantic region, class-specific textual prompts are employed to guide the extraction of CLIP embeddings, combining geometric and linguistic cues. In parallel, SAM segmentation provides purely visual masks and corresponding CLIP features. The language features from CityGML models and SAM are subsequently compressed into a shared latent space using a unified autoencoder. During training, semantic priors from CityGML and SAM are jointly optimized through coverage-aware blending, enabling geometry-aware open-vocabulary segmentation and querying. By coupling structured CityGML semantics with vision-language features, CityLangSplat enables geometry-aligned, hierarchy-aware semantic learning, leading to more accurate, interpretable open-vocabulary segmentation in urban environments, as qualitatively illustrated in Figure 1. The main contributions of this work are summarized as follows:

- We present CityLangSplat, a geometry-grounded language Gaussian Splatting framework that integrates CityGML supervision with vision language representations for hierarchical semantic understanding.
- A CityGML-guided language feature pipeline is introduced to project structured building semantics into the image domain and extract CLIP embeddings guided by class-specific text prompts.
- A dual-source training scheme jointly optimizes CityGML- and SAM-based language features through coverage-aware loss weighting, achieving geometry-aligned open-vocabulary segmentation in urban scenes.

2. Related Work

This section reviews related advances in Gaussian Splatting, urban semantic segmentation, 3D city modeling, and open-vocabulary understanding, which collectively underpin our proposed approach.

Gaussian Splatting Gaussian Splatting (GS) (Kerbl et al., 2023) has quickly evolved beyond its original goal of view synthesis, becoming a versatile framework for explicit and efficient 3D scene representation. Its parametric Gaussian primitives provide inherent geometric interpretability and enable efficient optimization compared to implicit volumetric representations such as NeRF (Mildenhall et al., 2021), which tend to exhibit limited geometric accuracy and consistency in detailed 3D reconstruction (Petrovska and Jutzi, 2024). Building upon these advantages, subsequent works have aimed to enhance the reconstruction accuracy (Jäger et al., 2025; Zhang et al., 2024b) and computational efficiency (Lee et al., 2024) of GS. Recent studies have further extended GS toward semantic understanding and multi-modal integration. These extensions demonstrate that explicit Gaussian representations can support not only high-fidelity rendering, but also richer semantic reasoning and cross-modal supervision.

Urban Semantic Segmentation Semantic understanding of urban environments has evolved from geometry-driven to data-driven approaches. Early methods relied on photogrammetric or LiDAR-based reconstruction with handcrafted geometric or contextual features (Niemeyer et al., 2014), but often failed under repetitive façade patterns, reflective materials, or strong occlusions. Deep learning methods later enabled façade-level segmentation and building-part recognition through CNNs or multi-view fusion (Wysocki et al., 2023), but face a challenging trade-off between accuracy and computational efficiency when applied to large-scale scenes. More recently, vision foundation models such as the SAM (Kirillov et al., 2023) have been employed to automatically generate large-scale façade masks that can be projected into 3D reconstructions. The continuous progress of such methods has facilitated more automated and scalable generation of structured 3D city models, contributing to the increasing availability of standardized representations.

3D City Modeling Standards Structured urban models provide geometry-aligned and hierarchically organized semantics for describing real-world scenes, enabling consistent interpretation of buildings and other urban elements. Among them, the CityGML standard (Gröger et al., 2012) defines a Level of Detail (LoD) framework for buildings, where LoD1 provides block models, LoD2 adds roof structures, and LoD3 explicitly represents façade elements such as windows. This standardized structure has led to the widespread adoption of CityGML, with more than 215 million open-access building models available worldwide (Wysocki et al., 2024). Recent work (Banno et al., 2025) has combined CityGML with 360° walkthrough videos for realistic urban visualization. However, despite their structured semantics, CityGML models are rarely integrated into modern vision frameworks, leaving a gap between urban knowledge and 3D scene understanding.

Open Vocabulary Understanding Open-vocabulary understanding aims to generalize semantic perception beyond a fixed set of predefined categories by aligning visual and linguistic representations. Advances in vision–language models (Radford et al., 2021) have demonstrated that large-scale contrastive or multimodal pretraining enables zero-shot recognition and cross-modal querying in 2D domains. Recent studies have extended it to 3D scene understanding through two complementary directions. One line of work distills language-aligned features from 2D foundation models into NeRF-based (Kerr et al., 2023; Zhang et al., 2024a) and GS-based representations (Qin et al., 2024; Zhou et al., 2024; Li et al., 2025), while another lifts 2D masks and grouping cues onto 3D Gaussian primitives for open-world grouping and segmentation (Ye et al., 2024; Wu et al., 2024). These 3D open-vocabulary approaches allow scene components to be queried and segmented using textual prompts, offering more flexible and interpretable 3D understanding. However, existing approaches remain largely data-driven and often rely on image appearance without structured priors or hierarchical semantics, which limits their reliability in complex urban environments. Integrating structured semantics from standardized 3D city models with open-vocabulary 3D representations thus offers a promising direction for achieving geometry-aware and interpretable language–driven scene understanding.

3. Methodology

Our CityLangSplat framework extends LangSplat (Qin et al., 2024) by integrating structured semantics from CityGML build-

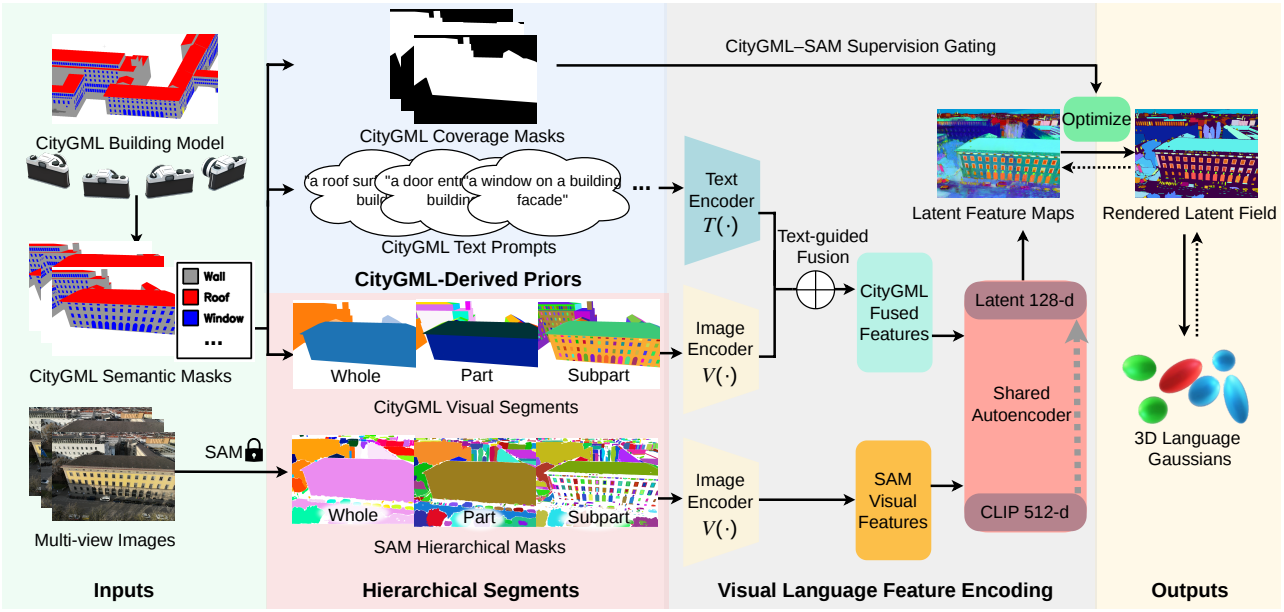

Figure 2. Overview of CityLangSplat framework. Both CityGML-derived semantic segments and SAM-based visual masks are first encoded via CLIP, and then compressed into a unified latent space through a shared autoencoder. The resulting latent feature maps provide pixel-aligned supervision for optimizing the latent vectors of 3D Language Gaussians, with CityGML coverage masks guiding the balance between geometry-grounded and visual cues.

ing models. As illustrated in Figure 2, the proposed method couples geometry-grounded CityGML supervision with vision language representations within a unified 3D Gaussian Splatting pipeline. We first generate CityGML-guided semantic features by projecting hierarchical building components into the image domain through differentiable raycasting (Sec 3.1). Subsequently, multi-scale visual features are extracted using the SAM and compressed into a shared latent space via a scene-specific autoencoder (Sec 3.2). Finally, a dual-feature learning scheme supervises the Gaussian latent features using both CityGML and SAM signals, enabling geometry-aligned, open-vocabulary urban scene understanding (Sec 3.3).

3.1 CityGML-guided Semantic Feature Generation

Given a set of calibrated images $\{I_t \in \mathbb{R}^{3\times H\times W}\}_{t=1}^{T}$ and their corresponding camera intrinsics and extrinsics $\{K_t, R_t, \mathbf{t}_t\}$, our goal is to convert the semantic CityGML building model into geometry-aligned image features that serve as structured supervision during training.

Unified Hierarchical Semantics Following the ambiguity-aware design of SAM (Kirillov et al., 2023) and its adoption in LangSplat (Qin et al., 2024), all semantic supervision in this work is organized into three hierarchical levels that capture coarse-to-fine structures. The index $l \in \{1, 2, 3\}$ denotes fine-, medium-, and coarse-scale semantics, corresponding to *Subpart*, *Part*, and *Whole*. For CityGML-derived supervision, semantic entities in the LoD3 model are assigned to this hierarchy, where façade elements such as windows correspond to *Subpart* ($l = 1$), façade surfaces such as walls to *Part* ($l = 2$), and the building instance to *Whole* ($l = 3$). This unified indexing is applied to both CityGML-derived semantic maps and SAM-derived masks, yielding level-aligned supervision $\{C_t^l, S_t^l\}$ for each image I_t.

CityGML-based Semantic Masks To incorporate CityGML semantics into the learning pipeline, the original boundary representation (B-Rep) geometry is first parsed into the CityJSON (Ledoux et al., 2019) encoding, which provides a lightweight and programmatically accessible representation for both geometry and semantics. The resulting geometric entities are then converted into polygonal meshes with per-face semantic labels $c \in \mathcal{C}$ (e.g., *roof*, *wall*, *window*). Each polygon is assigned a hierarchy index $l \in \{1, 2, 3\}$ following the semantic mapping defined in the previous section, yielding an explicit semantic mesh with consistent per-level annotations.

For each calibrated view t, the semantic mesh is projected into the image plane by raycasting with the camera model $\pi_t(\mathbf{X}) = \Pi\big(K_t[R_t \mid \mathbf{t}_t]\mathbf{X}\big)$, where $\Pi(\cdot)$ denotes the perspective division from homogeneous coordinates to pixel coordinates. Concretely, for each pixel $v \in \Omega = \{1, \dots, H\} \times \{1, \dots, W\}$, a ray is cast from the camera center through v into the semantic mesh to identify the closest polygon that this ray hits. The pixel is then assigned the semantic class of this front-most polygon, producing a set of hierarchy-aligned semantic maps

$$C_t^l(v) \in \{-1, 0, \dots, |\mathcal{C}|\}, \qquad l \in \{1, 2, 3\},$$

where negative values indicate invalid pixels.

In addition, a binary coverage mask $M_t(v) \in \{0, 1\}$ is defined as

$$M_t(v) = \begin{cases} 1, & \text{if } \exists\, l \in \{1, 2, 3\} \text{ with } C_t^l(v) \geq 0, \\ 0, & \text{otherwise.} \end{cases}$$

The mask indicates valid CityGML coverage and is used to confine CityGML-derived semantic and feature supervision to these regions during training.

Text-guided CLIP Feature Fusion Each semantic region in the CityGML-derived maps is further enriched with language features by combining visual and textual embeddings. Let $V(\cdot)$ denote the CLIP image encoder and $T(\cdot)$ the text encoder with a class-specific prompt τ_c (e.g., “a window on a building

façade”). For each level $l \in \{1, 2, 3\}$ and class $c \in \mathcal{C}$, a binary region mask

$$R_t^{l,c}(v) = \begin{cases} 1, & \text{if } C_t^l(v) = c \text{ and } M_t(v) = 1, \\ 0, & \text{otherwise} \end{cases}$$

is defined to select pixels that belong to class c and are covered by CityGML geometry. The corresponding masked image is encoded by the CLIP image encoder,

$$\mathbf{f}_{t,l,c}^{\text{gml}} = V\big(I_t \odot R_t^{l,c}\big), \qquad \mathbf{f}_c^{\text{text}} = T(\tau_c),$$

where $\odot$ denotes element-wise multiplication with the broadcast mask. The two embeddings are fused using a text-guidance weight $\alpha \in [0, 1]$:

$$\mathbf{f}_{t,l,c}^{\text{gml+txt}} = (1-\alpha)\,\mathbf{f}_{t,l,c}^{\text{gml}} + \alpha\,\mathbf{f}_c^{\text{text}}.$$

Each fused feature vector $\mathbf{f}_{t,l,c}^{\text{gml+txt}} \in \mathbb{R}^{512}$ is broadcast back to its region, forming a dense feature field $\mathbf{L}_t^l(v)$ aligned with the image geometry:

$$\mathbf{L}_t^l(v) = \sum_{c \in \mathcal{C}} \mathbf{f}_{t,l,c}^{\text{gml+txt}}\, R_t^{l,c}(v).$$

This dense field $\mathbf{L}_t^l$ serves as the geometry-aligned semantic representation. For latent-space supervision, the associated region-level vectors $\mathbf{f}_{t,l,c}^{\text{gml+txt}}$ are subsequently passed to the autoencoder and compressed together with SAM-based features.

3.2 Visual Feature Extraction and Compression

In parallel to the CityGML-guided semantic branch, purely visual features are extracted from each image using SAM (Kirillov et al., 2023). Consistent with the hierarchical indexing introduced above, SAM provides segment masks $S_t^{l,c}(v) \in \{0, 1\}$ for class c at levels $l \in \{1, 2, 3\}$. Each segment is applied as a mask to the RGB image and encoded by the CLIP image encoder $V(\cdot)$:

$$\mathbf{f}_{t,l,c}^{\text{sam}} = V\big(I_t \odot S_t^{l,c}\big).$$

To ensure consistent feature representations across modalities while preserving scene coverage, a lightweight autoencoder is first trained on visual features extracted from SAM. The same autoencoder is then applied to compress both SAM-based features $\mathbf{f}_{t,l,c}^{\text{sam}}$ and CityGML-guided fused features $\mathbf{f}_{t,l,c}^{\text{gml+txt}}$ into a shared latent space, ensuring cross-source compatibility.

Given an input feature vector $\mathbf{f}_{t,l,c} \in \mathbb{R}^{512}$ from either source, the autoencoder learns an encoding–decoding mapping:

$$\mathbf{z}_{t,l,c} = \mathcal{E}(\mathbf{f}_{t,l,c}), \qquad \hat{\mathbf{f}}_{t,l,c} = \mathcal{D}(\mathbf{z}_{t,l,c}),$$

where $\mathcal{E}(\cdot)$ and $\mathcal{D}(\cdot)$ denote the encoder and decoder networks, and $\mathbf{z}_{t,l,c} \in \mathbb{R}^{128}$ is the compressed latent embedding. Since CLIP embeddings are trained and compared using cosine similarity, their semantic information is mainly encoded in the feature direction rather than in the absolute magnitude (Radford et al., 2021). The autoencoder is therefore trained with a reconstruction loss that combines Euclidean and cosine terms:

$$\mathcal{L}_{\text{rec}} = \big\|\mathbf{f}_{t,l,c} - \hat{\mathbf{f}}_{t,l,c}\big\|_2^2 + \lambda_{\cos}\Big(1 - \cos\big(\mathbf{f}_{t,l,c}, \hat{\mathbf{f}}_{t,l,c}\big)\Big).$$

The Euclidean term preserves the overall scale of the features, while the cosine term preserves their angular similarity, encouraging reconstructed embeddings to remain aligned with the original CLIP feature space. This unified compression reduces memory and computational overhead and aligns the embedding distributions of SAM- and CityGML-based features in a consistent latent domain, facilitating stable joint optimization in the subsequent training stage.

3.3 Dual-feature Learning in 3DGS

The compressed latent features from both the CityGML-guided and SAM-based branches are integrated into a unified training scheme. This dual-feature learning stage enhances the standard 3DGS optimization by incorporating geometry-grounded and vision-based semantic supervision, enabling structured semantic understanding beyond purely photometric cues.

Semantic Feature Supervision Semantic training is performed on top of a photometrically pre-trained 3DGS reconstruction. Each Gaussian primitive $\mathcal{G}_i$ keeps its spatial and appearance parameters and is augmented with a learnable latent feature vector $\mathbf{g}_i \in \mathbb{R}^{128}$. For a training view t, the renderer splats these latent features into the image plane and produces a dense latent field $\hat{\mathbf{Z}}_t^l(v)$ at the selected hierarchy level l. This field is interpreted as the semantic prediction at pixel v.

On the supervision side, latent codes obtained from the autoencoder are used for both sources. CityGML guided fused features $\mathbf{f}_{t,l,c}^{\text{gml+txt}}$ and SAM based features $\mathbf{f}_{t,l,c}^{\text{sam}}$ are mapped to latent vectors by the encoder $\mathcal{E}$,

$$\mathbf{z}_{t,l,c}^{\text{gml}} = \mathcal{E}\big(\mathbf{f}_{t,l,c}^{\text{gml+txt}}\big), \qquad \mathbf{z}_{t,l,c}^{\text{sam}} = \mathcal{E}\big(\mathbf{f}_{t,l,c}^{\text{sam}}\big).$$

Using the region masks introduced above, these latent vectors are broadcast into dense supervision fields

$$\mathbf{Z}_t^{l,\text{CityGML}}(v) = \sum_{c \in \mathcal{C}} \mathbf{z}_{t,l,c}^{\text{gml}}\, R_t^{l,c}(v),$$
$$\mathbf{Z}_t^{l,\text{SAM}}(v) = \sum_{c \in \mathcal{C}} \mathbf{z}_{t,l,c}^{\text{sam}}\, S_t^{l,c}(v).$$

The coverage mask $M_t(v)$ from the CityGML-based semantic masks determines where CityGML supervision is available. In regions without CityGML coverage, that is $M_t(v) = 0$, the prediction is aligned only with the SAM supervision field. In CityGML-covered regions, that is $M_t(v) = 1$, the prediction is aligned with both SAM and CityGML supervision. The feature loss is decomposed into three terms

$$\mathcal{L}_{\text{SAM}} = \sum_t \sum_v \big(1 - M_t(v)\big)\big\|\hat{\mathbf{Z}}_t^l(v) - \mathbf{Z}_t^{l,\text{SAM}}(v)\big\|_1,$$

$$\mathcal{L}_{\text{SAM-GML}} = \sum_t \sum_v M_t(v)\big\|\hat{\mathbf{Z}}_t^l(v) - \mathbf{Z}_t^{l,\text{SAM}}(v)\big\|_1,$$

$$\mathcal{L}_{\text{CityGML}} = \sum_t \sum_v M_t(v)\big\|\hat{\mathbf{Z}}_t^l(v) - \mathbf{Z}_t^{l,\text{CityGML}}(v)\big\|_1.$$

The resulting dual feature objective is

$$\mathcal{L}_{\text{feat}} = \mathcal{L}_{\text{SAM}} + \lambda_{\text{sam}}\mathcal{L}_{\text{SAM-GML}} + \lambda_{\text{citygml}}\mathcal{L}_{\text{CityGML}}.$$

The hyperparameters λ_{sam} and λ_{citygml} control the relative strength of SAM and CityGML supervision in CityGML covered regions, while $\mathcal{L}_{\text{SAM}}$ provides SAM based guidance in

regions outside the CityGML coverage. In this semantic training phase, the photometrically pre-trained reconstruction is kept fixed and only the latent features are updated using $\mathcal{L}_{\text{feat}}$. The optimization therefore focuses on aligning the Gaussian latent representation with SAM-based open vocabulary cues and with geometry-grounded CityGML semantics in regions where CityGML supervision is available.

Open-Vocabulary Querying After training, each Gaussian primitive carries a geometry aware and language aligned latent feature vector $\mathbf{g}_i$. Given a user-defined text query τ_q, the query is encoded with the same CLIP text encoder $T(\cdot)$ and mapped into the latent space using the projection employed during training. This yields a query embedding $\mathbf{q}$ that is compatible with the Gaussian latent features. A cosine similarity score between the query and each Gaussian is then computed as

$$s_i = \frac{\langle \mathbf{q},\, \mathbf{g}_i \rangle}{\|\mathbf{q}\|_2 \, \|\mathbf{g}_i\|_2},$$

where $\| \cdot \|_2$ denotes the Euclidean norm. The similarity scores $\{s_i\}$ enable open vocabulary 3D querying and visualization by applying threshold-based selection or ranking over the Gaussian set. Architectural components such as *walls* and *windows* can be highlighted in the reconstructed scene with text prompts.

Through this dual-feature learning design, CityLangSplat combines the structured semantics of CityGML with the generalization capabilities of vision-language models. The resulting representation supports geometry-aligned, interpretable, and open-vocabulary semantic understanding within 3DGS framework.

4. Experiments

We evaluate the open-vocabulary semantic understanding of CityLangSplat by comparing model predictions with ground truth in both 2D image space and 3D Gaussian space. Across all experiments, we benchmark CityLangSplat against the baseline LangSplat (Qin et al., 2024) using a shared CLIP-based text prompt set for all classes.

4.1 Dataset and Evaluation Metrics

We conducted experiments on the publicly available TUM2TWIN (Wysocki et al., 2025a) and ZAHA (Wysocki et al., 2025b) datasets, both capturing the central campus of the Technical University of Munich and its surrounding urban areas. The TUM2TWIN dataset provides multi-modal data, including high-resolution UAV imagery and CityGML building models. The UAV imagery collection (Anders et al., 2025) comprises 1,179 photographs that cover over 70 campus buildings, providing rich visual and geometric diversity. For our experiments, we selected nine representative subsets, each corresponding to distinct building clusters containing approximately 20–30 images, covering a range of architectural styles and urban densities. These subsets were used for CityLangSplat training and visual reconstruction experiments.

For quantitative evaluation, we use the ZAHA dataset (Wysocki et al., 2025b), which contains more than 600 million semantically annotated MLS points organized according to the Level of Façade Generalization (LoFG) scheme and provides fine-grained façade labels (e.g., *wall, window, door, roof*) harmonized with CityGML. To evaluate image-space semantic predictions, ground-truth masks are obtained by projecting ZAHA labels into the image domain. Metrics are computed only on pixels with valid annotations. We report mean IoU (mIoU) as the primary metric and weighted IoU (wIoU) to account for class imbalance; pixel accuracy and the evaluated coverage are reported as auxiliary indicators, and per-class IoU is used for category-wise analysis. To assess semantic consistency in 3D, each Gaussian primitive is assigned a class by querying its decoded language features against the CLIP-based text prototypes. The resulting semantic labels on Gaussian centers are aligned with the ZAHA point cloud through a nearest-neighbor search within a fixed distance threshold. Scores are computed based on matched ground-truth points, and we report 3D mIoU and wIoU, along with 3D coverage (the percentage of matched points) and the mean match distance.

4.2 Implementation Details

Our implementation builds upon the LangSplat (Qin et al., 2024) codebase. Scene geometry is first reconstructed with 3DGS (Kerbl et al., 2023) for 30,000 iterations to obtain a photometric model. Training then continues for another 30,000 iterations with the language branch enabled to learn the 3D language field. All experiments run on a single NVIDIA RTX 4090 GPU. Structure-from-motion inputs (intrinsics, extrinsics, and sparse points) are generated with Pix4Dmatic (Pix4D SA, 2024) using default parameters and are directly used to initialize the pipeline. For data processing, CityGML building models are parsed and raycast with Open3D (Zhou et al., 2018) to produce per-view semantic masks and coverage maps aligned to each image. As the baseline, we use the original LangSplat implementation with its default hyperparameter configuration. For CityLangSplat, we adopt the dual-feature loss from Section 3.3 with fixed weights ($\lambda_{\text{sam}} = 1.0$, $\lambda_{\text{citygml}} = 0.5$).

4.3 Results

We assess CityLangSplat through language-guided semantic querying in both 2D image space and 3D Gaussian space.

Quantitative Evaluation For quantitative evaluation, we compare open-vocabulary semantic predictions with ZAHA ground truth in both 2D and 3D. For each façade class, a small, fixed set of short noun phrases is used, derived directly from its label and a few natural variants in a consistent manner (e.g., “door”, “building entrance”, “door opening” for the *door* class). Each phrase is encoded with the CLIP text encoder, and the resulting embeddings are averaged to obtain a single text prototype per class. These class-specific phrase sets are treated as evaluation hyperparameters and are kept the same for all façade classes and all compared methods to ensure fairness.

For 2D evaluation, per-class CLIP relevance maps are computed independently without softmax, normalized to $[0, 1]$, and the per-pixel label is assigned by the argmax across classes. Low-confidence pixels (below a fixed threshold 0.3) are ignored, and scoring is restricted to valid ZAHA projections. Table 1 reports 2D results across the nine subsets, including mIoU, wIoU, pixel accuracy, and evaluated coverage. CityLangSplat consistently surpasses LangSplat in mIoU and wIoU at comparable coverage. Figure 3 presents qualitative 2D results on representative scenes, comparing ground-truth masks, LangSplat, and our CityLangSplat. In 3D evaluation, each Gaussian primitive is assigned a class by taking the argmax over the shared text prototypes, and the resulting semantic labels are compared with ZAHA by nearest-neighbor matching within a fixed $0.2\,\text{m}$ radius. Metrics are computed

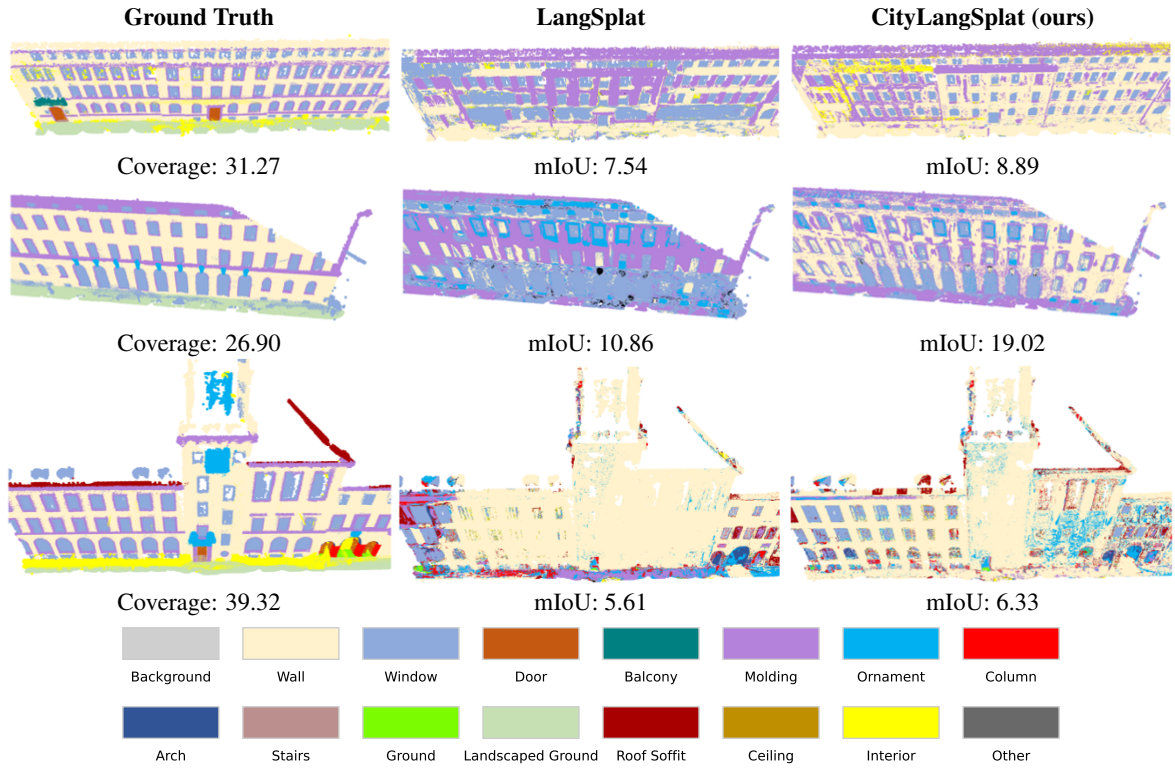

Figure 3. Qualitative 2D open-vocabulary semantics on representative scenes.

Table 1. 2D quantitative results evaluated against ZAHA ground truth projected into the image domain across nine subsets. Cov. denotes the percentage of pixels with valid ZAHA annotations. Higher is better; best results per row are shown in **green**.

ID	Cov.	LangSplat			CityLangSplat (ours)		
		mIoU	wIoU	Acc	mIoU	wIoU	Acc
1	26.90	10.86	13.38	27.04	**19.02**	**31.44**	**45.73**
2	39.32	5.61	22.30	**40.88**	**6.33**	**22.42**	40.53
3	33.43	**8.75**	30.40	51.42	8.74	**30.51**	**51.87**
4	77.53	15.67	37.46	56.97	**16.72**	**39.01**	**58.83**
5	34.05	8.60	17.61	27.39	**10.30**	**19.81**	**31.06**
6	20.33	10.01	22.22	35.43	**14.50**	**25.44**	**38.93**
7	31.27	7.54	16.29	29.58	**8.89**	**21.24**	**35.48**
8	32.20	**8.58**	**33.25**	**52.89**	7.36	31.75	52.67
9	7.64	4.65	6.56	19.68	**6.82**	**9.40**	**22.93**
Avg.	33.63	8.92	22.16	37.92	**10.96**	**25.67**	**42.00**

Table 2. 3D quantitative results evaluated against ZAHA ground truth via nearest-neighbor matching across nine subsets. Cov. denotes 3D coverage (percentage of matched points), and Dist. denotes the mean match distance (in meters). Best per row is highlighted in **green**.

ID	Cov.	Dist.	LangSplat		CityLangSplat	
			mIoU	wIoU	mIoU	wIoU
1	80.13	0.10	3.77	3.20	**4.56**	**4.08**
2	41.96	0.11	2.91	5.30	**3.28**	**6.19**
3	55.65	0.11	7.81	25.14	**8.27**	**24.85**
4	22.50	0.12	7.44	**11.99**	**7.95**	11.86
5	18.91	0.12	3.09	5.76	**3.20**	**5.85**
6	42.17	0.13	2.88	2.78	**2.90**	**2.82**
7	39.24	0.12	2.66	**4.13**	**2.74**	3.93
8	24.60	0.13	2.46	2.58	**3.64**	**4.37**
9	77.39	0.09	3.36	8.04	**5.33**	**17.56**
Avg.	44.73	0.11	4.04	7.66	**4.65**	**9.06**

on the matched ground-truth set. Table 2 reports 3D results, including mIoU, wIoU, 3D coverage (percentage of matched points), and mean match distance, and shows that CityLangSplat consistently outperforms LangSplat in 3D space.

Class-wise Analysis We analyze façade semantics using per-class IoU, precision, and recall on valid ground-truth regions. Predictions are compared with ZAHA labels within the evaluated coverage, and metrics are computed per image and then averaged per class across all subsets to obtain dataset-level scores. Images or subsets without a given class are excluded from that class's statistics. Table 3 summarizes results for representative classes with sufficient frequency. Figure 4 presents per-pixel cosine-similarity maps to the text query “*window*”. In addition to RGB images and ground-truth masks as references, the first column shows pre-trained similarity maps from SAM-derived and CityGML-derived features, and the second column shows the trained language-field similarity maps for LangSplat and CityLangSplat. CityLangSplat produces sharper, more localized window responses with fewer spurious activations.

4.4 Discussion

This section discusses the experimental results of CityLangSplat, focusing on its overall 2D and 3D evaluation outcomes, a per-class analysis of façade semantics, and its limitations.

Overall Performance in 2D and 3D CityLangSplat improves open-vocabulary semantics in both 2D and 3D. On 2D evaluation, as shown in Table 1, the average mIoU increases from 8.92 to 10.96 (a relative gain of 22.9%), and wIoU from 22.16 to 25.67 (a relative gain of 15.8%). Although these absolute scores remain modest, a pronounced gap exists between mIoU and wIoU. This difference is mainly attributable to severe

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-411-2026

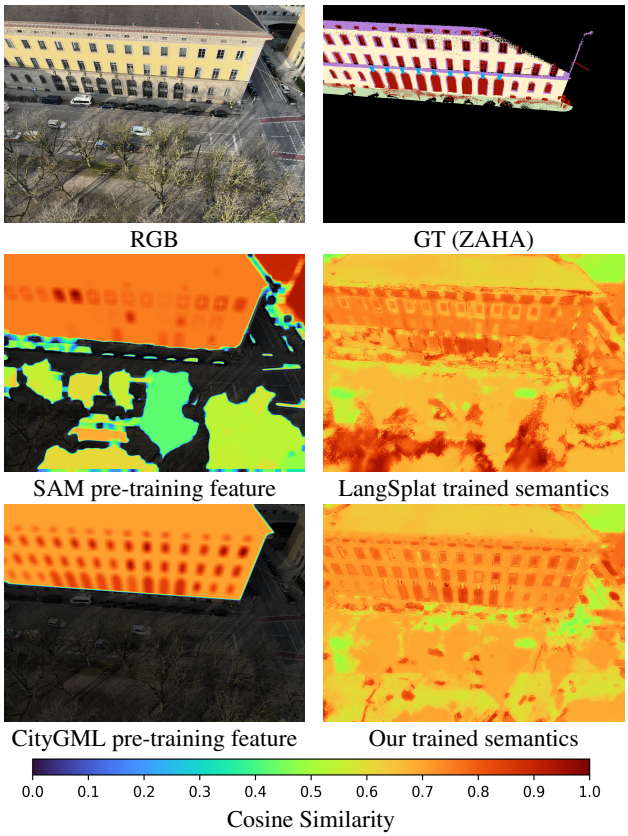

Figure 4. Per-pixel cosine-similarity maps for *window* (0–1, blue→red). Rows: RGB/GT; LangSplat pre-training vs. trained language-field similarity; CityLangSplat pre-training vs. trained language-field similarity. CityLangSplat yields sharper and cleaner *window* localization.

class imbalance in large-scale outdoor scenes, where extensive classes (e.g., *wall*) dominate the frequency-weighted average, while rare, small, and spatially sparse categories substantially lower the macro-averaged mIoU. In addition, classes such as *window* exhibit strong appearance variation due to illumination and reflections, which further complicates language-feature learning. Across most subsets, CityLangSplat predicts these challenging classes more accurately than LangSplat. Subset 8 in Table 1 is a notable case where LangSplat achieves slightly higher performance than CityLangSplat. It contains façades with highly irregular and occluded windows, including blinds and non-standard fillings that break the usual window pattern. Our geometry-guided fusion produces consistent, high-confidence window prototypes aligned with CityGML masks, acting as a smoothing prior on learned features. This improves robustness on most scenes but weakens the response to such atypical window appearances. In contrast, the purely image-driven features in LangSplat assign high similarity to these local patterns, which in this subset partly coincide with the projected ZAHA window labels and thus yield slightly higher scores.

In 3D evaluation results (Table 2), CityLangSplat also outperforms LangSplat. Overall scores are lower than their 2D counterparts, which is expected given that nearest-neighbor matching within a fixed radius is sensitive to the sparsity of the 3DGS reconstruction. Moreover, 3D evaluation assesses semantic geometric consistency across the entire scene, whereas 2D evaluation checks image-space agreement only on visible, projected pixels. In summary, accurately understanding open-vocabulary semantics in urban environments remains challenging. Despite this, fusing CityGML priors with vision language features yields consistent improvements in both 2D and 3D.

Table 3. Class-wise 2D façade semantics evaluated against ZAHA ground truth projected into the image domain, reporting per-class IoU, precision, and recall for representative façade classes, comparing LangSplat and CityLangSplat.

Class	LangSplat			CityLangSplat		
	IoU	Precision	Recall	IoU	Precision	Recall
Wall	41.62	53.39	68.78	**50.86**	**59.17**	**80.66**
Window	17.67	33.27	33.78	**21.95**	**44.75**	**33.81**
Door	0.00	0.00	0.00	**0.01**	**0.53**	**0.01**
Molding	5.27	11.01	11.33	**7.55**	**14.09**	**19.64**
Deco	0.62	0.87	1.22	**1.51**	**2.04**	**5.73**
Roof	3.48	8.23	5.71	**10.89**	**17.76**	**21.97**
Interior	**0.78**	**11.65**	0.86	0.62	11.15	**0.91**

Per-class Analysis A class-wise breakdown (Table 3) shows where the geometry-grounded supervision brings the largest semantic gains. CityLangSplat improves all major exterior façade classes. For *Wall*, the IoU increases from 41.62% to 50.86% and Recall from 68.78% to 80.66%, suggesting that the CityGML prior helps the model cover continuous façade surfaces more completely and consistently. The improvement for *Window* is also clear, with a relative IoU gain of 24%. As shown in Figure 4, windows often suffer from visual ambiguity, so purely vision-based features in LangSplat can mix window regions with surrounding wall textures or interior content. By enforcing geometry-aligned window regions from CityGML, CityLangSplat constrains the language features to the correct architectural component, leading to sharper localization and higher precision. The pre-training feature maps in Figure 4 further confirm that the CityGML branch already provides a cleaner signal than the purely visual SAM features for this challenging class. For finer-grained and less frequent classes such as *Molding*, *Deco*, and *Roof*, the relative gains are even larger. For instance, the IoU for *Roof* rises from 3.48% to 10.89%. These elements are often small, partly occluded, or visually similar to neighboring structures, where the explicit hierarchical labels and precise geometry in CityGML provide a strong prior for separating adjacent façade parts.

Limitations and Future Work CityLangSplat shows consistent improvements on most façade classes, while some challenging cases remain. A representative example is the *Interior* class, where performance slightly decreases, likely because interior regions are only indirectly visible through reflective windows and are therefore less consistently represented in the geometry-aligned semantic supervision. More generally, the current framework benefits most from accurate 3DGS reconstruction and close alignment with the CityGML model. In addition, the present evaluation focuses on building façade semantics and does not yet cover more diverse urban categories. Future work will extend the framework to broader urban semantics, investigate more robust geometry-appearance integration under imperfect alignment, and explore both stronger integration with recent language-GS variants and coarser supervision from widely available CityGML models such as LoD2.

5. Conclusions

We presented CityLangSplat, a geometry-grounded 3DGS framework that integrates structured CityGML supervision

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-411-2026

with vision-language features to enable open-vocabulary understanding in urban building scenes. Experiments on the TUM2TWIN and ZAHA datasets show consistent gains over LangSplat across 2D and 3D evaluations. CityLangSplat particularly improves the segmentation of façade components such as *wall* and *window*, with per-class mIoU gains of 9.24 and 4.28 points, while producing more geometry-consistent predictions for challenging classes. These results indicate that structured city-model priors can strengthen open-vocabulary semantic understanding in urban 3DGS, especially for building exteriors with complex appearance. Future work will extend the framework to broader urban object classes and explore more adaptive schemes for balancing CityGML and visual supervision under varying geometric reliability.

References

Anders, K., Wang, J., Wysocki, O., Huang, X., Liu, S., 2025. Uav laser scanning and photogrammetry of tum downtown campus. `https://doi.org/10.5281/zenodo.14899378`.

Banno, T., Takenawa, M., Wöhler, L., Ikehata, S., Aizawa, K., 2025. 360CityGML: Realistic and Interactive Urban Visualization System Integrating CityGML Model and 360 Videos. *IEEE Transactions on Visualization and Computer Graphics*.

Gröger, G., Kolbe, T. H., Nagel, C., Häfele, K.-H., 2012. OGC City Geography Markup Language CityGML Encod. Standard.

Jäger, M., Hillemann, M., Jutzi, B., 2025. FeatureGS: Eigenvalue-feature optimization in 3D Gaussian Splatting for geometrically accurate and artifact-reduced reconstruction. *ISPRS Open Journal of Photogrammetry and RS*, 17, 100100.

Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G., 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4), 139–1.

Kerr, J., Kim, C. M., Goldberg, K., Kanazawa, A., Tancik, M., 2023. Lerf: Language embedded radiance fields. *Proceedings of the IEEE/CVF international conference on computer vision*, 19729–19739.

Kirillov, A., Mintun, E., Ravi, N., 2023. Segment anything. *Proceedings of the IEEE/CVF International Conf. on Computer Vision*, 4015–4026.

Ledoux, H., Arroyo Ohori, K., Kumar, K., Dukai, B., Labetski, A., Vitalis, S., 2019. CityJSON: A compact and easy-to-use encoding of the CityGML data model. *Open Geospatial Data, Software and Standards*, 4(1), 1–12.

Lee, J. C., Rho, D., Sun, X., Ko, J. H., Park, E., 2024. Compact 3d gaussian representation for radiance field. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21719–21728.

Li, W., Zhao, Y., Qin, M., Liu, Y., Cai, Y., Gan, C., Pfister, H., 2025. Langsplatv2: High-dimensional 3d language gaussian splatting with 450+ fps. *arXiv preprint arXiv:2507.07136*.

Mildenhall, B., P. Srinivasan, P., Tancik, M., 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1), 99–106.

Niemeyer, J., Rottensteiner, F., Soergel, U., 2014. Contextual classification of lidar data and building object detection in urban areas. *ISPRS journal of photogrammetry and remote sensing*, 87, 152–165.

Petrovska, I., Jutzi, B., 2024. Vision through Obstacles: 3D Geometric Reconstruction and Evaluation of Neural Radiance Fields (NeRFs). *Remote Sensing*, 16(7), 1188.

Pix4D SA, 2024. Pix4Dmatic Software, Version 1.71.0. `https://www.pix4d.com/product/pix4dmatic` (24 April 2025).

Qin, M. et al., 2024. Langsplat: 3d language gaussian splatting. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 20051–20060.

Radford, A., Kim, J., Hallacy, C., 2021. Learning transferable visual models from natural language supervision. *International conference on machine learning*, PmLR, 8748–8763.

Wu, Y. et al., 2024. Opengaussian: Towards point-level 3d gaussian-based open vocabulary understanding. *Advances in Neural Information Processing Systems*, 37, 19114–19138.

Wysocki, O. et al., 2023. Scan2lod3: Reconstructing semantic 3d building models at lod3 using ray casting and bayesian networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6548–6558.

Wysocki, O. et al., 2025a. TUM2TWIN: Introducing the Large-Scale Multimodal Urban Digital Twin Benchmark Dataset. *arXiv preprint arXiv:2505.07396*.

Wysocki, O. et al., 2025b. ZAHA: Introducing the level of facade generalization and the large-scale point cloud facade semantic segmentation benchmark dataset. *2025 IEEE/CVF Winter Conf. on Applications of Computer Vision*, IEEE, 7648–7658.

Wysocki, O., Schwab, B., Beil, C., Holst, C., Kolbe, T. H., 2024. Reviewing Open Data Semantic 3D City Models to Develop Novel 3D Reconstruction Methods. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48, 493–500.

Ye, M., Danelljan, M., Yu, F., Ke, L., 2024. Gaussian grouping: Segment and edit anything in 3d scenes. *European conference on computer vision*, Springer, 162–179.

Zhang, H., Li, F., Ahuja, N., 2024a. Open-nerf: Towards open vocabulary nerf decomposition. *Proceedings of the IEEE/CVF Winter Conf. on Applications of Computer Vision*, 3456–3465.

Zhang, Q., Wysocki, O., Jutzi, B., 2025. GS4Buildings: Prior-Guided Gaussian Splatting for 3D Building Reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-4/W6-2025, 249–256.

Zhang, Q., Wysocki, O., Urban, S., Jutzi, B., 2024b. CDGS: Confidence-Aware Depth Regularization for 3D Gaussian Splatting. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVIII-2/W7-2024, 189–196.

Zhou, Q.-Y., Park, J., Koltun, V., 2018. Open3D: A modern library for 3D data processing. *arXiv preprint arXiv:1801.09847*.

Zhou, S., Chang, H., Jiang, S., Fan, Z., Zhu, Z., Xu, D., Chari, P., You, S., Wang, Z., Kadambi, A., 2024. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21676–21685.