ZeroSplat: Generalized Referring Segmentation in 3D Gaussian Splatting

Jiayu Ding[1,4⋆], Meilu Song[2*], Xiaoyi Zhang[3]
Hongbo Jin[1,4], Yichen Jin[1], and Xiangtian Si[3†]

[1]Peking University [2]North China Electric Power University
[3]China University of Geosciences [4]InkMind.AI

Abstract. Recent advancements in 3D Gaussian Splatting (3DGS) have enabled language-guided scene understanding. However, existing Referring 3D Gaussian Splatting (R3DGS) methods are fundamentally restricted to single-target queries. To reflect the ambiguity of real-world instructions, we introduce the Generalized Referring 3D Gaussian Splatting Segmentation (GR3DGS) task, which requires dynamically segmenting an arbitrary number of targets (0, 1, or N). To facilitate comprehensive evaluation of this new task, we construct two new benchmarks: GR-LERF and GR-ScanNet. Crucially, existing R3DGS paradigms exhibit fundamental technical bottlenecks that severely limit their performance on the GR3DGS task: they lack intrinsic 3D point-level understanding by operating merely on 2D rendered pixels, and they incur prohibitive computational overhead by requiring per-scene optimization to embed heavy semantic features. To dismantle these bottlenecks, we propose ZeroSplat, a novel training-free and zero-feature framework. ZeroSplat lifts 2D Vision-Language Model (VLM) priors into 3D space through robust multi-view geometric constraints. This strategy enables intrinsic point-level understanding without incurring any additional feature storage. Extensive experiments demonstrate that ZeroSplat significantly outperforms state-of-the-art methods across generalized and single-target scenarios while maintaining exceptional efficiency. *Project Page: https://inkmind-ai.github.io/ZeroSplat*

Keywords: 3D Gaussian Splatting · Referring Segmentation · 3D Scene Understanding

1 Introduction

3D Gaussian Splatting (3DGS) [15] represents scenes using explicit 3D Gaussians to enable high-quality real-time rendering. Beyond visual synthesis, recent research has evolved to equip 3DGS with semantic understanding capabilities. Initially, Open-Vocabulary 3DGS Understanding methods [9,14,20,21,25,33,38,42] distilled 2D foundation model features into 3D space, allowing users to query scenes using text. However, these approaches typically rely on fixed category

⋆ Equal contribution. † Corresponding author.

Fig. 1: Traditional R3DGS is limited to single-target cases(1). In contrast, GR3DGS can handle scenarios with any number of targets, including no target (2), single target, and multiple targets (3-5).

names or simple noun phrases. This limitation impedes free-form language understanding, core to Embodied AI [5, 17–19, 34] and multimodal LLM agent [1, 12, 13, 39, 40], with user queries carrying fine-grained attributes and intricate spatial relations. Hence, Referring 3D Gaussian Splatting (R3DGS) [7] fills this gap by segmenting targets from elaborate text prompts.

However, existing R3DGS paradigms [7] suffer from a critical limitation: they are strictly confined to a "single-target" setting. As illustrated in Fig. 1, these methods presuppose that each instruction corresponds to exactly one target in the scene. This assumption severely hinders flexibility in real-world scenarios, where user instructions are inherently uncertain, often involving multi-target requests (e.g., find all red chairs) or no-target queries (i.e., the object is absent). To address this, we introduce the **Generalized Referring 3D Gaussian Splatting Segmentation (GR3DGS)** task. This task mandates parsing instructions to segment an arbitrary number of targets (0, 1, or N), imposing higher demands on semantic discrimination and robustness against ambiguity. To facilitate evaluation, we construct **GR-LERF** for pixel-level assessment and **GR-ScanNet** for intrinsic point-level assessment via 3D annotations.

Beyond the task formulation, the existing technical paradigm encounters significant bottlenecks when applied to GR3DGS, suffering from two critical limitations: **(i) Lack of 3D Point-Level Understanding:** The current R3DGS method utilizes 2D rendered pixels as the fundamental unit for semantic processing rather than operating on discrete 3D Gaussian points. Fundamentally, this approach remains confined to 2D image-level understanding, failing to leverage the explicit 3D geometric structure of the scene to resolve complex spatial ambiguities. **(ii) Prohibitive Computational and Memory Overheads:** Furthermore, the existing method necessitates prolonged scene-specific optimization and requires embedding high-capacity semantic features into millions of Gaussian points. This massive storage footprint and peak memory consumption render it highly impractical for real-time applications and scalable deployment on resource-constrained devices.

To bridge these gaps, we propose **ZeroSplat**, a novel training-free and zero-feature framework tailored for GR3DGS. Diverging from the prevailing paradigm of coupling additional semantic parameters, ZeroSplat is built on the core insight that robust 3D semantic understanding does not mandate altering the intrinsic scene representation. Instead, it can be achieved by lifting 2D foundation model priors into 3D space through geometric constraints. By projecting semantic cues from 2D Vision-Language Models (VLMs) onto the 3D structure, our method directly filters and localizes targets within the original set of 3D Gaussians.

In summary, our contributions are as follows:

- We introduce a new task termed Generalized Referring 3D Gaussian Splatting Segmentation (GR3DGS).
- To support future research in GR3DGS, we construct GR-LERF and GR-ScanNet for evaluation at both pixel and point levels.
- To address GR3DGS challenges, we propose ZeroSplat, a zero-feature and training-free framework achieving intrinsic point-level understanding.
- Experiments show that our method outperforms existing approaches in both generalized and single-target scenarios.

2 Related Works

Preliminary: 3D Gaussian Splatting 3D Gaussian Splatting (3DGS) [15] represents 3D scenes using a set of explicit 3D Gaussians $\mathcal{G} = \{g_i\}_{i=1}^{N}$. Each Gaussian g_i is characterized by its mean position, covariance matrix (controlling scale and orientation), color, and opacity. To render a 2D image, these Gaussians are projected onto the image plane and blended in a depth-sorted order. The final color $C(p)$ for a pixel p is computed through alpha compositing [27]:

$$C(p) = \sum_{i=1}^{|\mathcal{G}_p|} c_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \tag{1}$$

where c_i and α_i denote the color and effective opacity of the i-th Gaussian intersecting pixel p. The effective opacity α_i is the product of the learned opacity and the spatial influence of the projected Gaussian. The term $T_i = \prod_{j=1}^{i-1}(1-\alpha_j)$ represents the accumulated transmittance, accounting for light attenuation from all preceding Gaussians along the viewing ray.

Language-grounded 3D Gaussian Splatting Following the success of 3D Gaussian Splatting (3DGS), recent studies have integrated open-vocabulary and natural language understanding into the 3DGS framework. Existing approaches primarily follow two paradigms: pixel-based and point-based methods. Pixel-based methods adopt a "render-then-match" strategy, where the scene is first rendered into dense 2D feature maps for semantic reasoning in the image space. Several works [7, 29, 30, 32, 41, 43] investigate open-vocabulary 3DGS understanding by distilling semantic features from 2D foundation models into view-consistent 3D representations to enable efficient semantic rendering. More recently, some

studies have explored referring 3DGS understanding. For example, ReferSplat [7] aligns 3D Gaussians with text queries using a position-aware cross-modal module to improve spatial reasoning. Although it builds an explicit 3D referring field, its localization still relies on identifying target pixels within rendered 2D images. Consequently, these methods treat 3D representations primarily as a proxy for rendering. They lack structural understanding of the scene and cannot directly identify or manipulate individual Gaussian primitives, limiting their application in interactive tasks [10,11,23]. To address these limitations, point-based methods employ a "match-then-render" paradigm, treating 3D Gaussian primitives as the basic units of understanding. For open-vocabulary tasks, methods such as OpenGaussian [38] and InstanceGaussian [20] use 2D masks from models like SAM to learn 3D-consistent instance features. Other works, including Dr.Splat [14], LUDVIG [25] and ExtrinSplat [4], lift 2D features directly onto 3D Gaussian points via feature aggregation or graph diffusion. While point-based methods have made progress in open-vocabulary scenarios, extending their advantages to referring 3DGS understanding remains largely unexplored.

Referring Segmentation Referring segmentation aims to localize target regions described by natural language queries. Initially developed in the 2D domain, Referring Image Segmentation [22,35] (RIS) has achieved remarkable success by aligning linguistic features with pixels. However, RIS remains confined to the image plane, lacking the spatial reasoning capabilities essential for real-world interaction. To address this limitation, 3D Referring Segmentation [6,8,24,36,37] (3D RES) extends language grounding to 3D data. Recently, to better reflect real-world complexities, the task has evolved into Generalized 3D Referring Segmentation [37] (3D-GRES). Breaking the constraint of single-object localization, 3D-GRES allows expressions to refer to an arbitrary number of targets, aiming to predict a binary mask covering all relevant 3D points. Despite these advancements, existing methods and datasets rely primarily on 2D images or 3D point clouds. Consequently, they cannot be applied to GR3DGS.

3 Method

3.1 Task Definition and Method Overview

Formally, given a 3D scene reconstructed from a set of multi-view images $\mathcal{I} = \{I_i\}_{i=1}^{N}$ and represented by a 3DGS field $\mathcal{G}$, along with a free-form natural language expression $\mathcal{T}$, the GR3DGS task aims to assign a binary semantic label to each Gaussian $g \in \mathcal{G}$. Unlike standard R3DGS, which strictly assumes that $\mathcal{T}$ corresponds to exactly one target, GR3DGS formulates a more realistic open-world setting. The expression $\mathcal{T}$ can refer to an arbitrary number of instances. Consequently, the model must dynamically segment multiple targets, a single target, or output an empty mask if the queried object is absent from the scene. This setting requires the framework to perform precise 3D spatial reasoning while maintaining robust semantic discrimination to reject false positives.

To tackle GR3DGS, we propose a three-stage end-to-end framework that hierarchically connects unstructured text to 3D geometric anchors (Figure 2).

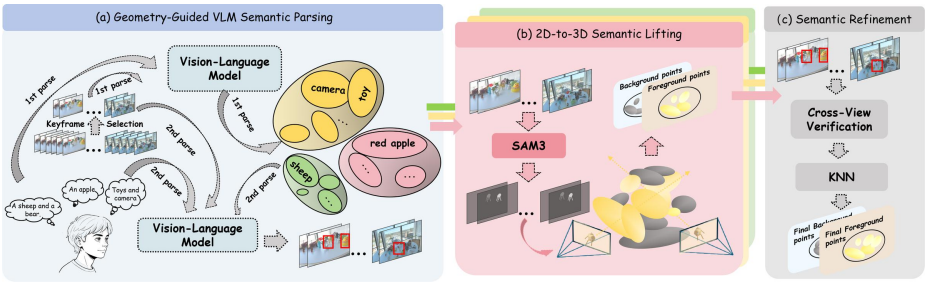

Fig. 2: Overview of our method. (a) Keyframes are first extracted from the input scene. A VLM then performs a two-stage parsing: the first extracts semantic labels from text via multi-view keyframe analysis, while the second localizes targets with 2D bounding boxes using these labels, referring text, and keyframes. (b) Guided by these labels, partial multi-view 2D masks are extracted across all scene views and used to back-project objects into 3D point groups. (c) Finally, erroneous Gaussians are filtered using the VLM-predicted 2D boxes, and internal 3D Gaussian structures are filled via a KD-Tree-based KNN spatial label diffusion algorithm.

First, we select geometric keyframes via curvature evaluation and use multi-stage interactions with a Vision-Language Model (VLM) to extract semantic labels and 2D localizations from the text (Sec. 3.2). Second, we use SAM3 to generate 2D semantic masks and lift them into the 3D Gaussian field (Sec. 3.3). Finally, we apply multi-view verification and a spatial diffusion algorithm to refine the 3D semantic structure (Sec. 3.4).

3.2 Geometry-Guided VLM Semantic Parsing

The first step of our pipeline parses the free-form textual query into semantic and spatial priors. Directly segmenting scenes using complex instructions often introduces ambiguity. To address this, we extract a concise semantic label from the query to facilitate downstream 2D mask generation. However, this label denotes a broad category that includes both the intended target and irrelevant instances. To resolve this ambiguity, we employ a VLM to generate 2D bounding box anchors based on the original query. These anchors provide spatial constraints to accurately isolate the referred target from the broader semantic category.

Geometry-Guided Keyframe Selection. To apply the VLM efficiently, we avoid processing all N frames of a video sequence, which causes high computational cost, latency, and view redundancy. Instead, we greedily select a compact subset $\mathcal{I}_{key}$ of K keyframes based on scene geometric curvature. In 3D scenes, complex geometric regions such as edges and corners typically contain richer semantic information than flat areas. Given the explicit point cloud $\mathcal{P}$ formed by the Gaussian centers, we first perform sub-voxel downsampling on $\mathcal{P}$ to obtain a representative subset $\mathcal{P}_{sub}$ at a resolution of γs, where s denotes the base voxel size and γ is the downsampling ratio. We then compute the eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$ of the covariance matrix formed by neighbors within a search

radius $r = \eta s$, where η is a fixed coefficient. We define the local surface variation rate as $\sigma_{\mathbf{p}} = \lambda_3/(\lambda_1+\lambda_2+\lambda_3+\epsilon)$, where ϵ is a small constant to ensure numerical stability. We then map $\sigma_{\mathbf{p}}$ to a saliency weight $w_{\mathbf{p}}$ using min-max normalization, scaling it to a target interval $[w_{min}, w_{max}]$:

$$w_{\mathbf{p}} = w_{min} + (w_{max} - w_{min})\frac{\sigma_{\mathbf{p}} - \sigma_{min}}{\sigma_{max} - \sigma_{min}} \tag{2}$$

To ensure efficient scene coverage, we discretize the scene into a voxel grid. We define the weight $W(v)$ of each voxel v as the maximum saliency weight $w_{\mathbf{p}}$ among its internal points. Let $\mathcal{U}$ be the set of voxels covered by the selected keyframes, initialized as $\mathcal{U} = \emptyset$. For each candidate frame I_i, we back-project its pixels into 3D space using the depth map and camera parameters to obtain the set of valid observed voxels $\mathcal{V}(I_i)$. We evaluate the value of adding a new frame using marginal gain:

$$\text{Gain}(I_i|\mathcal{U}) = \sum_{v \in \mathcal{V}(I_i) \setminus \mathcal{U}} W(v) \tag{3}$$

In each iteration, we add the frame with the maximum marginal gain to $\mathcal{I}_{key}$ and merge its visible voxels into $\mathcal{U}$, stopping when K frames are selected. This strategy filters redundant background views, maximizing the coverage of high-value geometric regions while minimizing data redundancy.

Hierarchical VLM Reasoning. With the selected keyframes $\mathcal{I}_{key}$, we execute a two-stage VLM interaction to establish the aforementioned priors. The first stage performs semantic label extraction. We input the referring text $\mathcal{T}$ and the keyframes $\mathcal{I}_{key}$ into the VLM to extract a concise semantic label set $\mathcal{C} = \{C_1, C_2, \ldots, C_M\}$ from the verbose description. The second stage performs 2D geometric localization. We feed the extracted label C_j, the original text $\mathcal{T}$, and the keyframes $\mathcal{I}_{key}$ back into the model. Prompted accordingly, the VLM outputs a normalized 2D bounding box for the target object in each keyframe. These bounding boxes act as crucial spatial priors for filtering 3D Gaussian artifacts in subsequent lifting stages.

3.3 2D Semantic Mask Generation and 3D Lifting

With the semantic and spatial priors established by the VLM, this module instantiates these 2D cues into a precise 3D segmentation. We first extract and adaptively fuse high-fidelity 2D masks using the extracted labels. Crucially, we then lift these 2D observations into the 3D Gaussian space by exploiting intrinsic volume rendering properties, and perform strict cross-view background cropping.

Adaptive View Selection and Mask Fusion. We employ an VLM to parse the referring expression into a concise semantic label set $\mathcal{C} = \{C_j\}_{j=1}^{L}$, containing L semantic labels, which are then input into SAM3 to extract multi-view 2D masks. For each category c_j and frame I_i, SAM3 generates multiple candidate masks. Given a text prompt, SAM3 generates multiple candidate masks for each frame I_i. Let $m_i^{(1)}$ and $m_i^{(2)}$ be the dominant mask with the highest confidence

and the sub-optimal mask in frame I_i, with confidences $c_i^{(1)}$ and $c_i^{(2)}$ respectively. To prevent noisy masks in poor views from degrading 3D multi-view geometric consistency, we adaptively filter the image sequence. We define a high-confidence threshold τ_{high}, a base threshold τ_{base}, and a fallback threshold τ_{safe} to form candidate view sets $\mathcal{I}_{high} = \{I_i \mid c_i^{(1)} > \tau_{high}\}$, $\mathcal{I}_{base} = \{I_i \mid c_i^{(1)} > \tau_{base}\}$, and $\mathcal{I}_{safe} = \{I_i \mid c_i^{(1)} > \tau_{safe}\}$. To balance semantic purity and geometric coverage, we determine the final valid view set $\mathcal{I}_{sel}$ with quantity threshold N_{target} and N_{safe} as follows:

$$\mathcal{I}_{sel} = \begin{cases} \mathcal{I}_{high} & \text{if } |\mathcal{I}_{high}| \geq N_{\text{target}} \\ \text{Top}_{N_{\text{target}}}(\mathcal{I}_{base}) & \text{else if } |\mathcal{I}_{base}| > 0 \\ \text{Top}_{\min(|\mathcal{I}_{safe}|, N_{\text{safe}})}(\mathcal{I}_{safe}) & \text{otherwise} \end{cases} \tag{4}$$

This strategy ensures that we prioritize high-confidence views while maintaining a sufficient number of frames (up to N_{target}) for reconstruction. Next, for each $I_i \in \mathcal{I}_{sel}$ and each category $C_j \in \mathcal{C}$, we generate a per-category mask $M_{i,j}$. Because SAM3 often over-segments objects, we merge the sub-optimal mask $m_i^{(2)}$ with the dominant mask $m_i^{(1)}$ if its confidence $c_i^{(2)}$ exceeds a strict threshold τ_{merge}:

$$M_i = m_i^{(1)} \cup (\mathbb{I}(c_i^{(2)} > \tau_{merge}) \cdot m_i^{(2)}) \tag{5}$$

To aggregate the semantic label sets corresponding to each referring expression, the final semantic mask $\hat{M}_i$ for frame I_i is obtained via a cross-category union:

$$\hat{M}_i = \bigcup_{j=1}^{L} M_{i,j} \tag{6}$$

Mask-Based Back-Projection Initialization. To lift 2D semantics to 3D space, we leverage the volume rendering properties of 3DGS to assign semantic labels to individual Gaussians. First, we calculate the rendering contribution of a single Gaussian from a specific view. In standard forward rendering, the contribution weight $w(r, g_j)$ of the j-th Gaussian g_j along ray r is defined by its accumulated transmittance and opacity:

$$w(r, g_j) = T(r, g_j)\alpha(r, g_j) \tag{7}$$

where $T(r, g_j)$ is the accumulated transmittance before the ray reaches g_j, and $\alpha(r, g_j)$ is the opacity. We then aggregate the multi-view semantic responses for each Gaussian. For a Gaussian g_j, we check its projected pixel set $\mathcal{R}_i$ across all valid views $I_i \in \mathcal{I}_{sel}$. Using the 2D final mask value $M_i(r) \in \{0, 1\}$ at each pixel, we compute a global score $W_k(g_j)$ for g_j being foreground ($k = 1$) or background ($k = 0$):

$$W_k(g_j) = \sum_{I_i \in \mathcal{I}_{sel}} \sum_{r \in \mathcal{R}_i} \mathbb{I}(M_i(r) = k) w_i(r, g_j) \tag{8}$$

Finally, we build the initial foreground 3D Gaussian set $\mathcal{G}_{fg}$ using hard assignment. A Gaussian is added to $\mathcal{G}_{fg}$ if its foreground score is strictly higher than its background score ($W_1(g_j) > W_0(g_j)$). Otherwise, it is classified as background.

Cross-View Background Cropping. To further improve the purity of the 3D semantic field, we design a cross-view background cropping mechanism. For each Gaussian g currently in $\mathcal{G}_{fg}$, we compute its 2D projection across all valid views in $\mathcal{I}_{sel}$. We count the number of times it projects within the valid imaging area as $N_{val}(g)$, and the number of times it falls onto a background region as $N_{conflict}(g)$. We define the conflict ratio as:

$$\rho_g = \frac{N_{conflict}(g)}{N_{val}(g) + \epsilon}, \tag{9}$$

where ϵ is a small constant introduced to avoid division by zero.If ρ_g exceeds a threshold τ_{conf}, the Gaussian is deemed a geometric artifact, and its label is corrected to background. This mechanism leverages dense multi-view consensus to significantly reduce boundary noise.

3.4 Multi-View Verification and Spatial Refinement

While mask-based lifting and background cropping provide initial 3D segmentation, mask inaccuracies introduce artifacts. To resolve this, we enforce multi-view verification and spatial continuity.

Cross-View Back-Projection Verification. Utilizing the keyframe set $\mathcal{I}_{key}$, we reference the 2D bounding box $\mathcal{B}_i$ of the target in each keyframe $I_i \in \mathcal{I}_{key}$ generated by the VLM. If there are multiple instances, we take their spatial union. For each Gaussian $g \in \mathcal{G}_{fg}$, we project it onto keyframe I_i to obtain the 2D pixel coordinates $\mathbf{u}_{g,i}$. We then count the valid observation frequency $N_{vis}(g)$ and the out-of-bounds frequency $N_{out}(g)$ for Gaussian g across the keyframes:

$$N_{vis}(g) = \sum_{I_i \in \mathcal{I}_{key}} \mathbb{I}_{vis}(g, I_i), \quad N_{out}(g) = \sum_{I_i \in \mathcal{I}_{key}} \mathbb{I}_{out}(g, I_i) \tag{10}$$

Here, the visibility indicator $\mathbb{I}_{vis}(g, I_i) = 1$ if $\mathbf{u}_{g,i}$ is within the image viewport. The out-of-bounds indicator $\mathbb{I}_{out}(g, I_i) = 1$ if the point is visible in the viewport but falls outside $\mathcal{B}_i$. To avoid over-cropping the target geometry due to single-view occlusions or 2D localization errors, we compute the cross-view out-of-bounds ratio $R_{out}(g) = \frac{N_{out}(g)}{N_{vis}(g)+\epsilon}$ as a robust filtering metric, where ϵ is a small constant. We introduce a minimum observation threshold τ_{views} and an out-of-bounds tolerance threshold τ_{box}. If a Gaussian g satisfies $N_{vis}(g) \geq \tau_{views}$ and $R_{out}(g) > \tau_{box}$, it is predominantly outside the bounding box across multiple views. Consequently, we correct it to background; otherwise, it retains its foreground label.

Local Refinement via Spatial Consistency. Occlusions and mask edge errors often create internal cavities within 3D targets. To reconstruct structural integrity, we apply a 3D K-Nearest Neighbor (KNN) spatial label diffusion algorithm using KD-Trees. For an unlabeled Gaussian g_{null}, we query its k nearest

neighboring Gaussians. If the fraction of neighbors belonging to the foreground set $\mathcal{G}_{fg}$ exceeds a reliability threshold, i.e.,

$$\frac{1}{k} \sum_{g_j \in \mathrm{KNN}(g_{null})} \mathbb{I}(g_j \in \mathcal{G}_{fg}) \geq \tau_{knn} \tag{11}$$

we treat the Gaussian as a missing internal structure and update its label to foreground. This diffusion mechanism effectively fills local discontinuities, ensuring the final 3D semantic output is structurally coherent and complete.

4 Experiments

4.1 Benchmark

To advance research on Generalized Referring 3D Gaussian Splatting Segmentation (GR3DGS), we introduce two highly challenging benchmark datasets: **GR-LERF** and **GR-ScanNet**. For 2D pixel-level alignment, we build **GR-LERF** on top of LERF scenes [16], and manually curate a large collection of GR3DGS referring instructions to evaluate pixel-level comprehension under the GR3DGS setting. However, datasets for assessing point-level understanding in GR3DGS remain scarce. To fill this gap, we construct **GR-ScanNet** based on ScanNet [3], adopting the 10 representative indoor scenes used in OpenGaussian [38] and designing complex GR3DGS instructions tailored to these scenes. All instructions in both benchmarks are produced via a rigorous manual annotation and cross-validation protocol (see Appendix), ensuring accurate ground-truth supervision.

4.2 Implementation Details

We implement ZeroSplat in PyTorch using a single NVIDIA RTX 4090D GPU. For geometry guided keyframe selection, we discretize scenes with a base voxel size of s = 0.1, compute curvature within a radius of $r = \eta s = 0.25$ at a resolution of $\gamma s = 0.05$, where $\gamma = 0.5$ and $\eta = 2.5$, and rescale surface variation to a saliency interval $[w_{min}, w_{max}] = [1, 10]$ to greedily select $K = 30$ keyframes ($\mathcal{I}_{key}$). We deploy qwen3-vl-30b-a3b-instruct on these keyframes to extract labels and generate 2D bounding boxes. We set high-confidence threshold $\tau_{high} = 0.6$ and a base threshold $\tau_{base} = 0.3$, with a quantity threshold of $N_{target} = 30$ and $N_{safe} = 6$. Specifically, we employ a fallback threshold $\tau_{safe} = 0$ for open-vocabulary datasets, while $\tau_{safe} = 0.15$ is used for referring expression datasets. Mask fusion applies a strict threshold $\tau_{merge} = 0.8$. Cross view back-Projection Verification necessitates a conflict ratio threshold $\tau_{conf} = 0.8$. Cross view background cropping necessitates a minimum observation threshold $\tau_{views} = 8$ and an out-of-bounds tolerance threshold $\tau_{box} = 0.8$, with ϵ a small constant set to 10^{-6}. For geometric refinement, we apply 3D KNN spatial diffusion over $k = 40$ neighbors with a reliability threshold $\tau_{knn} = 0.8$.

Table 1: Comparison of representative 3D semantic understanding methods by task support and resource cost. "Referring Task" indicates language-based referring segmentation (ours supports generalized $0/1/N$). "Scene Opt." denotes per-scene optimization. "Train Time", "Storage", and "Peak VRAM" report per-scene optimization time, extra feature storage beyond 3DGS, and peak GPU memory usage, respectively.

Method	Venue	Domain	Referring Task	Scene Opt.	Train Time	Storage	Peak VRAM
2D pixel-level methods							
LEGaussians [32]	CVPR'24	2D Pixel	No	Required	∼2h	∼3GB	∼20 GB
LangSplat [29]	CVPR'24	2D Pixel	No	Required	∼2h	∼3GB	∼20 GB
Feature-3DGS [43]	CVPR'24	2D Pixel	No	Required	∼1h	∼3GB	∼26 GB
GS-Grouping [41]	ECCV'24	2D Pixel	No	Required	∼1h	-	∼28 GB
GOI [30]	MM'24	2D Pixel	No	Required	∼1h	-	∼24 GB
Occam's LGS [2]	BMVC'25	2D Pixel	No	None	None	∼3GB	∼12 GB
3DVLGS [28]	ICLR'25	2D Pixel	No	Required	∼2h	∼3GB	∼28 GB
ReferSplat [7]	ICML'25	2D Pixel	Yes (single-target)	Required	∼2h	∼3GB	∼28 GB
3D point-level methods							
OpenGaussian [38]	NeurIPS'24	3D Point	No	Required	∼1h	∼3GB	∼22 GB
InstanceGaussian [20]	CVPR'25	3D Point	No	Required	∼2h	∼3GB	∼24 GB
Dr.Splat(Top-40) [14]	CVPR'25	3D Point	No	None	∼1h	∼3GB	∼24 GB
LUDVIG [25]	ICCV'25	3D Point	No	None	None	∼3GB	∼22 GB
Ours	–	3D Point	Yes (generalized)	None	None	0	∼10 GB

4.3 Efficiency and Versatility Analysis

Table 1 summarizes the trade-off between task capability and resource consumption across different technical paradigms in 3D semantic understanding, positioning ZeroSplat within the current landscape. In terms of task capability, most 3D semantic methods are primarily designed for category-level retrieval and are not applicable to language expressions with complex spatial constraints. While referring segmentation frameworks introduce natural-language interaction, their reasoning is often confined to 2D rendered views and is restricted to the single-target setting, making generalized 3D referring (i.e., handling 0, 1, or N instances) challenging. ZeroSplat fills this gap by enabling generalized referring understanding directly in 3D space. From an efficiency perspective, prior methods typically rely on scene-specific optimization and incur non-trivial training costs, often accompanied by additional feature storage. In contrast, ZeroSplat adopts a decoupled, training-free design that requires neither scene optimization nor extra feature storage; specifically, it stores no semantic features per Gaussian and supports plug-and-play inference. This design offers a practical pathway for deploying 3D semantic understanding on resource-constrained devices.

4.4 Generalized Referring 3D Gaussian Splatting Segmentation

Settings. 1) Task. We evaluate GR3DGS, where a model must segment the targets specified by a natural-language instruction. The instruction may correspond to 0, 1, or N instances. **2) Datasets.** We evaluate on GR-LERF and GR-ScanNet. GR-LERF measures pixel-level segmentation accuracy on 2D rendered views, while GR-ScanNet measures point-level segmentation accuracy in 3D space. **3) Baselines.** We compare with representative 3D semantic understanding methods. We use the hyperparameters and model settings reported in

Table 2: Quantitative results on GR-LERF and GR-ScanNet, measured by mIoU.

	GR-LERF					GR-ScanNet
Method	Ramen	Teatime	Figurines	Waldo	Mean	mIoU
Pixel-based						
LangSplat [29]	12.7	34.6	15.6	11.4	18.6	-
GOI [30]	7.5	**60.0**	18.8	40.2	31.6	-
LEGaussians [32]	4.2	2.5	2.7	4.5	3.5	-
Occam's LGS [2]	9.8	45.7	16.3	12.9	21.2	-
GS-Grouping [41]	29.1	49.2	15.6	19.3	28.3	-
Feature-3DGS [43]	1.6	11.4	0.6	17.4	7.8	-
ReferSplat [7]	4.3	10.9	11.6	8.3	8.8	-
Point-based						
OpenGaussian [38]	7.3	10.5	4.5	9.1	7.9	19.8
InstanceGaussian [20]	6.9	6.8	11.3	14.7	9.9	24.5
Dr.Splat(Top-40) [14]	12.0	8.7	7.1	11.7	9.9	21.5
LUDVIG [25]	25.3	32.5	23.9	13.5	23.8	15.1
Ours	**46.5**	56.3	**52.1**	**48.4**	**50.8**	**41.2**

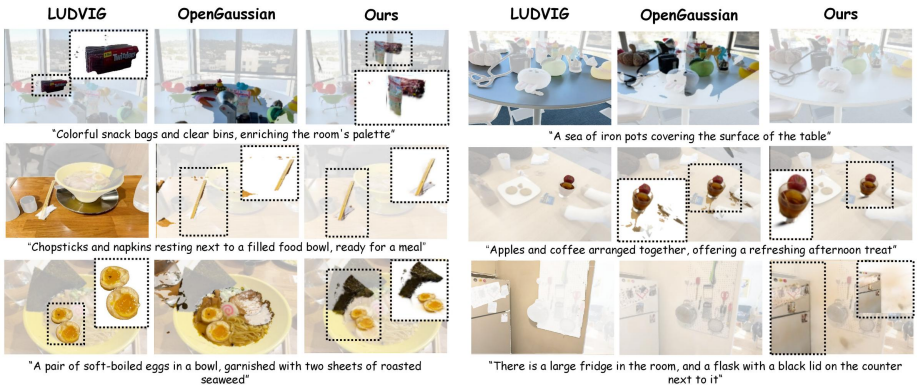

Fig. 3: Qualitative results on object selection from the GR-LERF dataset.

the original papers to ensure fair comparisons. Due to representation constraints, some pixel-only methods are not evaluated on GR-ScanNet.

Results on GR-LERF. As shown in Table 2, ZeroSplat significantly outperforms all baselines in pixel-level segmentation. Qualitatively, as shown in Figure 3, prior methods like Opengaussian and LUDVIG struggle with multi-target scenes, often merging or missing nearby instances due to over-smoothed embeddings. In contrast, ZeroSplat uses VLM-based compositional reasoning to accurately segment all text-specified targets, yielding significantly higher recall.

Results on GR-ScanNet. Quantitatively, ZeroSplat surpasses existing frameworks and establishes a new state-of-the-art for intrinsic 3D point-level segmentation, as shown in Table 2. Qualitatively, visual comparisons in Figure 4 reveal that baseline methods struggle to interpret complex spatial and functional queries. InstanceGaussian and DrSplat exhibit severe over-segmentation, frequently bleeding into irrelevant background regions, whereas OpenGaussian fails to effectively localize targets, resulting in sparse or missing predictions. In

Table 3: Quantitative results for R3DGS on Ref-LERF measured by mIoU.

Method	Ramen	Teatime	Figurines	Waldo	Mean
Grounded SAM [31]	14.1	16.9	16.0	16.2	15.8
LangSplat [29]	12.0	7.6	17.9	17.9	13.9
SPIn-NeRF [26]	7.3	11.7	9.7	10.3	9.8
GS-Grouping [41]	27.9	14.8	8.6	6.3	14.4
GOI [30]	27.1	22.9	16.5	15.7	20.5
ReferSplat [7]	**35.2**	31.3	25.7	**24.4**	29.2
Ours	30.4	**41.4**	**37.8**	21.3	**32.7**

Table 4: Quantitative results for open-vocabulary 3D semantic segmentation on ScanNet measured by mIoU.

Method	19 cls.	15 cls.	10 cls.
OpenGaussian [38]	24.7	30.1	38.3
InstanceGaussian [20]	40.7	42.5	47.9
Dr.Splat(Top-40) [14]	29.6	38.2	**50.8**
LUDVIG [25]	33.9	37.4	46.4
Ours	**44.5**	**43.7**	49.7

Table 5: Quantitative results for open-vocabulary object selection on LERF measured by mIoU.

Method	Ramen	Teatime	Figurines	Waldo	Mean
Pixel-based					
LEGaussians [32]	46.0	60.3	40.8	39.4	46.6
LangSplat [29]	51.2	65.1	44.7	44.5	51.4
Feature-3DGS [43]	43.7	58.8	40.5	39.6	45.7
GS-Grouping [41]	45.5	60.9	40.0	38.7	46.3
GOI [30]	52.6	63.7	44.5	41.4	50.6
ReferSplat	55.1	50.1	**67.5**	48.9	55.4
Occam's LGS [2]	51.0	70.2	58.6	**65.3**	61.3
3DVLGS [28]	**61.4**	**73.5**	58.1	54.8	**62.0**
Point-based					
OpenGaussian [38]	31.0	60.4	39.3	22.7	38.4
InstanceGaussian [20]	24.6	**63.4**	45.5	29.2	40.7
Dr.Splat(Top-40) [14]	24.7	57.2	53.4	39.1	43.6
LUDVIG [25]	42.3	58.6	58.0	42.8	50.4
Ours	**42.7**	57.6	**58.1**	**51.2**	**52.4**

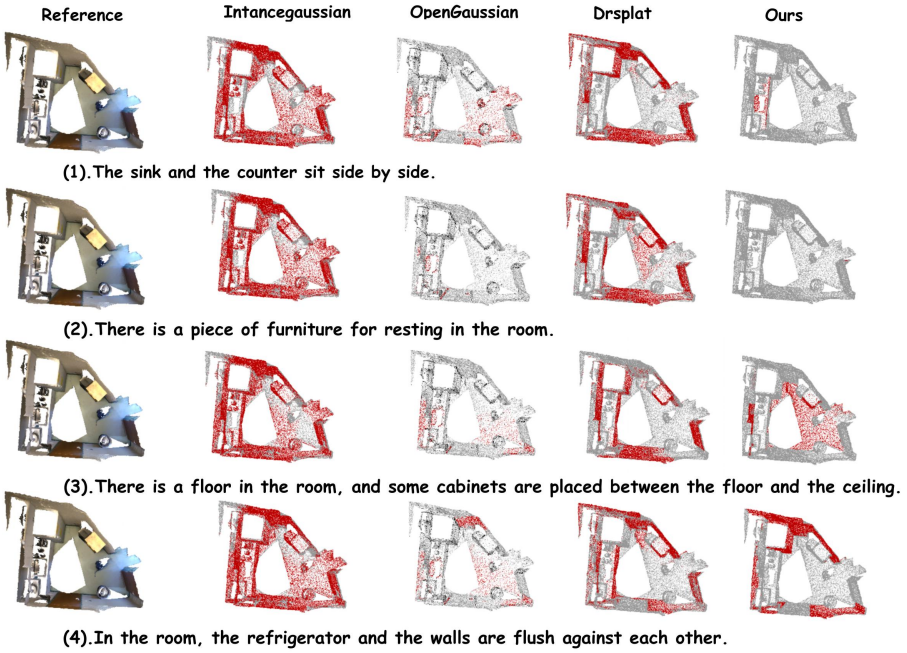

Fig. 4: Qualitative results on object selection from the GR-ScanNet dataset.

contrast, our approach accurately grounds intricate natural language descriptions, isolating the exact queried instances with crisp boundaries and high semantic purity.

Table 6: Ablation of core components on GR-ScanNet.

VLM	KNN	mIoU
		24.7
✓		38.6
	✓	26.2
✓	✓	**41.2**

Table 7: Ablation of internal VLM mechanisms on GR-ScanNet.

Method	mIoU
w/o Semantic Label Extraction	24.3
w/o Bounding Box Filter	40.1
Random Keyframe Sampling	39.6
Full VLM Module	**41.2**

Table 8: Comparison of different spatial refinement strategies on GR-ScanNet.

Method	mIoU
w/o Refinement	38.6
3D Radius Search	39.8
3D KNN Diffusion (Ours)	**41.2**

4.5 Referring 3D Gaussian Splatting Segmentation

Settings. 1) Task. This task requires localizing and segmenting a single target object in a 3D scene given a complex natural-language instruction. Unlike category-level retrieval, the instruction often specifies spatial relations or fine-grained attributes, which demands strong understanding of 3D structure and object properties. **2) Baselines.** We compare ZeroSplat against ReferSplat [7] and several pixel-level understanding methods. Notably, unlike these baselines, our approach distinguishes itself by achieving intrinsic point-level understanding. **3) Datasets.** We evaluate on the standard test set released by ReferSplat.
Results. As shown in Table 3, ZeroSplat achieves a strong mean mIoU of 32.7. Notably, although ZeroSplat is entirely training-free, it still outperforms the fully supervised ReferSplat (+3.5 mIoU).

4.6 Open-Vocabulary 3D Gaussian Splatting Segmentation

Settings. 1) Task. This task localizes and segments a single target object in a 3D scene given a textual category. **2) Datasets.** We conduct extensive evaluations on LERF and ScanNet to assess performance from complementary perspectives. Specifically, we use LERF to measure 2D performance by comparing rendered-view segmentations against its 2D annotations, and we use ScanNet to measure 3D performance using its precise point-level semantic labels to directly evaluate 3D localization and Gaussian segmentation. **3) Baselines.** We compare against a range of representative open-vocabulary understanding methods. Due to representational limitations, some pixel-only methods are not evaluated on ScanNet. **4) Adaptation.** For this task, we bypass the VLM-driven semantic parsing and geometric anchoring stages, as they are specifically designed for complex referring instructions. Instead, we directly use the category names as semantic labels for our 2D-to-3D lifting and refinement pipeline.
Results. Although our method is not designed for open-vocabulary semantic segmentation benchmarks, ZeroSplat demonstrates stable and competitive performance. As shown in Tables 4 and 5, ZeroSplat delivers leading results on both datasets: it achieves 52.4 mIoU on LERF, surpassing the previous best by 2.0 mIoU and establishing a new state of the art; on ScanNet, it attains the best performance under both the 19-class and 15-class evaluation protocols. These results suggest that our pipeline generalizes well beyond its original design goal and remains effective even under this simplified category-driven setting.

4.7 Ablation Study

To evaluate the contribution of each proposed component, we conduct extensive ablation studies on the GR-ScanNet dataset using the mIoU metric.

Effectiveness of Key Components. We validate our core components in Table 6. The baseline mask-lifting strategy yields a sub-optimal 24.7 mIoU due to unconstrained multi-view inconsistencies. Integrating the VLM for spatial anchoring significantly boosts performance to 38.6 mIoU (+13.9), effectively pruning cross-view false positives. Applying the 3D KNN diffusion alone improves structural completeness to 26.2 mIoU (+1.5). Combining both modules yields the best performance (41.2 mIoU), demonstrating that global semantic localization and local geometric refinement are highly complementary.

VLM Parsing and Geometric Anchoring. Table 7 ablates the VLM module's mechanisms. Removing semantic label extraction causes a severe performance collapse (to 24.3 mIoU), as feeding free-form prompts into the 2D segmentation model introduces high ambiguity. Replacing geometry-guided keyframe selection with uniform random sampling decreases mIoU to 39.6, highlighting the need for geometric variation to resolve multi-view ambiguities. Discarding the 2D bounding box filter drops performance to 40.1 mIoU, confirming its necessity as a spatial prior for suppressing misclassified background Gaussians.

Spatial Refinement Strategies. Table 8 compares strategies for repairing internal cavities caused by view-dependent occlusions. A fixed-radius 3D search struggles (39.8 mIoU) because Gaussian density varies drastically across scenes; a static radius fails to balance sparse and dense regions. In contrast, our 3D KNN diffusion dynamically adapts to local point density, robustly filling structural voids while preserving sharp object boundaries, achieving the peak 41.2 mIoU.

5 Conclusion

In this paper, we introduce the Generalized Referring 3D Gaussian Splatting Segmentation (GR3DGS) task. Unlike the conventional single-target paradigm, GR3DGS addresses real-world complexities by handling multiple targets and absent objects. To tackle this task, we propose ZeroSplat, a zero-feature and training-free framework for intrinsic point-level 3D scene understanding. By lifting 2D Vision-Language Model (VLM) priors into 3D space via multi-view geometric constraints and spatial diffusion, ZeroSplat eliminates the need for expensive per-scene optimization and auxiliary semantic feature storage. Furthermore, we construct two benchmarks, GR-LERF and GR-ScanNet, to evaluate performance at both the pixel and point levels. Extensive experiments show that ZeroSplat significantly outperforms existing state-of-the-art methods in generalized, single-target, and open-vocabulary scenarios, while maintaining high efficiency as a plug-and-play solution. We believe our framework and benchmarks will serve as a solid foundation for future research in Embodied AI and interactive 3D scene understanding.

References

1. Chen, N., Liu, L., Li, Z., Zeng, Z., Zhu, Z., Cong, W., Hong, J., Yang, Y., Tu, Z., Wang, Y., et al.: A physics-grounded benchmark for multi-agent dynamics in world models. In: The 2nd Workshop on Foundation Models Meet Embodied Agents at CVPR 2026
2. Cheng, J., Zaech, J.N., Gool, L.V., Paudel, D.P.: Occam's lgs: An efficient approach for language gaussian splatting. In: 36th British Machine Vision Conference 2025, BMVC 2025, Sheffield, UK, November 24-27, 2025. BMVA (2025)
3. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017)
4. Ding, J., Liu, X., Pan, Z., Long, S., Li, G.: Extrinsplat: Decoupling geometry and semantics for open-vocabulary understanding in 3d gaussian splatting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 31019–31028 (2026)
5. Ding, J., Tang, H., Jin, H., Gao, W., Li, G.: 3d instruction ambiguity detection. arXiv preprint arXiv:2601.05991 (2026)
6. He, S., Ding, H.: Refmask3d: Language-guided transformer for 3d referring segmentation. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 8316–8325 (2024)
7. He, S., Jie, G., Wang, C., Zhou, Y., Hu, S., Li, G., Ding, H.: ReferSplat: Referring segmentation in 3d gaussian splatting. In: International Conference on Machine Learning (ICML)
8. Huang, P.H., Lee, H.H., Chen, H.T., Liu, T.L.: Text-guided graph neural networks for referring 3d instance segmentation. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, pp. 1610–1618 (2021)
9. Jiang, M., Jia, S., Gu, J., Lu, X., Zhu, G., Dong, A., Zhang, L.: Votesplat: Hough voting gaussian splatting for 3d scene understanding. arXiv preprint arXiv:2506.22799 (2025)
10. Jin, H., Ding, J., Xie, S., Luo, G., Li, G.: Vista: Mitigating semantic inertia in video-llms via training-free dynamic chain-of-thought routing. arXiv preprint arXiv:2505.11830 (2026)
11. Jin, H., Wang, C., Tang, H., Du, Z., Jiang, X., Tian, J., Zhang, Q., Ding, J.: Contextguard: Structured self-auditing for context learning in language models. arXiv preprint arXiv:2605.26827 (2026)
12. Jin, H., Zhu, M., Tian, J., Jiang, X., Du, Z., Tang, H., Xie, S., Zhang, Q., Ding, J.: Context-cot: Enhancing context learning via high-quality reasoning synthesis. arXiv preprint arXiv:2605.25354 (2026)
13. Jin, H., Zhu, R., Ding, J., Zhang, W., Li, G.: Himac: Hierarchical macro-micro learning for long-horizon llm agents. arXiv preprint arXiv:2603.00977 (2026)
14. Jun-Seong, K., Kim, G., Yu-Ji, K., Wang, Y.C.F., Choe, J., Oh, T.H.: Dr. splat: Directly referring 3D gaussian splatting via direct language embedding registration. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 14137–14146 (2025)
15. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3D Gaussian Splatting for Real-Time Radiance Field Rendering. arXiv preprint arXiv:2308.04079 (2023), arXiv:2308.04079
16. Kerr, J., Kim, C.M., Goldberg, K., Kanazawa, A., Tancik, M.: Lerf: Language embedded radiance fields. In: International Conference on Computer Vision (ICCV) (2023)

17. Kim, M.J., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., et al.: Openvla: An open-source vision-language-action model. arXiv preprint arXiv:2406.09246 (2024)
18. Li, C., Xu, Y., Feng, J., Ding, J.: Lmm-track4d: Eliciting 4d dynamic reasoning in lmms via trajectory-grounded dialogue. arXiv preprint arXiv:2605.19390 (2026)
19. Li, D., Liu, L., Liu, B., Zhou, S., Feng, J., Lu, Z., Zheng, M., You, C., Fan, Z.: Egocentric world model for photorealistic hand-object interaction synthesis. arXiv preprint arXiv:2603.13615 (2026)
20. Li, H., Wu, Y., Meng, J., Gao, Q., Zhang, Z., Wang, R., Zhang, J.: Instance-gaussian: Appearance-semantic joint gaussian representation for 3D instance-level perception. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 14078–14088 (2025)
21. Liang, S., Wang, S., Li, K., Niemeyer, M., Gasperini, S., Navab, N., Tombari, F.: Supergseg: Open-vocabulary 3d segmentation with structured super-gaussians. arXiv preprint arXiv:2412.10231 (2024)
22. Liu, C., Lin, Z., Shen, X., Yang, J., Lu, X., Yuille, A.: Recurrent multimodal interaction for referring image segmentation. In: Proceedings of the IEEE international conference on computer vision. pp. 1271–1280 (2017)
23. Liu, L., Li, D., Liang, Y., Jiang, S., Vijay, H., Hu, H., Xu, X., Liu, Z., Shakkottai, S., Li, M., et al.: Egotl: Egocentric think-aloud chains for long-horizon tasks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2017–2027 (2026)
24. Liu, X., Xu, X., Li, J., Zhang, Q., Wang, X., Sebe, N., Ma, L.: Less: Label-efficient and single-stage referring 3d segmentation. Advances in Neural Information Processing Systems **37**, 11164–11185 (2024)
25. Marrie, J., Menegaux, R., Arbel, M., Larlus, D., Mairal, J.: Ludvig: Learning-free uplifting of 2d visual features to gaussian splatting scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2025)
26. Mirzaei, A., Aumentado-Armstrong, T., Derpanis, K.G., Kelly, J., Brubaker, M.A., Gilitschenski, I., Levinshtein, A.: Spin-nerf: Multiview segmentation and perceptual inpainting with neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20669–20679 (2023)
27. Munkberg, J., Hasselgren, J., Shen, T., Gao, J., Chen, W., Evans, A., Müller, T., Fidler, S.: Extracting triangular 3d models, materials, and lighting from images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8280–8290 (2022)
28. Peng, Q., Planche, B., Gao, Z., Zheng, M., Choudhuri, A., Chen, T., Chen, C., Wu, Z.: 3d vision-language gaussian splatting. arXiv preprint arXiv:2410.07577 (2024)
29. Qin, M., Li, W., Zhou, J., Wang, H., Pfister, H.: Langsplat: 3D language gaussian splatting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20051–20060 (2024)
30. Qu, Y., Dai, S., Li, X., Lin, J., Cao, L., Zhang, S., Ji, R.: GOI: Find 3D gaussians of interest with an optimizable open-vocabulary semantic-space hyperplane. In: Proceedings of the 32nd ACM International Conference on Multimedia. ACM (2024)
31. Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., et al.: Grounded sam: Assembling open-world models for diverse visual tasks. arXiv preprint arXiv:2401.14159 (2024)
32. Shi, J.C., Wang, M., Duan, H.B., Guan, S.H.: Language embedded 3D gaussians for open-vocabulary scene understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5333–5343 (2024)

33. Sun, W., Zhou, Y., Jiao, J., Li, Y.: Cags: Open-vocabulary 3d scene understanding with context-aware gaussian splatting. arXiv preprint arXiv:2504.11893 (2025)
34. Tang, H., Cao, M., Liu, R., Liang, X., Li, L., Li, G., Liang, X.: Video spatial reasoning with object-centric 3d rollout. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 40, pp. 9395–9403 (2026)
35. Wang, Z., Lu, Y., Li, Q., Tao, X., Guo, Y., Gong, M., Liu, T.: Cris: Clip-driven referring image segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11686–11695 (2022)
36. Wu, C., Ji, J., Wang, H., Ma, Y., Huang, Y., Luo, G., Fei, H., Sun, X., Ji, R., et al.: Rg-san: Rule-guided spatial awareness network for end-to-end 3d referring expression segmentation. Advances in Neural Information Processing Systems **37**, 110972–110999 (2024)
37. Wu, C., Ma, Y., Chen, Q., Wang, H., Luo, G., Ji, J., Sun, X.: 3d-stmn: Dependency-driven superpoint-text matching network for end-to-end 3d referring expression segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 5940–5948 (2024)
38. Wu, Y., Meng, J., Li, H., Wu, C., Shi, Y., Cheng, X., Zhao, C., Feng, H., Ding, E., Wang, J., et al.: Opengaussian: Towards point-level 3d gaussian-based open vocabulary understanding. Advances in Neural Information Processing Systems **37**, 19114–19138 (2024)
39. Xiao, X., Zhang, Y., Li, X., Wang, T., Wang, X., Wei, Y., Hamm, J., Xu, M.: Visual instance-aware prompt tuning. In: Proceedings of the 33rd ACM International Conference on Multimedia. pp. 2880–2889 (2025)
40. Xiao, X., Zhang, Y., Zhao, L., Liu, Y., Liao, X., Mai, Z., Li, X., Wang, X., Xu, H., Hamm, J., Lin, X., Xu, M., Wang, Q., Wang, T., Han, C.: Prompt-based adaptation in large-scale vision models: A survey. Transactions on Machine Learning Research (2026)
41. Ye, M., Danelljan, M., Yu, F., Ke, L.: Gaussian grouping: Segment and edit anything in 3D scenes. In: European conference on computer vision. pp. 162–179. Springer (2024)
42. Yin, H., Zhan, H., Xu, Y., Yeh, R.A.: Semantic consistent language gaussian splatting for point-level open-vocabulary querying. arXiv preprint arXiv:2503.21767 (2025)
43. Zhou, S., Chang, H., Jiang, S., Fan, Z., Zhu, Z., Xu, D., Chari, P., You, S., Wang, Z., Kadambi, A.: Feature 3Dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21676–21685 (2024)