SplatReasoner: Enhancing Embodied Reasoning and Grounding by Novel View Synthesis

Kim Yu-Ji[1] Dahye Lee[2] Kim Jun-Seong[1] Nam Hyeon-Woo[1]
GeonU Kim[2] Yongjin Kwon[3] Yu-Chiang Frank Wang[4]
Jaesung Choe[4] Tae-Hyun Oh[2]

[1]POSTECH [2]KAIST [3]ETRI [4]NVIDIA
ugkim@postech.ac.kr
https://splatreasoner.github.io/

Abstract. Vision-Language Models (VLMs) provide strong image/video-level reasoning, but their use in embodied scene understanding remains limited by the fixed viewpoints available in episodic RGB-D memories. Such observations can miss query-relevant evidence due to occlusion, object truncation, limited fields of view, or suboptimal view composition. We present SplatReasoner, a framework that introduces novel view synthesis into the VLM reasoning process by leveraging 3D Gaussian Splatting (3DGS). Given a user query about a 3D scene, SplatReasoner retrieves and synthesizes camera viewpoints that provide the most relevant visual evidence for answering the query and grounding the referred entities in 3D. Experiments show that query-conditioned novel view synthesis improves embodied reasoning and grounding over fixed-view memory and language-embedded 3DGS baselines.

Keywords: Embodied Reasoning · 3D Grounding · Novel View Synthesis

1 Introduction

Building embodied agents capable of understanding 3D environments [30] and executing natural-language instructions [6, 64] remains a foundational goal in computer vision and robotics. Recently, Vision-Language Models (VLMs) [3, 7, 15, 21, 32] have demonstrated advanced reasoning and grounding capability for semantic understanding. As these models have proven effective at interpreting image and video data, research has naturally pivoted toward applying their capabilities to embodied tasks in 3D scenes. However, bridging this gap is a critical challenge, as it requires equipping 2D-native VLMs with a cohesive memory of past observations and a robust, structural understanding of 3D space.

Previous approaches [4, 7, 9, 14, 15, 35, 57] demonstrate that VLMs can perform complex reasoning and grounding across multi-view images when provided with structured memory built from RGB-D observations. While these approaches make notable progress toward memory-based embodied reasoning, they typically operate on a set of pre-captured image sequences as a structured memory and

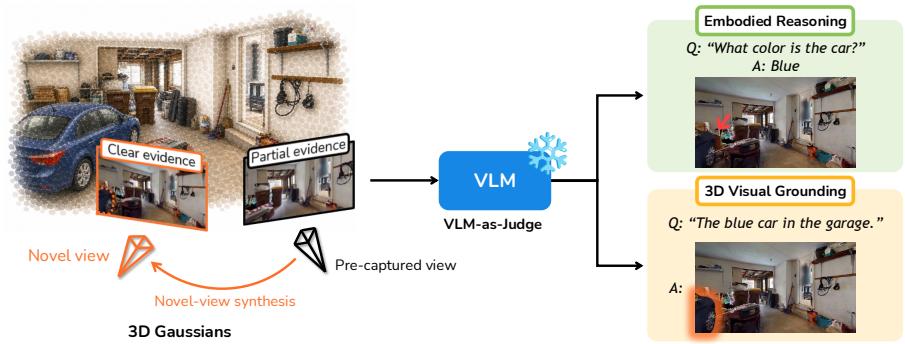

Fig. 1: SplatReasoner aims to enhance embodied reasoning capacity by integrating VLM with novel view synthesis of the 3DGS representation. Specifically, SplatReasoner retrieves and synthesizes informative camera viewpoints to provide the most relevant visual evidence, thereby facilitating improved query answering and 3D entity grounding.

object-level abstractions, which naturally constrain the investigation of 3D environments into a fixed set of 2D views. Relying on a fixed set of pre-recorded images inherently restricts a VLM's reasoning and grounding capabilities. Consequently, these models often struggle with sub-optimal view compositions, such as occlusions, small objects, and truncations at image boundaries. While the state-of-the-art methods, including recent 3D-Mem [57], demonstrated strong reasoning capabilities, they particularly faces challenges in these cases by design.

In this context, 3D Gaussian Splatting (3DGS) [24] can be an advantageous representation alternative to the fixed-view based memory by virtue of its favorable novel view synthesis functionality. As a separate line of research, language-driven 3D scene understanding methods [1, 10, 17, 23, 29, 37, 41, 48, 53, 54] have been proposed to connect 3D and language with the novel view synthesis function, enabling open-vocabulary 3D scene understanding. These approaches also successfully demonstrate the effectiveness in alignment between language and 3D representations. However, their similarity computation strategies are primarily optimized for simple, category-level text queries (e.g., "chair" or "table"). Consequently, it remains challenging to extend them to support the complex reasoning and grounding tasks required for embodied applications.

To address these bottlenecks, we introduce SplatReasoner, a framework that performs VLM-guided embodied reasoning and grounding based on novel view synthesis of 3DGS, as shown in Fig. 1. For a given query, SplatReasoner (1) selects initial candidate views by matching simplified queries to 3D Gaussians, and (2) conducts a VLM-as-Judge mechanism with synthesized novel views to finalize the view selection for downstream tasks. Instead of relying on pre-defined fixed views, our pipeline introduces 3D Gaussian novel view synthesis to improve VLM-guided reasoning and grounding for embodied tasks. The contributions of our paper are summarized as follows:

– We introduce SplatReasoner, a novel framework enhancing VLM-guided embodied reasoning and grounding by integrating novel view synthesis of 3DGS.

– We demonstrate the critical importance of VLM-as-Judge with novel view synthesis for spatial reasoning tasks, effectively overcoming the strict viewpoint limitations of existing fixed-view memory approaches.
– We achieve strong performance improvements on embodied question answering, and we demonstrate that our enhanced reasoning pipeline enables precise 3D visual grounding.

2 Related Work

Embodied Reasoning with 2D VLMs. To perform embodied reasoning tasks [4, 9, 35], recent approaches [3, 21, 32, 33, 39, 49] integrate VLMs with a structured memory built from RGB or RGB-D [14,35,57] observations. Specifically, a text-centric method generates captions [5] from RGB data, which are then fed to LLMs [38,50] that reason without direct visual context. In contrast, structured abstraction methods build scene graphs from RGB-D data [14, 25, 46, 59, 62], representing objects as nodes and relationships as edges. Nevertheless, a significant limitation of both approaches is their reliance on high-level abstraction. By discarding low-level geometric and visual details, these methods hinder the fine-grained understanding required for complex tasks.

While the development of VLMs has enabled the use of visually rich, multi-view images, this introduces a new challenge: long visual sequences often overwhelm models and limit effective reasoning [13, 47, 52]. 3D-Mem [57] offers a solution by compressing these observations into "Memory Snapshots." This method removes visual redundancy and stores object-level annotations. Consequently, VLMs can access only the most relevant snapshots for a given query, allowing them to reason efficiently while maintaining the visual information needed for complex question answering and spatial reasoning. A primary bottleneck of these memory-based systems is their reliance on pre-captured, static views. While they may exhibit strong semantic reasoning, they fall short on tasks that demand 3D localization, novel view exploration, or precise spatial grounding.

Language-embedded 3D Gaussian (3DGS). 3DGS [24] is a modern 3D representation that enables fast, high-fidelity 3D reconstruction and real-time rendering. It has evolved beyond rendering, as several works [1,10,17,22,23,29,37, 41,48,53,54,58,65] have augmented it with language features for open-vocabulary reasoning. For example, prior approaches [23, 41, 48, 53] have embedded CLIP features [42] into each Gaussian, enabling open-vocabulary object retrieval and segmentation.

The core challenge with existing embedding-augmented 3DGS approaches is that they primarily rely on static embedding similarity. While this works with simple, category-level queries, it struggles to capture richer linguistic reasoning, such as compositional instructions. For example, given the query "What color are pillows in the kitchen?", such methods may retrieve all pillows in the scene, rather than those specifically associated with the kitchen. This behavior arises because their visual reasoning operates at the level of individual Gaussians, without explicit mechanisms for iterative grounding or contextual aggregation.

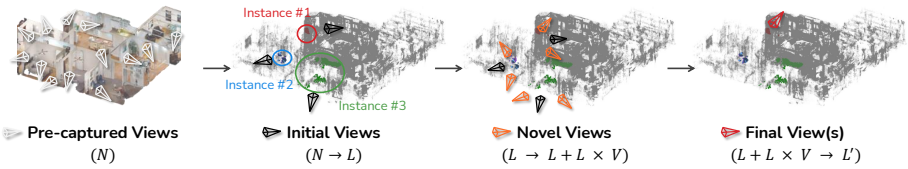

Fig. 2: Overview. Given N pre-captured views, we sample L' images to perform embodied tasks. First, we select L initial views with N initial view selection ($N \gg L$). Then, we augment each initial view with V novel views. Finally, we proceed with the final view selection with VLMs to find most informative views to answer the query.

More recently, ReferSplat [17] aligns fine-grained textual descriptions with Gaussian features, leading to a deeper understanding of spatial relationships. However, because its feature updates are rendering-based, it remains less suitable for direct 3D search [23]. By coupling VLM reasoning [3, 21, 32, 33, 39, 49] with a modern 3D representation, specifically 3DGS [24], our method enables direct search and grounding in 3D space. Our approach also allows agents to actively synthesize novel viewpoints, disambiguate queries, and ground targets with high spatial accuracy through advanced VLM reasoning.

Embodied Reasoning with 3D VLMs. Recent advances in embodied reasoning have introduced 3D VLMs [18–20, 55] that process 3D representations directly [2, 6, 34, 64]. Conventional approaches rely on point cloud-based encoders [19, 20, 55] that require extensive task-specific training and specialized architectures. This reliance introduces inconsistencies when making direct comparisons against our input modality. While recent studies have integrated 3DGS into VLMs [16], these methods still require additional fine-tuning. In contrast, our approach enables 2D pre-trained models to perceive and reason in 3D environments without further training by leveraging rich 2D VLMs.

3 Method

Given a user query $\mathcal{Q}$ about a 3D scene, our goal is to retrieve and synthesize the camera viewpoints that provide the most relevant visual evidence for answering the query and grounding the referred entities in 3D. The scene is observed through N posed RGB images $\mathcal{I} = \{I_i\}_{i=1}^{N}$ with camera poses $\mathcal{H} = \{H_i\}_{i=1}^{N}$, where each pose $H_i = [\mathbf{R}_i|\mathbf{t}_i] \in \mathbb{R}^{3\times4}$ denotes the extrinsic matrix composed of rotation $\mathbf{R}_i$ and translation $\mathbf{t}_i$.

An overview of our framework is shown in Fig. 2. To balance efficiency and evidence quality, we decompose this process into initial view selection (Sec. 3.1), which retrieves visible query-relevant regions from N input views to L selected views ($N \gg L$). Then, the system proceeds with the final view selection (Sec. 3.2) which refines these observations through synthesized views when the original images are insufficient. The selected final views are utilized for downstream tasks (Sec. 3.4), such as embodied question answering and 3D visual grounding.

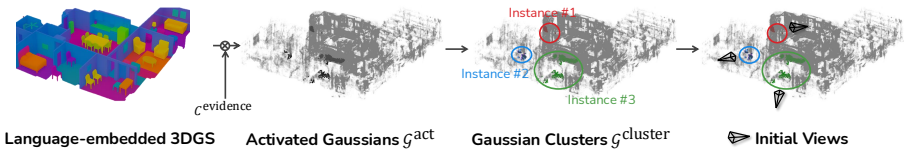

Fig. 3: Initial view selection. First, we identify the activated Gaussians $\mathcal{G}^{\text{act}}$ using a set of evidence categories $\mathcal{C}^{\text{evidence}}$ extracted from the query. Next, these activated Gaussians are grouped into clusters $\{\mathcal{G}_l^{\text{cluster}}\}_{l=1}^{L}$ that serve as instance-level candidates. Finally, the initial viewpoints are determined by maximizing the visibility score.

3.1 Initial View Selection in Language-embedded 3D Gaussians

We first localize the query-relevant region in 3D and then search for viewpoints that maximize its visibility as shwon in Fig. 3. By measuring visibility in 3D rather than in 2D pixel space, our approach avoids biases from occlusion and camera distance, enabling volume-aware viewpoint selection.

Construction of language-embedded 3D Gaussians. We first reconstruct the scene as a set of M 3D Gaussians,

$$\mathcal{G} = \{G_j\}_{j=1}^{M}, \tag{1}$$

where each Gaussian G_j contains its geometric and appearance parameters from 3DGS, such as position, scale, opacity, and color. To use this representation as a queryable memory, we additionally attach semantic information to each Gaussian. Depending on the downstream task and the comparison protocol, this semantic attribute is represented in one of two forms.

For embodied reasoning, we follow 3D-Mem [57] and use a closed-set object detector [51] to assign each Gaussian a category label $C_j \in \mathcal{C}$, where $\mathcal{C}$ denotes the detector vocabulary. This category-level memory is suitable for retrieving common evidence objects, such as *pillow* or *cushion*, from natural-language questions. For 3D object localization and 3D referring segmentation, we adopt the open-set pipeline of Dr. Splat [23]. Specifically, we segment the input images with SAM [26], extract a CLIP [42] feature for each segment, and register these features to the reconstructed Gaussians. The resulting Gaussian G_j is associated with an embedding f_j that can be compared with text embeddings. In both settings, we use the direct registration strategy from [23] to aggregate multi-view semantic observations into the 3DGS representation, yielding a compact structured memory that can be searched directly in 3D space.

Initial view selection from evidence. The initial view selection stage converts the user query into a small set of 3D candidate regions and chooses one pre-recorded view for each region. This stage reduces the number of images passed to VLMs while preserving the visual evidence that is likely to answer $\mathcal{Q}$. It consists of three steps: extracting evidence categories from the query, clustering the activated Gaussians into instance-level candidates, and selecting the most visible input view for each cluster.

Gaussian search from query. Natural-language questions often contain reasoning intents that are not directly usable as 3D search keys. We therefore first use a Large Language Model (LLM) to extract a set of evidence categories $\mathcal{C}^{\text{evidence}} \subseteq \mathcal{C}$ from the query. These categories are objects or semantic concepts that are likely to provide evidence for answering the query. For example:

> **Prompt:** You should retrieve helpful objects in order.
> **Query:** Where can I take a nap?
> **Categories ($\mathcal{C}$):** radiator, cushion, sink, pillow, picture, ...
> **Evidence categories ($\mathcal{C}^{\text{evidence}}$):** pillow, cushion.

For embodied reasoning, each Gaussian has a closed-set category label C_j. We therefore activate the Gaussians whose labels match the selected evidence categories:

$$\mathcal{G}^{\text{act}} = \{G_j \mid C_j \in \mathcal{C}^{\text{evidence}}\}, \tag{2}$$

where $\mathcal{G}^{\text{act}}$ denotes the set of query-activated Gaussians.

For 3D grounding, where each Gaussian is represented by an open-set feature f_j, activation is based on text-image feature similarity. Let t_c be the text embedding of category $c \in \mathcal{C}^{\text{selected}}$, and let $\text{sim}(\cdot,\cdot)$ denote the similarity function, implemented as cosine similarity in our experiments. We activate Gaussians whose accumulated similarity to the selected evidence categories exceeds a threshold τ:

$$\mathcal{G}^{\text{act}} = \{G_j \mid \text{sim}(f_j, t_c) \geq \tau\}. \tag{3}$$

We set $\tau = 0.5$ for 3D grounding.

Clustering 3D Gaussians for instance assignment. A single Gaussian is a local primitive and does not by itself represent an entire object. We therefore group activated Gaussians into spatial clusters that serve as instance-level candidates. Let $\mathbf{x}_j \in \mathbb{R}^3$ denote the 3D center of Gaussian G_j. We cluster $\mathcal{G}^{\text{act}}$ according to the spatial distribution of $\{\mathbf{x}_j\}$ using HDBSCAN [11, 36]:

$$\{\mathcal{G}_l^{\text{cluster}}\}_{l=1}^{L} = \text{Cluster}(\mathcal{G}^{\text{act}}), \tag{4}$$

where $\mathcal{G}_l^{\text{cluster}}$ is the l-th Gaussian cluster and L is the number of clusters adaptively determined by the algorithm. When category labels are available, clustering is applied within the same semantic category to avoid grouping different object types. Since clustering is used to reduce the number of candidate viewpoints for VLM processing, coarse grouping is sufficient even if nearby parts of the same semantic region are merged.

Initial view selection with visibility score. Given L Gaussian clusters $\{\mathcal{G}_l^{\text{cluster}}\}_{l=1}^{L}$, we select one initial input pose for each cluster from $\mathcal{H}$. The selected initial pose set is denoted as

$$\mathcal{H}^{\text{init}} = \{H_l^{\text{init}}\}_{l=1}^{L}, \tag{5}$$

where H_l^{init} is the input view chosen for cluster $\mathcal{G}_l^{\text{cluster}}$. For each pair of cluster $\mathcal{G}_l^{\text{cluster}}$ and input pose H_i, we render a cluster activation map using the splatting

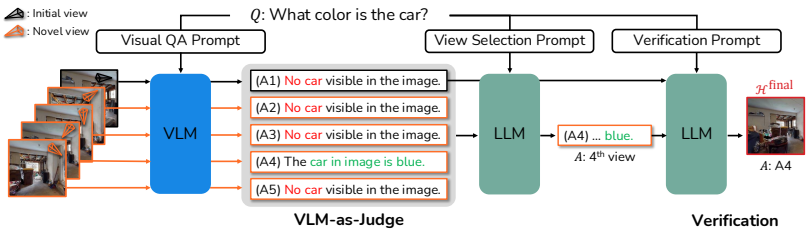

Fig. 4: Final view selection. For each initial view, we synthesize novel views and employ a VLM-as-Judge framework to select the most informative viewpoints from both the initial and novel view pools. To ensure that the synthesized views provide richer information compared to the initial ones, we conduct a verification stage.

renderer [24, 66]. Let $M_{l,i}^{\text{visible}}$ be the set of pixels with positive activation in this rendered map.

To account for occlusion, we do not simply count projected Gaussians. Instead, for every pixel $\mathbf{u} \in M_{l,i}^{\text{visible}}$, we identify the Gaussian in the full scene that contributes the maximum rendering weight:

$$\mathcal{G}_{l,i}^{\max} = \left\{ G_{j^*} \;\middle|\; \mathbf{u} \in M_{l,i}^{\text{visible}},\ j^* = \underset{m \in \{1,\dots,M\}}{\arg\max}\ w_m(H_i, \mathbf{u}; G_m) \right\}. \tag{6}$$

Here, $w_m(H_i, \mathbf{u}; G_m)$ is the rendering weight of Gaussian G_m at pixel $\mathbf{u}$ under camera pose H_i. If a target cluster is occluded, the dominant Gaussian at the corresponding pixel is likely to come from a foreground object rather than from the target cluster. The visible subset of cluster $\mathcal{G}_l^{\text{cluster}}$ is therefore defined as

$$\mathcal{G}_{l,i}^{\text{vis}} = \mathcal{G}_{l,i}^{\max} \cap \mathcal{G}_l^{\text{cluster}}. \tag{7}$$

We then define the visibility score S^{vis} S^{vis} as the fraction of cluster Gaussians that are visible from H_i:

$$S^{\text{vis}} S_{l,i}^{\text{vis}} = \frac{|\mathcal{G}_{l,i}^{\text{vis}}|}{|\mathcal{G}_l^{\text{cluster}}|}. \tag{8}$$

Finally, the initial pose for cluster l is selected by

$$H_l^{\text{init}} = H_{i_l^*}, \quad i_l^* = \underset{i \in \{1,\dots,N\}}{\arg\max}\ S^{\text{vis}} S_{l,i}^{\text{vis}}. \tag{9}$$

The visibility-based selection favors views where the queried instance is not merely projected into the image but is actually visible after considering occluding Gaussians.

3.2 Novel View Synthesis

Unlike previous study [57], which are constrained to pre-recorded images for visual reasoning, our model integrates novel-view synthesis directly into the reasoning

pipeline. This pipeline is effective when the initial camera views are suboptimal due to scene ambiguities such as occlusions or complex object relationships. We therefore perturb the initial viewpoints via novel-view rendering. We generate perturbed initial views with novel view synthesis and employ a VLM-as-Judge [3, 21] module to assess their informativeness, ultimately selecting more suitable viewpoints for the given user queries. As illustrated in Fig. 4, VLMs serve as judges for both novel-view adjustments and verification.

Given the initial views $\mathcal{H}^{\text{init}} = \{H_l^{\text{init}}\}_{l=1}^{L}$, we synthesize V novel views around each H_l^{init}, denoted as

$$\mathcal{H}_l^{\text{novel}} = \{\widetilde{H}_{l,v}\}_{v=1}^{V}, \quad l = 1, \ldots, L,$$

$$\mathcal{H}^{\text{novel}} = \bigcup_{l=1}^{L} \mathcal{H}_l^{\text{novel}} = \{\widetilde{H}_{l,v} \mid l = 1, \ldots, L,\ v = 1, \ldots, V\},$$

where $V = 4$, corresponding to the left, right, forward, and backward perturbations. Thus, the total number of synthesized novel views is $L \times V$. We define its final view candidate set by combining the original initial view and its synthesized novel views: $\mathcal{H}^{\text{candi}} = \mathcal{H}^{\text{init}} \cup \mathcal{H}^{\text{novel}}$.

3.3 Final View Selection

We employ VLMs for the final 3D view selection because visibility scores alone are insufficient to measure the informativeness of novel views.

VLM-as-Judge for estimating view informativeness. For the VLM-as-Judge stage, we first generate candidate final views $\mathcal{H}^{\text{candi}}$. Each candidate view is then evaluated independently by feeding its rendered image and the query to VLMs [3, 21], because VLMs tend to struggle with long visual sequences [13, 28, 40, 47, 52]. Utilizing text for robust aggregation is a well-established strategy to mitigate multimodal integration biases [27, 28, 40, 61]. Following this, the VLM distills recovered missing or occluded evidence from each synthesized view into per-view text candidates, and the LLM selects the best answer among these predictions [44], implicitly identifying the most effective final view. Our results also empirically demonstrate the efficacy of relying on LLMs for this selection as shwon in Table 5. The cardinality of the final view set $\mathcal{H}^{\text{final}} = \{H_f\}_{f=1}^{L'}$ varies depending on the target downstream task. For the embodied question-answering task, we set $L' = L$, where each initial view is independently refined by selecting the single best perspective from its synthesized novel views. For single target object localization (e.g., ScanRefer [6]) and 3D referring segmentation, $L' = 1$. Lastly, for multiple target object localization tasks such as Multi3DRefer [64], L' varies dynamically based on the user query.

Verification. After the final view selection stage, we perform verification [31] by comparing the final views $\mathcal{H}^{\text{final}} = \{H_f\}_{f=1}^{L'}$ with their corresponding L' initial views $\{H_f^{\text{init}}\}_{f=1}^{L'}$ ($L' \leq L$). We prioritize final views, especially when the initial views fail to provide a definitive answer. Recent advancements in VLMs have led

to behaviors in which models state responses, such as "I don't know" [63], when the visual cues are insufficient. Our pipeline mitigates this issue by identifying and selecting the specific view that provides the necessary visual cues, thereby improving overall performance.

3.4 Downstream Tasks

Episodic embodied question-answering (EM-EQA). For EM-EQA, we render the final poses $\mathcal{H}^{\text{final}}$ into RGB images and feed these images with the query $\mathcal{Q}$ to a VLM [39]. Compared with methods that rely only on pre-recorded views [14, 35, 57], our novel view adjustment can provide visual evidence from viewpoints that were not available in the original episodic memory. This enables the VLM to answer from more query-relevant observations while keeping the number of input views compact.

3D visual grounding. For 3D visual grounding, the selected final poses provide spatial constraints for localizing target entities in the Gaussian scene. Let $\mathcal{F}(H_l^{\text{final}})$ denote the viewing frustum of the l-th final pose. We first restrict the activated Gaussians to those lying inside the union of final-view frustums:

$$\mathcal{G}^{\text{frustum}} = \left\{ G_j \in \mathcal{G}^{\text{act}} \;\middle|\; \mathbf{x}_j \in \bigcup_{l=1}^{L} \mathcal{F}(H_l^{\text{final}}) \right\} \tag{10}$$

where $\mathbf{x}_j \in \mathbb{R}^3$ denote the 3D center of Gaussian G_j. The final grounded region is then selected from $\mathcal{G}^{\text{frustum}}$ using the VLM reasoning output associated with user query $\mathcal{Q}$. In this way, SplatReasoner combines direct 3D Gaussian activation with query-driven view selection, enabling fine-grained localization for both 3D object localization and 3D referring segmentation.

4 Experiments

SplatReasoner integrates 3D Gaussian Splatting (3DGS) and Vision-Language Models (VLMs), enabling selection of final viewpoints that directly correspond to the user's query. The final viewpoints highlight the specific regions relevant to the user query, enabling embodied behavior such as embodied question-answering in Sec. 4.1 or 3D grounding including 3D object localization (single or multiple targets with declarative query) and 3D referring segmentation (question query) in Sec. 4.2. The necessity of our pipeline is shown by ablation studies in Sec. 4.3.

Dataset. We evaluate SplatReasoner on two datasets that assess its capabilities in embodied reasoning and 3D scene understanding. To measure embodied reasoning capability, we use the evaluation dataset released by OpenEQA [35]. Specifically, we focus on the Episodic-Memory Embodied Question Answering (EM-EQA) task, which reflects practical scenarios, *e.g.*, smart glasses that rely on the history of past observations to assist users. OpenEQA question-answer pairs are constructed from Habitat-Matterport 3D [43] and ScanNet [8, 45], covering seven types of questions with over 1,600 queries in about 180 environments.

Table 1: Evaluation on EM-EQA. We measure the semantic equivalent using LLM-Match and frame efficiency using the average number of frames. Methods marked with * and Human score are reported from OpenEQA, while all other results are reproduced. We report OpenEQA [35] results since the code is not publicly available. Average frames denote the number of final views are used to answer queries. † indicates that while novel views are utilized for the view adjustment, the actual number of input frames fed into the VLMs remains consistent with the reported average frames.

Method	LLM-Match (↑)		Average Frames
	Closed Model [21]	Open Model [3]	
BlindLLM [38]	34.8	29.4	0
Frame Captions	24.1	31.7	0
CG Captions* [14]	36.5	-	0
SVM Captions* [35]	38.9	-	0
Multi-Frame	49.1	48.2	3.0
3D-Mem [57]	54.6	50.8	2.7
SplatReasoner (ours)	**57.8**	**51.6**	**2.6**†
Human	86.8		Full

Table 2: 3D grounding results. We evaluate recent studies, such as Dr. Splat, using both category-level and sentence-based queries. ReferSplat is tested with sentence-based queries, as it is specifically designed to interpret detailed textual descriptions. Our method follows the pipeline of SplatReasoner. We evaluate our method on 3D object localization using ScanRefer and Multi3DRefer, as well as on 3D referring segmentation.

Method	ScanRefer [6]			Multi3DRefer [64]			3D Referring Segmentation		
	3D mIoU ↑	Acc@8 ↑	Acc@10 ↑	3D mIoU ↑	Acc@8 ↑	Acc@10 ↑	3D mIoU ↑	Acc@8 ↑	Acc@10 ↑
Dr. Splat (category) [23]	8.73	34.53	28.53	4.69	18.33	13.67	10.03	41.71	33.86
Dr. Splat (sentence) [23]	9.44	34.28	29.35	4.14	15.40	12.97	10.56	**45.21**	43.21
ReferSplat [17]	3.14	10.68	7.32	1.25	2.08	1.30	2.34	2.04	0.00
SplatReasoner (Ours)	**11.12**	**35.84**	**32.75**	**6.32**	**29.59**	**24.51**	**12.46**	45.14	**45.14**

In addition, we also evaluate 3D grounding with complex linguistic user queries, such as 3D object localization [6,64] and 3D referring segmentation [8,35]. In 3D object localization, we evaluate our model under two configurations: a single-target setting [6], where each query corresponds to exactly one object, and a multi-target setting [64], where a query can refer to multiple objects. Since existing benchmarks predominantly use point clouds, we curate a 3DGS-based evaluation. For the 3D object localization task, we utilize ScanRefer [6] validation set (141 scenes & 9,508 queries), *e.g.*, this is the bed with blue sheets near the desk and Multi3DRefer [64] multi-target validation set (133 scenes & 2,757 queries), *e.g.*, this is a black chair facing the door. To address more diverse formats of user queries such as questions, we manually generate 49 spatial queries (*e.g.*, "Which bed is closest to the window?") that cover diverse attribute-based and spatial relationships. Each scene contains multiple instances from the same object category, requiring the model to distinguish between them using precise spatial cues and attributes.

4.1 Embodied Question Answering (EQA)

Comparisons. We compare our method with six types of agents. BlindLLM [38, 56] infers their answer without visual input and only with the user question. Frame Captions uses LLaVA-v1.5 [32] to caption 50 sampled frames, feeding captions and the question to LLM [38, 56]. CG Captions and SVM Captions construct textual scene graphs from episodic memory using ConceptGraphs [14] and Sparse Voxel Map [35], respectively. Multi-Frame directly provides three frames through unified sampling along with the question. 3D-Mem [57] leverages object-centric multi-view Memory Snapshots to preserve spatial and semantic consistency while reducing redundant frames. Ours utilizes semantic 3DGS as an episodic memory integrated with VLM [3, 21] interaction, enabling a compact, informative, and effective representation.

We reproduce all results except for CG and SVM Captions, for which the code and prompts are not publicly available. For the 3D-Mem evaluation, we follow the extracted snapshots provided by the 3D-Mem authors, which contain a small omission (about 1%), and therefore evaluate 1,623 questions for all reproductions. As 3D-Mem constructs its memory based on closed-set categories, we also use closed-set semantic categories in our method for a fair comparison.

Settings. We follow the EM-EQA setting proposed in OpenEQA [35], where the agent answers a question solely from past visual observations without further exploration. Each agent receives episodic memories or sampled frames as input and generates an open-ended textual answer via its VLMs. Performance is evaluated using the LLM-Match metric [35] which measures semantic agreement between predicted and ground-truth answers by prompting a strong language model [38] to judge whether the two answers are semantically equivalent. This metric better captures conceptual correctness than exact string matching, providing a more reliable assessment of reasoning and grounding performance in embodied settings. We test our method with both of a closed-source model (GPT-4o) [21] and an open-source model (Qwen3-VL-8B) [3] for the reproducibility and accessibility.

Results. Table 1 reports the LLM-Match performance and average frames across different baselines. BlindLLM and Captions achieve relatively low scores indicating that relying solely on textual information is insufficient for embodied reasoning. Incorporating visual signals significantly boosts performance: Multi-Frame results highlight the importance of visual context for understanding and reasoning. 3D-Mem further improves performance through its compact yet informative Snapshot Memory architecture. Finally, SplatReasoner achieves the highest performance while maintaining a similar average number of frames, validating the effectiveness and efficiency of our semantic 3DGS representation and novel-view adjustment approach. For the open-source model, we employ the Qwen3-VL-8B model. Although it shows a relatively constrained performance gap due to the smaller model capability compared to closed-source models, we still observe consistent improvements. We expect this performance gap to widen significantly with more capable, larger-scale models.

Fig. 5: Qualitative results of the 3D object localization (ScanRefer, single target). Compared to the competing methods, ours more accurately identifies the fine-grained target locations referenced by the sentence query.

Fig. 6: Qualitative results of the 3D object localization (Multi3DRefer, multi targets). Compared to competing methods, ours achieves fine-grained localization even when a query references multiple target locations.

Fig. 7: Qualitative results of the 3D referring segmentation. Our model can process queries in question formats.

4.2 3D Grounding

Competing methods. We compare our method against recent language-guided 3D Gaussian Splatting approaches: Dr. Splat [23] and ReferSplat [17]. Dr. Splat lifts multi-view CLIP [42] image embeddings into the 3D Gaussian space for language-based querying. We optimize the model following the original paper's settings. ReferSplat [17] optimizes 3D Gaussians using fine-grained textual expressions for referring understanding. Because ReferSplat requires paired textual annotations for supervision but lacks an official automated data generation

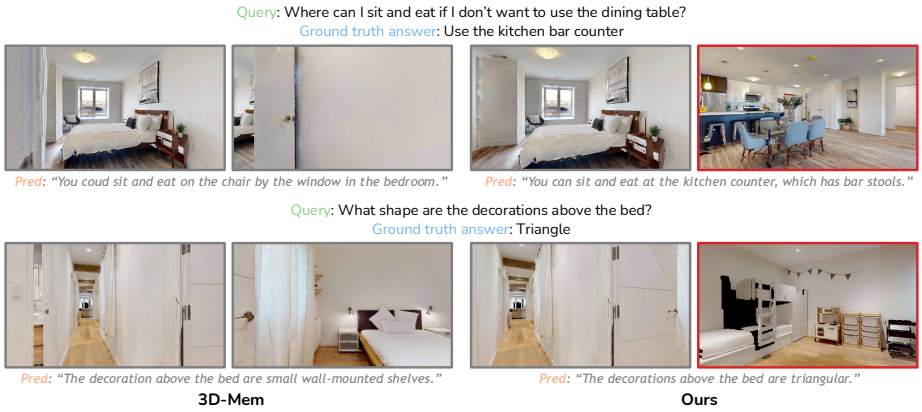

Fig. 8: Initial view selection. Compared to 3D-Mem, our visibility-based policy can select more informative views by accounting for the actual visibility of target objects, rather than relying solely on the existence of categories within a given view.

Fig. 9: Final view selection. We apply novel view synthesis to refine initial views, leveraging 3DGS capability. Our results show that even when the initial views lack sufficient information to answer the question, the novel-view adjustment process can recover the necessary viewpoints, enabling more accurate question-aligned answers.

pipeline, we manually constructed the necessary training data by using foundation models [26, 60] to generate textual descriptions for the object masks.

Settings. We measure the 3D mean Intersection over Union (3D mIoU) between the activated Gaussians and the ground-truth Gaussians. Since the scene is represented by Gaussians, we compute the mIoU based on the volumetric overlap induced by the Gaussian scales and opacities, following [23].

For the 3D object localization, we directly leverage ScanRefer's segmentation ground truths into the 3DGS. Since no fine-grained grounding benchmarks matching our specific question queries exist for 3D referring segmentation, we manually annotate target instances to construct a paired dataset.

The ground-truth 3D Gaussian labels are distilled by computing the Mahalanobis distance between the ground-truth point clouds from the dateset [8] and 3D Gaussians, following the procedure in [23]. Given a user query (sentence or question), each model identifies the corresponding activations over all 3D Gaussians in the scene. A higher mIoU indicates better grounding accuracy, meaning that the activated Gaussians more precisely align with the target region

referenced by the user query. For details of our evaluation dataset construction, please refer to the supplementary material.

Results. Table 2 reports 3D mIoU and Acc@k scores. Acc@k denotes the percentage of predictions whose IoU with the ground truth exceeds k%. We normalize the computed Gaussian volumes by the 90% value and clip their range between 0 and 1, which helps suppress excessive floating Gaussians that appear in vanilla 3DGS [12]. While ReferSplat is designed to interpret detailed textual descriptions, its rendering-based optimization is poorly aligned with direct 3D search, resulting in lower mIoU. In contrast, SplatReasoner more reliably identifies the question-related 3D regions thanks to the combination of directly registered semantic 3DGS and VLM-based reasoning.

In Figs. 5, 6, and 7, we observe that our method aligns more closely with the ground truth, demonstrating stronger fine-grained grounding capability. This capability is particularly important for embodied exploration and navigation tasks, where the system must precisely reflect user queries. Dr. Splat is designed to find objects without distinguishing instances, which limits its ability to localize fine-grained target regions (category) and, consequently, makes it struggle to directly interpret complex language queries (sentence). ReferSplat is designed to respond to handle complex language queries, but is rendering-based optimization is not compatible with direct 3D search, leading to limited 3D localization.

4.3 Ablation Study

We use 184 embodied question answering (EQA) questions for initial design choices, and evaluate verification stage on the full set of 1,623 questions. All ablation studies are conducted using closed-source models [21, 39, 49].

Score function for initial view. We compare two score functions for selecting initial views: a volume score, computed by replacing the Gaussian count with

$$\text{Volume}(G_j) = s_j^x s_j^y s_j^z \alpha_j, \qquad (11)$$

where s denotes the scale values of each axis and α indicates the opacity of each Gaussian. The volume-based formulation performs worse because large outlier Gaussians can disproportionately dominate the score, leading to strong bias and unreliable view selection. In contrast, the visibility score effectively acts as a proxy for object surface coverage, independent of Gaussian scale or training dynamics. This yields more stable visibility estimates and produces noticeably better LLM-Match performance, as shown in Table 3.

Table 3: Ablation on initial view selection strategy (184 questions). Because 3DGS is volumetric, we also compute volumetric visibility scores in addition to the non-volumetric version.

Method	LLM-Match (↑)	Average Frames
3D-Mem [57]	45.4	3.1
Volume score	47.8	2.7
Visibility score (ours)	**48.2**	2.7

View selection pipeline. As shown in Table 4, the initial view selection alone already improves performance over 3D-Mem, demonstrating the effectiveness of our visibility score. Figure 8 presents snapshots of 3D-Mem and our initial view selection. We observe that the visibility scores successfully

guide the selection of more informative views. With the final view selection, integrated with novel views and the verification stage, performance further increases, indicating that 3DGS helps uncover more informative viewpoints. As shown in Fig. 9, final views better capture evidence relevant to the question, which in turn supports VLMs in generating more accurate answers. † indicates that while novel views are utilized during the view adjustment stage, the actual number of input frames fed into the VLMs remains consistent with the reported average frames.

Table 4: Ablation study on final view selection (184 questions). We demonstrate the necessity of each component in our method.

Method	LLM-Match (↑)	Average Frames
3D-Mem [57]	45.4	3.1
Initial view selection	48.2	2.7
Final view selection (ours)	**50.5**	2.7†

VLM-as-Judge design. For the final view selection, we perform a VLM-as-Judge process for novel-view adjustment. To evaluate the informativeness of each candidate view, we rely solely on text-based descriptions and feed them into LLMs [44]. As demonstrated in Table 5, our design choice empirically outperforms VLMs that incorporate visual information, while offering a more computationally efficient and lightweight alternative.

Table 5: Ablation study on VLM-as-Judge design (184 questions).

Method	LLM-Match (↑)	Average Frames
3D-Mem [57]	45.4	3.1
Image	47.1	2.7†
Image + Text	46.9	2.7†
Text (ours)	**50.5**	2.7†

Verification process. In the verification stage, we compare initial views with final candidate views generated through novel-view adjustment. A final viewpoint is selected only if it contains previously missing information. As shown in Table 6, our approach achieves improved performance compared to the baseline without the verification step.

Table 6: Ablation study on verification process (1,623 questions).

Method	LLM-Match (↑)	Average Frames
Final view	54.5	2.6†
Final view + verification (ours)	**57.8**	2.6†

5 Conclusion

We introduce SplatReasoner, a framework utilizing 3D Gaussians for compact episodic memories to enable VLM reasoning and 3D grounding. By integrating semantic 3DGS, visibility-based view selection, and novel view synthesis, SplatReasoner retrieves and refines query-relevant perspectives, overcoming the occlusions and detail loss typical of fixed-view systems. While currently designed for passive episodic-memory scenarios rather than active robotic navigation, extending SplatReasoner to active, long-horizon exploration with sequential decision-making remains a promising future avenue.

Acknowledgements. This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant (No. RS-2020-II200004, Development of Previsional Intelligence based on Long-term Visual Memory Network (25%); No. RS-2026-25518317, Development of AI memory mechanism that reflects human cognitive principles (25%); No.RS-2019-II191906, Artificial Intelligence Graduate School Program(POSTECH) (25%)), the National Research Foundation of Korea(NRF) grant (No. RS-2024-00453301 (25%)), and the InnoCORE program (N10250156) funded by the Korea government (MSIT).

References

1. An, H., Jung, J., Kim, M., Kim, C., Jeon, M., Han, J., Fukuda, K., Narihira, T., Ko, H., Kim, J., et al.: C3g: Learning compact 3d representations with 2k gaussians. In: CVPR (2026)
2. Azuma, D., Miyanishi, T., Kurita, S., Kawanabe, M.: Scanqa: 3d question answering for spatial scene understanding. In: CVPR (2022)
3. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. In: arXiv preprint arXiv:2511.21631 (2025)
4. Cangea, C., Belilovsky, E., Liò, P., Courville, A.: Videonavqa: Bridging the gap between visual and embodied question answering. In: BMVC (2019)
5. Changpinyo, S., Kukliansy, D., Szpektor, I., Chen, X., Ding, N., Soricut, R.: All you may need for VQA are image captions. In: North American Chapter of the Association for Computational Linguistics (2022)
6. Chen, D.Z., Chang, A.X., Nießner, M.: Scanrefer: 3d object localization in rgb-d scans using natural language. In: ECCV (2020)
7. Cheng, A.C., Yin, H., Fu, Y., Guo, Q., Yang, R., Kautz, J., Wang, X., Liu, S.: Spatialrgpt: Grounded spatial reasoning in vision-language models. Advances in Neural Information Processing Systems **37**, 135062–135093 (2024)
8. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: CVPR. pp. 5828–5839 (2017)
9. Das, A., Datta, S., Gkioxari, G., Lee, S., Parikh, D., Batra, D.: Embodied question answering. In: CVPR (2018)
10. Ding, J., Liu, X., Pan, Z., Long, S., Li, G.: Extrinsplat: Decoupling geometry and semantics for open-vocabulary understanding in 3d gaussian splatting. In: CVPR (2026)
11. Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: International Conference on Knowledge Discovery and Data Mining (KDD) (1996)
12. Fan, Z., Wang, K., Wen, K., Zhu, Z., Xu, D., Wang, Z., et al.: Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps. In: NeurIPS (2024)
13. Ge, J., Chen, Z., Lin, J., Zhu, J., Liu, X., Dai, J., Zhu, X.: V2pe: Improving multimodal long-context capability of vision-language models with variable visual position encoding. In: ICCV (2025)
14. Gu, Q., Kuwajerwala, A., Morin, S., Jatavallabhula, K.M., Sen, B., Agarwal, A., Rivera, C., Paul, W., Ellis, K., Chellappa, R., et al.: Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In: IEEE International Conference on Robotics and Automation (2024)
15. Guo, Q., De Mello, S., Yin, H., Byeon, W., Cheung, K.C., Yu, Y., Luo, P., Liu, S.: Regiongpt: Towards region understanding vision language model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13796–13806 (2024)
16. Halacheva, A.M., Zaech, J.N., Wang, X., Paudel, D.P., Van Gool, L.: Gaussianvlm: Scene-centric 3d vision-language models using language-aligned gaussian splats for embodied reasoning and beyond. IEEE Robotics and Automation Letters (2025)
17. He, S., Jie, G., Wang, C., Zhou, Y., Hu, S., Li, G., Ding, H.: Refersplat: Referring segmentation in 3d gaussian splatting. In: ICML (2025)

18. Hong, Y., Zhen, H., Chen, P., Zheng, S., Du, Y., Chen, Z., Gan, C.: 3d-llm: Injecting the 3d world into large language models. In: NeurIPS (2023)
19. Huang, H., Chen, Y., Wang, Z., Huang, R., Xu, R., Wang, T., Liu, L., Cheng, X., Zhao, Y., Pang, J., et al.: Chat-scene: Bridging 3d scene and large language models with object identifiers. In: NeurIPS (2024)
20. Huang, J., Yong, S., Ma, X., Linghu, X., Li, P., Wang, Y., Li, Q., Zhu, S.C., Jia, B., Huang, S.: An embodied generalist agent in 3d world. In: ICML (2024)
21. Hurst, A., Lerer, A., Goucher, A.P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al.: Gpt-4o system card. In: arXiv preprint arXiv:2410.21276 (2024)
22. Ji, Y., Zhu, H., Tang, J., Liu, W., Zhang, Z., Tan, X., Xie, Y.: Fastlgs: Speeding up language embedded gaussians with feature grid mapping. In: AAAI (2025)
23. 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: CVPR (2025)
24. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM TOG (2023)
25. Kim, N., Kwon, O., Yoo, H., Choi, Y., Park, J., Oh, S.: Topological semantic graph memory for image-goal navigation. In: Annual Conference on Robot Learning (2022)
26. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: ICCV (2023)
27. Lee, M.J., Gong, D., Cho, M.: Video summarization with large language models. In: CVPR (2025)
28. Lee, M., Park, Y., Hwang, D., Kim, Y., Oh, S.J., Choe, J.: Enhancing multi-image understanding through delimiter token scaling. In: ICLR (2026)
29. Lee, S., Wang, Z., Wang, Y., Lee, G.H.: Embodiedsplat: Online feed-forward semantic 3dgs for open-vocabulary 3d scene understanding. In: CVPR (2026)
30. Lei, X., Wang, M., Zhou, W., Li, H.: Gaussnav: Gaussian splatting for visual navigation. IEEE Transactions on Pattern Analysis and Machine Intelligence **47**(5), 4108–4121 (2025)
31. Liao, Y.H., Mahmood, R., Fidler, S., Acuna, D.: Can large vision-language models correct semantic grounding errors by themselves? In: CVPR (2025)
32. Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning. In: CVPR (2024)
33. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: NeurIPS (2023)
34. Ma, X., Yong, S., Zheng, Z., Li, Q., Liang, Y., Zhu, S.C., Huang, S.: Sqa3d: Situated question answering in 3d scenes. In: ICLR (2023)
35. Majumdar, A., Ajay, A., Zhang, X., Putta, P., Yenamandra, S., Henaff, M., Silwal, S., Mcvay, P., Maksymets, O., Arnaud, S., et al.: Openeqa: Embodied question answering in the era of foundation models. In: CVPR (2024)
36. Malzer, C., Baum, M.: A hybrid approach to hierarchical density-based cluster selection. In: 2020 IEEE International Conference on multisensor fusion and integration for intelligent systems (MFI). IEEE (2020)
37. Marrie, J., Ménégaux, R., Arbel, M., Larlus, D., Mairal, J.: Ludvig: Learning-free uplifting of 2d visual features to gaussian splatting scenes. In: ICCV (2025)
38. OpenAI: Gpt-4 technical report. `https://arxiv.org/abs/2303.08774` (2023)
39. OpenAI: Gpt-4v(ision) system card. `https://openai.com/research/gpt-4v-system-card` (2023)
40. Park, Y., Lee, M., Chun, S., Choe, J.: Mitigating cross-image information leakage in lvlms for multi-image tasks. arXiv preprint arXiv:2508.13744 (2025)

41. 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)
42. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: ICML (2021)
43. Ramakrishnan, S.K., Gokaslan, A., Wijmans, E., Maksymets, O., Clegg, A., Turner, J.M., Undersander, E., Galuba, W., Westbury, A., Chang, A.X., et al.: Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. In: Conference on Neural Information Processing Systems Datasets and Benchmarks Track (2021)
44. Ranasinghe, K., Li, X., Kahatapitiya, K., Ryoo, M.S.: Understanding long videos with multimodal language models. In: ICLR (2025)
45. Rozenberszki, D., Litany, O., Dai, A.: Language-grounded indoor 3d semantic segmentation in the wild. In: ECCV (2022)
46. Saxena, S., Buchanan, B., Paxton, C., Liu, P., Chen, B., Vaskevicius, N., Palmieri, L., Francis, J., Kroemer, O.: Grapheqa: Using 3d semantic scene graphs for real-time embodied question answering. In: Annual Conference on Robot Learning (2025)
47. Sharma, A., Saxon, M., Wang, W.Y.: Losing visual needles in image haystacks: Vision language models are easily distracted in short and long contexts. In: Findings of the Association for Computational Linguistics: EMNLP 2024 (2024)
48. Shi, J.C., Wang, M., Duan, H.B., Guan, S.H.: Language embedded 3d gaussians for open-vocabulary scene understanding. In: CVPR (2024)
49. Singh, A., Fry, A., Perelman, A., Tart, A., Ganesh, A., El-Kishky, A., McLaughlin, A., Low, A., Ostrow, A., Ananthram, A., et al.: Openai gpt-5 system card. In: arXiv preprint arXiv:2601.03267 (2025)
50. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al.: Llama 2: Open foundation and fine-tuned chat models. In: arXiv preprint arXiv:2307.09288 (2023)
51. Varghese, R., Sambath, M.: Yolov8: A novel object detection algorithm with enhanced performance and robustness. In: 2024 International conference on advances in data engineering and intelligent computing systems (ADICS). IEEE (2024)
52. Wang, Z., Yu, W., Ren, X., Zhang, J., Zhao, Y., Saxena, R., Cheng, L., Wong, G., See, S., Minervini, P., Song, Y., Steedman, M.: Mmlongbench: Benchmarking long-context vision-language models effectively and thoroughly. In: NeurIPS (2025)
53. 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. In: NeurIPS (2024)
54. Xiong, B., Liu, R., Xu, K., Chen, M., Feng, A.: Splat feature solver. In: ICLR (2026)
55. Xu, R., Wang, X., Wang, T., Chen, Y., Pang, J., Lin, D.: Pointllm: Empowering large language models to understand point clouds. In: ECCV (2024)
56. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025)
57. Yang, Y., Yang, H., Zhou, J., Chen, P., Zhang, H., Du, Y., Gan, C.: 3d-mem: 3d scene memory for embodied exploration and reasoning. In: CVPR (2025)
58. Ye, M., Danelljan, M., Yu, F., Ke, L.: Gaussian grouping: Segment and edit anything in 3d scenes. In: ECCV (2024)
59. Yin, H., Xu, X., Wu, Z., Zhou, J., Lu, J.: Sg-nav: Online 3d scene graph prompting for llm-based zero-shot object navigation. In: NeurIPS (2024)

60. Yuan, Y., Li, W., Liu, J., Tang, D., Luo, X., Qin, C., Zhang, L., Zhu, J.: Osprey: Pixel understanding with visual instruction tuning. In: CVPR (2024)
61. Zeng, A., Attarian, M., Ichter, B., Choromanski, K., Wong, A., Welker, S., Tombari, F., Purohit, A., Ryoo, M., Sindhwani, V., et al.: Socratic models: Composing zero-shot multimodal reasoning with language. In: ICLR (2023)
62. Zhang, C., Delitzas, A., Wang, F., Zhang, R., Ji, X., Pollefeys, M., Engelmann, F.: Open-vocabulary functional 3d scene graphs for real-world indoor spaces. In: CVPR (2025)
63. Zhang, H., Diao, S., Lin, Y., Fung, Y., Lian, Q., Wang, X., Chen, Y., Ji, H., Zhang, T.: R-tuning: Instructing large language models to say 'i don't know'. In: North American Chapter of the Association for Computational Linguistics (2024)
64. Zhang, Y., Gong, Z., Chang, A.X.: Multi3drefer: Grounding text description to multiple 3d objects. In: ICCV (2023)
65. 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: CVPR (2024)
66. Zwicker, M., Pfister, H., Van Baar, J., Gross, M.: Ewa volume splatting. In: Proceedings Visualization, 2001. VIS'01. IEEE (2001)