The Point, the Vision and the Text: Does Point Cloud Boost Spatial Reasoning of Large Language Models?

Anonymous ACL submission

Abstract

3D Large Language Models (LLMs) leveraging spatial information in point clouds for 3D spatial reasoning attract great attention. Despite some promising results, the advantages of point clouds over other modalities remain unclear. Moreover, existing 3D benchmarks are insufficient for fairly evaluating the ability of multimodal LLMs to comprehend spatial concepts. To address these challenges, we introduce ScanReQA, a 3D spatial reasoning benchmark encompassing text, vision, and point cloud modalities. We then evaluate the performance of text, 2D, and 3D LLMs on the benchmark to compare the effectiveness of different modalities in understanding spatial concepts. Furthermore, we analyze the reasoning mechanisms behind 3D LLMs using point clouds. Our findings reveal that: 1) current 3D LLMs struggle with reasoning about binary spatial relationships, 2) 3D LLMs with point cloud input do not outperform other modalities, and 3) 3D LLMs exhibit the attention sink phenomenon similar to that in 2D LLMs, impairing spatial reasoning. We think these conclusions can help the next step of 3D LLMs and also offer insights for foundation models in other modalities.

1 Introduction

Equipping models with spatial reasoning is essential for bridging the gap between artificial and human intelligence and is a critical step toward general intelligence. While LLMs (Touvron et al., 2023; Achiam et al., 2023; Brown et al., 2020; Bai et al., 2023) and vision-language models (VLMs) (Zhu et al., 2024; Bai et al., 2023; Team et al., 2023) have demonstrated strong performance in tasks such as commonsense reasoning (Davis, 2023), math problem solving (Liu et al., 2025), and scientific QA (Lu et al., 2022), they remain limited in spatial reasoning due to the absence of 3D perception. To address this, recent efforts incorporate 3D spatial information from point clouds into LLMs (Zhu et al., 2023; Man et al., 2024b), enabling progress in tasks such as 3D captioning, spatial QA, and embodied navigation.

Intuitively, 3D LLMs equipped with spatial cues from point clouds should outperform text-only LLMs or VLMs in spatial reasoning. However, existing results tell a different story. Current 3D LLMs achieve only around 50% accuracy on 3D QA benchmarks (Azuma et al., 2022; Ma et al., 2022), falling short of SOTA results in visual (Liu et al., 2024c) and text-based QA (Hendrycks et al., 2020). Moreover, they often fail to generalize spatial relationships across scene variations (Man et al., 2024a), raising concerns about *whether point cloud inputs genuinely enhance the spatial reasoning capabilities of 3D LLMs*.

To answer this question, there are two challenges to overcome: 1) how to evaluate the spatial reasoning capacities of 3D LLMs? Although current 3D benchmarks include various spatial reasoning tasks (Azuma et al., 2022; Ma et al., 2022; Zhi et al., 2025), the spatial relationships in these benchmarks often exhibit a bias towards object semantics. Some spatial relationships consistently co-occur with specific objects (Peng et al., 2025), causing models to overfit to these fixed spatial relationships when encountering similar objects. 2) How can consistent scene information be represented across text, vision, and point cloud modalities to enable a fair comparison of spatial reasoning abilities among text, 2D, and 3D LLMs? It is essential to ensure that all three modalities convey the necessary spatial reasoning cues while not introducing information not grounded in the original scene.

To address the first challenge, we construct ScanReQA, an unbiased 3D QA benchmark based on ScanQA as shown in Figure 1. ScanReQA focuses on binary spatial relationships in ScanQA that can be represented by reversible spatial concepts (e.g., *"left"* and *"right"*) and a pair of objects. We design both forward and reverse spatial reasoning ques-

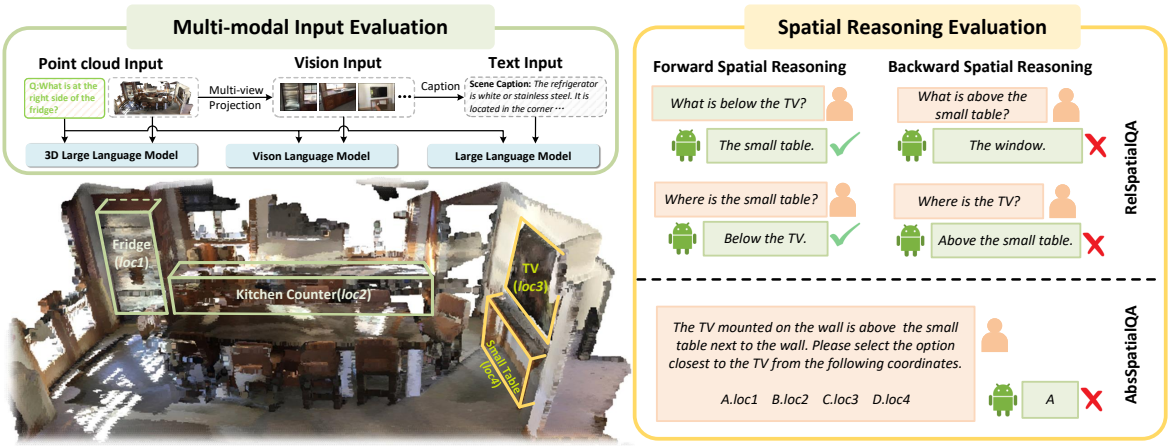

Figure 1: **The overview of our evaluation framework on 3D LLMs.** In the multimodal input evaluation, we generate point cloud inputs along with their visual and textual counterparts, which are fed into the corresponding models. For spatial reasoning evaluation, 3D LLMs are tasked with both forward and backward spatial reasoning, as well as predicting the location of the referenced object. *loc#* is the 3D coordinate of the corresponding object.

tions for each object pair, such as *"what is A to B"* and *"what is B to A"*. This design ensures that forward and reversed spatial relationships occur with equal frequency across object pairs, mitigating biases in both object and relationship distributions. Notably, ScanReQA is **NOT** an independent benchmark from ScanQA; rather, it complements ScanQA by specifically addressing the bias caused by the co-occurrence of spatial relationships with particular objects.

To address the second challenge, we design an evaluation framework as shown in Figure 1. All scene representations across different modalities are derived from original scene videos, with point cloud and textual representations generated through 3D projection and image captioning, respectively. Subsequently, we solely input each modality's scene representation along with the corresponding questions into the respective LLMs to compare the performance differences. We present the performance of models with different modality inputs and highlight that LLMs and VLMs can achieve competitive results even without the point cloud input.

Furthermore, we conduct a comprehensive analysis of how 3D LLMs process point cloud inputs by examining attention distributions and semantic token flow across model layers. Specifically, we visualize attention to point cloud tokens at different layers, and assess the impact of token category on spatial reasoning performance via masking different parts of the inputs. Additionally, we adopt the logit lens to interpret point cloud embeddings as textual tokens, offering intuitive insights into the model's reasoning process.

In this work, we introduce an unbiased 3D QA benchmark for binary spatial relationship reasoning and conduct a comprehensive evaluation of the state-of-the-art (SOTA) multimodal LLMs. The results show that LLMs without point cloud input outperform 3D LLMs by more than 10%, with the average accuracy of 3D LLMs on ScanReQA being approximately 5%. Furthermore, in-depth analysis reveals the attention sink and semantic misalignment phenomena in 3D LLMs, offering valuable insights for advancing the spatial reasoning capabilities of these models.

The main contributions are summarized below:

- We present the first systematic evaluation of how multimodal inputs affect 3D spatial reasoning, providing a comprehensive assessment of 3D LLMs' reasoning capabilities.
- We introduce ScanReQA, the first unbiased 3D QA benchmark designed to evaluate existing 3D LLMs on binary spatial relationship reasoning.
- We conduct an in-depth analysis of attention patterns and token-level semantic flow in 3D LLMs, revealing two critical factors that affect spatial reasoning: attention sink and semantic misalignment.

2 Related Work

2.1 Aligning 3D point cloud with LLM

Unlike 2D vision-language tasks where models like CLIP (Radford et al., 2021) and large-scale image–text datasets enable effective feature alignment, 3D understanding remains challenging due to the scarcity of paired point cloud–text data.

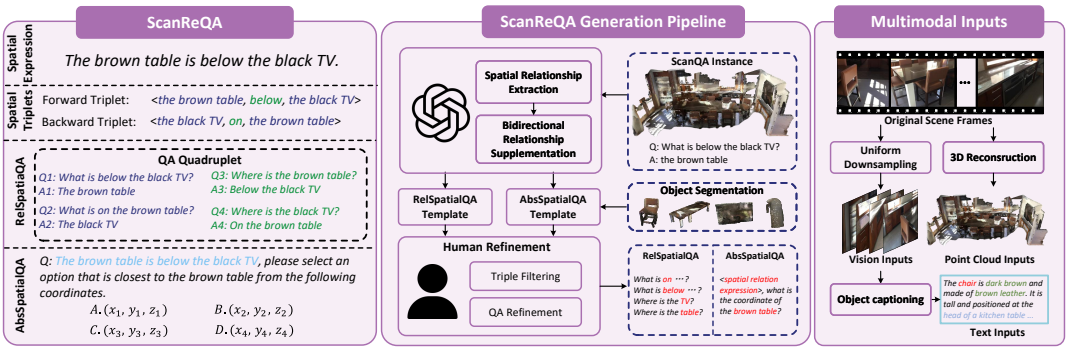

Figure 2: **Overview of ScanReQA and its generation pipeline.** *Left*: ScanReQA is derived from spatial expressions in the ScanQA dataset. Binary spatial relationships are extracted and represented as forward and backward triplets, which are used to construct QA quadruplets in RelSpatialQA and coordinate-based options in AbsSpatialQA. Each QA quadruplet involves questions reasoning about both objects and their spatial relationships. *Middle*: LLMs extract spatial expressions and generate triplets, which are filled into QA templates and later refined by human annotators. *Right*: Original scene frames are uniformly downsampled and reconstructed into 3D point clouds to produce 2D/3D inputs, while visual inputs are used for object captioning to generate textual descriptions.

Most existing methods follow a two-stage framework: a 3D encoder first extracts geometric features, and a projector maps them into the token space of LLMs. Representative works include 3D-LLM (Hong et al., 2023), PointLLM (Xu et al., 2025), 3D-VLA (Zhen et al., 2024), and Agent3D-Zero (Zhang et al., 2025b). Common 3D encoders include Mask2Former (Cheng et al., 2022), SAM (Kirillov et al., 2023), Point-BERT (Yu et al., 2022), Uni-3D (Zhang et al., 2023a), and VoxelNet (Zhou and Tuzel, 2018), while some approaches enhance semantics using pre-trained scene graph models.

To capture fine-grained spatial semantics, models such as 3DMIT (Li et al., 2024), LEO (Huang et al., 2023b), and Chat-3D (Wang et al., 2023) segment scenes and encode object-level features with dedicated encoders for attributes (e.g., color, size, shape) and inter-object relations. Distinctively, LL3DA (Chen et al., 2024a) introduces a unified query generator that fuses visual prompts, textual instructions, and 3D embeddings into fixed-length tokens before aligning them with the token space.

2.2 MLLM on spatial reasoning

Typical 3D spatial reasoning tasks include captioning, grounding, question answering, dialogue, planning, navigation, and manipulation. Recent multimodal LLMs have shown strong potential for complex spatial reasoning and human–agent interaction. However, while models such as LLaVA (Liu et al., 2024b), Flamingo (Alayrac et al., 2022), BLIP-2 (Li et al., 2023), and PaLM-E (Driess et al., 2023) perform well in 2D vision-language tasks, they remain limited in 3D understanding and reasoning.

To bridge this gap, recent studies extend LLMs or 2D VLMs to 3D domains by integrating point cloud features with textual inputs, leveraging LLMs' semantic reasoning ability to generate context-aware outputs. Moreover, the planning, decision-making, and tool-use capabilities of MLLMs make them promising backbones for embodied 3D agents that can navigate (Huang et al., 2023b; Rana et al., 2023; Zheng et al., 2024), interact with objects (Mirjalili et al., 2023), and use tools (Hong et al., 2024) while performing complex spatial reasoning.

3 ScanReQA Design and Construction

In this part, we first introduce the insights of benchmark design and then clarify the detailed construction pipeline.

3.1 Benchmark Design Principles

Due to factors like dataset distribution differences and biases affecting model performance, it is crucial to eliminate these confounders to ensure that the model's performance accurately reflects its spatial reasoning abilities. Thus, the design of ScanReQA follows three key principles.

Data distribution consistency *This principle eliminates the effect of dataset distribution differences on model performance.* If the model understands a spatial concept rather than fitting to a specific QA pattern, it should be able to identify the same spatial relationship across different datasets. However, since model performance is affected by dataset distributions, we align Scan-

ReQA with ScanQA (Azuma et al., 2022) in terms of scene selection, spatial relationship types, and question formats. This ensures that any performance differences between the two benchmarks are primarily due to the model's understanding of spatial relationships, rather than differences in dataset distribution. Specifically, ScanReQA aligns with ScanQA by 1) selecting the same 3D scenes from ScanNet (Dai et al., 2017), 2) containing the same spatial relationships, and 3) maintaining consistent question styles and formats.

QA Unbiasedness *This eliminates the model's bias towards specific spatial relationships and objects.* In current benchmarks such as ScanQA, certain object categories appear with significantly higher frequency than others. A similar imbalance is observed in the distribution of spatial relations. As a result, models trained on ScanQA tend to develop biases toward specific object types and spatial relation patterns. To address this issue, ScanReQA builds upon ScanQA by explicitly constructing QA pairs that cover a diverse range of object categories and spatial relations. This design ensures a more balanced frequency distribution across different types of objects and spatial relations in the dataset, thereby promoting fairness and reducing evaluation bias.

Multimodal information consistency *This eliminates the effect of multimodal information inconsistency on model performance.* To evaluate and compare the spatial reasoning performance of large models across different modalities, ScanReQA represents scene information using text, vision, and point clouds. To ensure a fair comparison, all modality data is derived from the original ScanNet scene videos. Specifically, point cloud data is obtained through 3D projection, images are generated from different viewpoints via video downsampling, and scene text descriptions are created through image captioning. This approach ensures consistency across modalities, thereby guaranteeing a fair evaluation.

3.2 Benchmark Generation Pipeline

Our proposed benchmark, ScanReQA, focuses on binary spatial relationships, which reflect the fundamental spatial reasoning capabilities of models. The underlying insight is that if a model understands binary spatial relationships, given "A is left to B," it should be able to reason that "B is right to A." Moreover, understanding binary spatial relationships is a necessary capability for handling more complex spatial relationships.

Generally, a binary spatial relationship can be formalized as a triplet $\langle t, r, a \rangle$ (Dai et al., 2017; Zhang et al., 2023b), where t, r, and a denote the target object, the spatial relationship phase, and the anchor object, respectively. If the relationship is reversible, it also has an equivalent representation $\langle a, r^{-1}, t \rangle$. For simplicity, we refer to $\langle t, r, a \rangle$ as the forward triplet and $\langle a, r^{-1}, t \rangle$ as the backward triplet. We define the spatial reasoning capacity of 3D LLMs in two aspects: 1) the ability to infer the third element of a forward/backward triplet given any two elements, which is formulated as:

$$\begin{aligned} &LLM(a, r; \theta) \Rightarrow t, \quad LLM(t, r^{-1}; \theta) \Rightarrow a, \\ &LLM(a, t; \theta) \Rightarrow r, \quad LLM(t, a; \theta) \Rightarrow r^{-1}, \end{aligned} \tag{1}$$

where θ is the model weights. 2) The ability to infer the target object's coordinates given a spatial triplet, which is formulated as:

$$\begin{aligned} &LLM(a, r, t; \theta) \Rightarrow C_t, \\ &LLM(t, r^{-1}, a; \theta) \Rightarrow C_a, \end{aligned} \tag{2}$$

where C_t and C_a denote the numerical coordinates of the target and anchor objects, respectively. To assess these two capabilities, ScanReQA introduces two distinct QA datasets: RelSpatialQA and AbsSpatialQA, which contain QAs about direction and coordinate spatial reasoning. The complete generation pipeline is depicted in Figure 2.

Binary Spatial Relationship Extraction Following the first benchmark design principle outlined in the previous section, ScanReQA is constructed based on the binary spatial relationships extracted from ScanQA. Specifically, we prompt the LLM to: 1) convert the QA pair into a spatial relationship expression, as illustrated in Figure 2, 2) verify the reversibility of the spatial relationship, and 3) represent the relationship as a triplet, referred to as a spatial triplet.

Bidirectional Relationship Supplementation Following the second principle, we supplement each extracted relationship with its reversible counterpart. For example, if the original triplet is $\langle A, left, B \rangle$, the reversed triplet would be $\langle B, right, A \rangle$. We then query both the relationship of A to B and B to A, as well as what is to the left of B and what is to the right of A. In this case, each binary spatial relationship corresponds to four QAs. This ensures that the frequency of binary relationships and objects in the answers remains balanced.

Template-based QA generation We follow the

first principle to design four QA templates for RelSpatialQA and one template for AbsSpatialQA based on the ScanQA style. The QA templates are shown in Figure 2.

Multi-modal input generation Following the last principle, we generate scene representations in text, vision, and point cloud modalities, all sourced from ScanNet scene videos. For *video* to *point cloud* conversion, we directly use the ScanNet point cloud obtained by 3D scene reconstruction. For *video* to *vision* conversion, we uniformly downsample the original video in 64 frames to match the input format for video LLMs (Zheng et al., 2025; Chen et al., 2024b). For *video* to *text* conversion, we follow SceneVerse (Jia et al., 2025) by projecting object point clouds onto a 2D plane to identify their presence in RGB frames, and then use BLIP2 (Li et al., 2023) to generate object captions. Details of the conversion process are provided in Appendix A.1.

Quality Check To address LLM hallucination, we implement quality control. For RelSpatialQA, human annotators verify the correctness of triplets, discarding those that are illogical or nonexistent. Grammatical errors are also corrected. For AbsSpatialQA, annotators filter out incorrect spatial expressions and their corresponding QAs.

Dataset Statistics After refinement, we obtain a total of **5,523** QA pairs in RelSpatialQA. Note that, according to Equation 1 and Figure 1, a binary spatial relationship corresponds to four QA pairs in RelSpatialQA. We assume that if a model can correctly answer all four QAs, it demonstrates a complete understanding of the binary spatial relationship. We denote these four QAs as a QA quadruplet. In total, RelSpatialQA contains **1,104** quadruplets. In AbsSpatialQA, a total of **2,672** QAs are collected.

4 Evaluation on Multi-modal Inputs

To investigate the effect of different modalities, we conduct controlled experiments by altering the input modalities of MLLMs. Since scenes can be represented in text, vision, or 3D point clouds, our key idea is to replace one modality with another (e.g., substituting point clouds with images or images with text). We design three modality combinations for LLMs, VLMs, and 3D LLMs, respectively: 1) Text-only: scene descriptions as input; 2) Vision-only: multi-view images of the scene; 3) Point cloud-Vision: matches the original 3D LLM input, including point clouds with either RGB values or multi-view images.

5 Experiments

We first provide the details of the experimental setup and then demonstrate the following: overall performance of MLLMs, ablations of multimodal inputs, and in-depth analysis of 3D MLLM results. These correspond to addressing the following three research questions (RQs):

- **RQ1**: What's the performance of 3D LLMs compared to other models?
- **RQ2**: Do 3D LLMs truly perform reasoning?
- **RQ3**: How do 3D LLMs leverage point clouds to answer spatial-related questions?

5.1 Experimental Setup

Models We evaluate three different types of large models from existing SOTA works. 1) LLMs: Basic LLMs that accept only text input. We select competitive LLMs, including Qwen2.5-Max (Hui et al., 2024), DeepSeek V3 (Liu et al., 2024a), and LLaMA-3 (Dubey et al., 2024). 2) VLMs: Multi-modal LLMs that support visual input. For evaluation, we choose GPT-4o (Achiam et al., 2023), Gemini-1.5 Flash (Team et al., 2024), LLaMA-3-11B (Dubey et al., 2024), InternVL 8B/26B (Chen et al., 2024b), and Qwen VL 2B/7B (Bai et al., 2023). 3) 3D LLMs: 3D multi-modal LLMs that support point cloud input. We evaluate SOTA methods, including 3D-LLM (Hong et al., 2023), LL3DA (Chen et al., 2024a), LEO (Huang et al., 2023b), Chat-3D v2 (Wang et al., 2023) (we term it as Chat-3D for simplicity), and Chat-Scene (Huang et al., 2023a). The inputs for the three types of models are TI, VI, and PVI, respectively.

Datasets We evaluate all models on three 3D QA benchmarks, ScanQA, SQA3D, and our proposed ScanReQA. For ScanReQA, only 3D LLMs are evaluated on the AbsSpatial QA dataset.

Metrics We use the refined top-1 exact match (EM@1) introduced in LEO (Huang et al., 2023b) for overall performance comparison. For RelSpatialQA, we calculate the accuracy and recall of understanding spatial relationships, which are defined as follows:

$$\text{Accuracy} = \frac{N^{\text{quad}}_{\text{correct}}}{N^{\text{quad}}_{\text{total}}}, \ \text{Recall} = \frac{N^{\text{quad}}_{\text{correct}}}{N^{\text{ScanQA}}_{\text{correct}}}. \quad (3)$$

where $N^{\text{quad}}_{\text{correct}}$, $N^{\text{quad}}_{\text{total}}$, $N^{\text{ScanQA}}_{\text{correct}}$ denote the number of correctly answered quadruplets, the total number of quadruplets, and the number of correctly

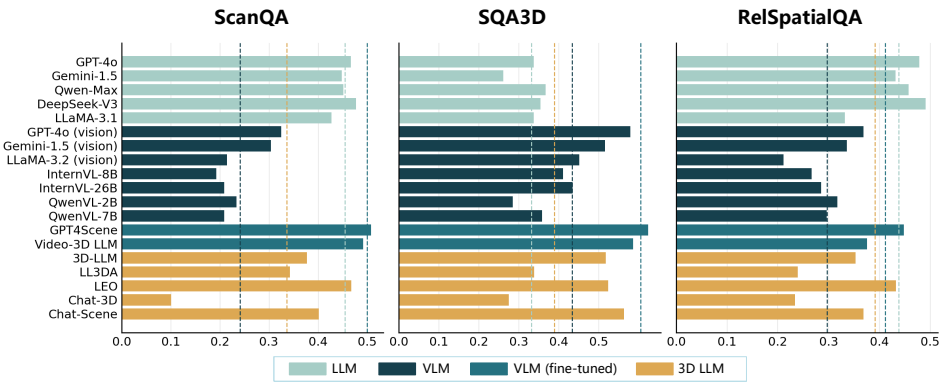

Figure 3: **Performance overview on ScanQA, SQA3D, and ScanReQA with different modality input.**

answered QAs in ScanQA, respectively. More detailed explanations of recall are provided in Appendix A.4. For AbsSpatial QA, we simply calculate the accuracy of multiple-choice questions.

5.2 What's the performance of 3D LLMs?

The EM@1 scores of all evaluated models are presented in Figure 3, with vertical dashed lines indicating the average EM@1 for each model type.

3D LLMs with point cloud do not exhibit overwhelming advantages over LLMs and VLMs The average EM@1 scores of 3D LLMs in the three datasets are 33.8%, 44.4% and 32.6%, while the highest scores achieved are 50.0%, 60.5%, and 43.8%, respectively. The performance gap between 3D LLMs and the best-performing models exceeds 10% on all three datasets, and 3D LLMs only marginally outperform zero-shot VLMs and LLMs.

Text and vision modalities exhibit competitive performance As shown in Figure 3, zero-shot LLMs achieve 46.7% and 44.5% EM@1 on ScanQA and RelSpatialQA, ranking second and first, respectively. Zero-shot VLMs reach 43.2% EM@1, the second-best performance on SQA3D. Moreover, **fine-tuning significantly enhances the spatial reasoning ability of VLMs**, yielding the highest EM@1 in both ScanQA and SQA3D, and second best in RelSpatialQA. These results highlight the strong potential of VLMs with multiview image inputs for 3D spatial understanding.

In summary, existing 3D LLMs with point cloud inputs do not exhibit clear advantages over LLMs utilizing plain text or vision inputs.

5.3 Do 3D LLMs Perform Reasoning?

To answer RQ2, we assume that the 3D LLM should be able to answer the quadruplet QAs in RelSpatialQA if it truly understands the binary spatial relationship. We also investigate whether 3D LLMs can leverage the coordinate information of the point clouds for more complex spatial reasoning. Thus, we analyze the performance of baseline models on ScanReQA.

Table 1: Quantitative results on ScanReQA.

Methods	RelSpatialQA		AbsSpatialQA
	Acc.	Recall	Acc.
Text-only			
GPT-4o	17.7	26.4	-
Gemini-1.5 Flash	15.8	24.1	-
Qwen2.5-Max	15.8	23.9	-
DeepSeek-V3	19.8	27.6	-
LLaMA-3-8B	11.0	16.5	-
Average	16.0	23.7	-
Vision-only			
GPT-4o	4.5	7.9	-
Gemini-1.5 Flash	4.8	8.5	-
LLaMA-3-11B	2.3	3.2	-
InternVL-8B	2.4	4.1	-
InternVL-26B	3.1	6.7	-
QwenVL-2B	3.9	7.1	-
QwenVL-7B	2.6	5.9	-
Average	3.3	6.2	-
Vision-only (*fine-tuned*)			
GPT4Scene	12.2	17.5	-
Video-3D-LLM	10.3	14.5	-
Average	11.3	16.0	-
Point cloud-vision			
3D-LLM	5.7	8.7	24.2
LL3DA	5.3	9.7	1.5
LEO	11.2	12.9	0.1
Chat-3D	1.0	2.5	0.0
Chat-Scene	8.4	13.3	0.0
Average	6.3	9.4	5.4
Overall Average	8.3	12.7	-

Elementary spatial reasoning remains challenging We report the accuracy and recall of all models on RelSpatialQA. The results show that: 1) overall accuracy and recall are extremely low, at only 8.3% and 12.7%, respectively; 2) LLMs achieve the highest accuracy while zero-shot VLMs achieve the lowest, with 3D LLMs falling between them. These findings indicate that current models struggle to understand the basic binary spatial relationships for reasoning. Furthermore, incorporat-

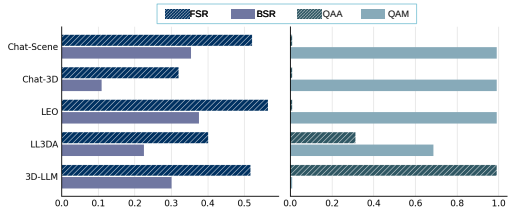

Figure 4: Left: The EM@1 of forward/backward QAs. Right: the ratio of instruction following in AbsSpatialQA. FSR, BSR, QAA, and QAM denote forward, backward reasoning, QA alignment, and mismatch.

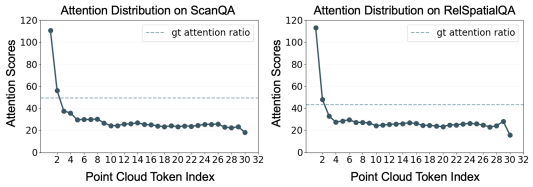

Figure 5: Attention distribution over token indices.

ing point cloud information does not significantly improve the spatial relationship understanding or reasoning capabilities.

3D coordinates in point clouds are not sufficiently utilized We argue that the primary advantage and uniqueness of point clouds lie in their 3D coordinate information, which enables models to predict object locations and reason about spatial relationships. However, as shown in Table 1, most 3D LLMs achieve near-zero accuracy on AbsSpatialQA. Even the best-performing model, 3D-LLM, only reaches an accuracy of 24.2%, which is lower than random guessing (25%). These results indicate that although 3D LLMs are capable of reasoning about qualitative spatial relationships (e.g., RelSpatialQA or ScanQA), they still lack the ability to perform precise quantitative spatial reasoning.

3D LLMs struggle to reason for backward spatial relationships Due to the low accuracy and recall (reported in Table 1) of answering QA quadruplets in RelSpatialQA, we assume that 3D LLMs fail to reason about the reversed spatial relationships even if they correctly answer the forward spatial QAs. To verify this hypothesis, we evaluate the accuracy of 3D LLMs on forward and backward spatial QAs, respectively. The results are depicted in Figure 4. The accuracy for forward spatial relationships is typically above 50%, whereas for backward relationships, it drops significantly to around 30%. This result suggests that 3D LLMs struggle to leverage the knowledge of forward spatial triplets to reason for the backward ones, indicating substantial improvement space for 3D LLMs to understand and reason about spatial relationships.

Table 2: Ablations on masking different tokens.

	w/ sink (tokens)	wo/ sink	w/ GT (tokens)	wo/ GT
ScanQA	29.5	31.8	29.5	26.5
RelSpatialQA	28.5	28.4	28.5	26.3

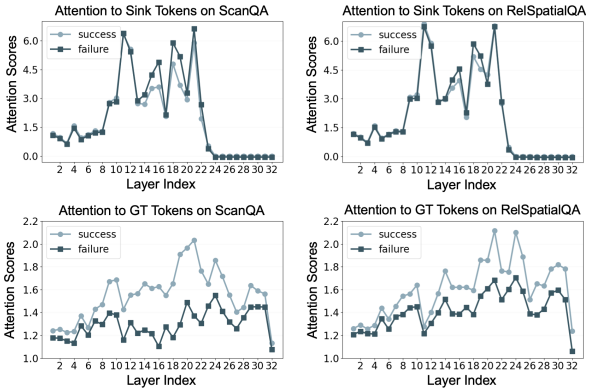

Figure 6: The attention distribution of successful and failed cases over transformer layers.

3D LLMs suffer from QA mismatch Although multiple-choice questions (MCQs) are a common QA format for LLMs, we find that a significant portion of the errors arise from QA mismatches, where the 3D LLMs' responses are irrelevant to the question intent. The ratio of QA mismatches is shown in Figure 4. The results indicate that most 3D LLMs (e.g., Chat-Scene, Chat-3D, and LEO) fail to follow the given instructions and instead output their reasoning process, which is the primary cause of their poor performance on AbsSpatialQA.

In summary, existing 3D LLMs demonstrate limited spatial understanding and reasoning capabilities, and struggle to explicitly exploit the coordinate information of point clouds.

5.4 How do 3D LLMs Leverage Point Clouds? A Representation Perspective.

To answer this question, we conduct an in-depth analysis of the relationship between model attention to point clouds and QA accuracy, as well as interpret point cloud embeddings into the vocabulary to explore the information flow during the reasoning process. Our study focuses on LEO (Huang et al., 2023b), a representative open-source 3D LLM using the popular Vicuna-7B language model.

Analysis from the Attention Perspective We analyze the model's response attention to input tokens by computing the average attention scores from response tokens to point cloud tokens across all transformer layers. Details of the computation are provided in Appendix A.2. Representative 3D LLMs such as Chat-Scene, Chat-3D, and LEO segment object-level point clouds and encode each ob-

ject into a single token, which is then fed into the LLM backbone. *To account for the unordered nature of point clouds, we randomly shuffle the point cloud tokens prior to input*. The key observations and findings are summarized below.

- **Attention sink exists in 3D LLM** As depicted in Figure 5, the results show that the model's responses consistently exhibit high attention to the first few object tokens. In other words, attention is concentrated on fixed token positions, which is known as the attention sink phenomenon observed in VLMs (Xiao et al., 2023). We refer to these fixed-position tokens as **sink tokens**.
- **Attention on sink tokens does not contribute to reasoning** We mask sink tokens in the input and evaluate the 3D LLM's performance, as shown in Table 2. We also compare the attention scores of sink tokens between successful and failed QA cases (Figure 6). The results show that masking sink tokens does not reduce QA accuracy and even slightly improves it on the ScanQA dataset. This suggests that sink tokens contribute little to reasoning and may even hinder the model's reasoning process, consistent with findings in VLMs (Xiao et al., 2023).
- **Attention on QA-Relevant Object Tokens is necessary for reasoning** We denote the object tokens correspond to anchor or target objects in the spatial relationship as **ground-truth (GT) tokens**. To assess their importance, we 1) mask these tokens and measure performance drop, and 2) compare their attention scores in successful and failed QA cases. As shown in Table 2 and Figure 6, greater attention to GT tokens correlates with higher QA accuracy, while masking them results in a performance drop of more than 2%. This highlights their necessity for effective spatial reasoning.

Analysis from the Feature Perspective We use the *logit lens* (Nostalgebraist, 2020) method to see how the representations of point cloud tokens evolve through the transformer layers. The logit lens technique uses a decoder to interpret intermediate embeddings into the vocabulary. Detailed experimental settings are provided in Appendix A.3. We select QA cases whose GT tokens have the highest attention scores and depict the logit lens results in Figure 7.

- **Point cloud representations evolve into interpretable vocabulary tokens** As shown in Figure 7, we find that the point cloud embeddings in the shallow and late layers correspond to tokens in the vocabulary that describe its original object in the scene. This demonstrates the applicability of the logit lens to 3D LLMs, extending its use beyond text and vision (Neo et al., 2024) to the point cloud modality.
- **Incorrect token interpretation leads to reasoning failure** As shown in the second case in Figure 7, although the model attends to the correct tokens, it fails to align point cloud semantics with the corresponding textual vocabulary during inference, resulting in incorrect answers.
- **Attention on incorrect object token leads to reasoning failure** As depicted in the last case in Figure 7, even if the model interprets the point cloud token correctly, it pays more attention on irrelevant objects, leading to an incorrect answer.

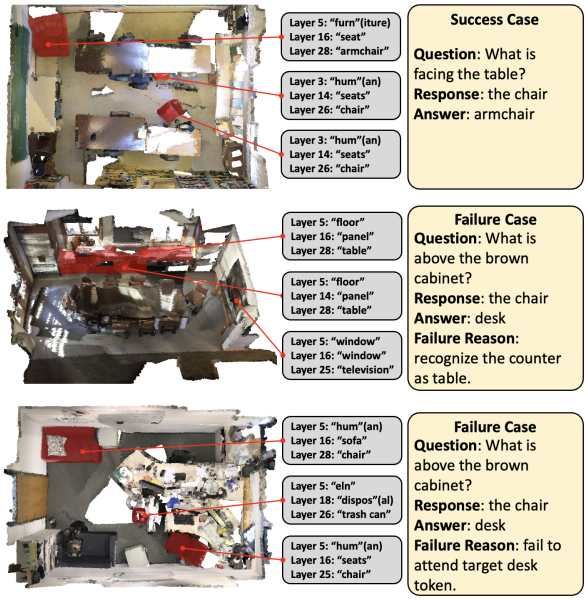

Figure 7: The attention distribution of successful and failed cases over transformer layers.

In summary, we validate and conclude two essential abilities for 3D LLMs' spatial reasoning: attention on the question-related point cloud tokens and correct semantic alignment between the point cloud and text.

6 Conclusion

Our results suggest that current 3D LLMs are still in the early stage of advancing 3D spatial reasoning. Compared with 2D LLMs, their reasoning remains less mature, particularly in aligning point cloud and textual modalities and in leveraging the quantitative geometric information. Consequently, there remains substantial room for improvement in effectively utilizing point cloud information to enhance spatial reasoning performance.

7 Limitations

First, this work focuses primarily on binary spatial reasoning, without extending the analysis to more complex multi-object spatial relationships. While binary relations form the foundation of higher-order spatial reasoning, it remains an open question whether current 3D LLMs exhibit similar reasoning behaviors in multi-object scenarios.

Second, although we provide an in-depth analysis of the current challenges faced by 3D LLMs, we do not propose concrete model-level improvements for enhancing spatial reasoning ability. One possible direction, similar to techniques used in 2D LLMs, is the attention reassignment (Kang et al., 2025) to encourage the model to attend to more informative tokens. However, the effectiveness of such strategies still requires further empirical validation.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, and 1 others. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. 2022. Scanqa: 3d question answering for spatial scene understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19129–19139.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization. *Text Reading, and Beyond*, 2.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. 2024a. Ll3da: Visual interactive instruction tuning for omni-3d understanding reasoning and planning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26428–26438.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, and 1 others. 2024b. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198.

Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. 2022. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299.

Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839.

Ernest Davis. 2023. Benchmarks for automated commonsense reasoning: A survey. *ACM Computing Surveys*, 56(4):1–41.

Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, and 1 others. 2023. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 2023. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 36:20482–20494.

Yining Hong, Zishuo Zheng, Peihao Chen, Yian Wang, Junyan Li, and Chuang Gan. 2024. Multiply: A multisensory object-centric embodied large language model in 3d world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26406–26416.

Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, and 1 others. 2023a. Chat-scene: Bridging 3d scene and large language models with object identifiers. *arXiv preprint arXiv:2312.08168*.

Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. 2023b. An embodied generalist agent in 3d world. *arXiv preprint arXiv:2311.12871*.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, and 1 others. 2024. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*.

Baoxiong Jia, Yixin Chen, Huangyue Yu, Yan Wang, Xuesong Niu, Tengyu Liu, Qing Li, and Siyuan Huang. 2025. Sceneverse: Scaling 3d vision-language learning for grounded scene understanding. In *European Conference on Computer Vision*, pages 289–310. Springer.

Seil Kang and 1 others. 2025. See what you are told: Visual attention sink in large multimodal models. *arXiv:2503.03321*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, and 1 others. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Zeju Li, Chao Zhang, Xiaoyan Wang, Ruilong Ren, Yifan Xu, Ruifei Ma, Xiangde Liu, and Rong Wei. 2024. 3dmit: 3d multi-modal instruction tuning for scene understanding. In *2024 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, pages 1–5. IEEE.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024b. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Tianqiao Liu, Zui Chen, Zhensheng Fang, Weiqi Luo, Mi Tian, and Zitao Liu. 2025. Matheval: A comprehensive benchmark for evaluating large language models on mathematical reasoning capabilities. *Frontiers of Digital Education*, 2(2):16.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, and 1 others. 2024c. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. 2022. Sqa3d: Situated question answering in 3d scenes. *arXiv preprint arXiv:2210.07474*.

Yunze Man, Liang-Yan Gui, and Yu-Xiong Wang. 2024a. Situational awareness matters in 3d vision language reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13678–13688.

Yunze Man, Shuhong Zheng, Zhipeng Bao, Martial Hebert, Liang-Yan Gui, and Yu-Xiong Wang. 2024b. Lexicon3d: Probing visual foundation models for complex 3d scene understanding. *arXiv preprint arXiv:2409.03757*.

Reihaneh Mirjalili, Michael Krawez, Simone Silenzi, Yannik Blei, and Wolfram Burgard. 2023. Lan-grasp: Using large language models for semantic object grasping. *arXiv preprint arXiv:2310.05239*.

Clement Neo, Luke Ong, Philip Torr, Mor Geva, David Krueger, and Fazl Barez. 2024. Towards interpreting visual information processing in vision-language models. *arXiv preprint arXiv:2410.07149*.

Nostalgebraist. 2020. Interpreting gpt: The logit lens. https://www.alignmentforum.org/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens. Accessed: 2024-09-23.

Ruiying Peng, Kaiyuan Li, Weichen Zhang, Chen Gao, Xinlei Chen, and Yong Li. 2025. Understanding and evaluating hallucinations in 3d visual language models. *arXiv preprint arXiv:2502.15888*.

Zhangyang Qi, Zhixiong Zhang, Ye Fang, Jiaqi Wang, and Hengshuang Zhao. 2025. Gpt4scene: Understand 3d scenes from videos with vision-language models. *arXiv preprint arXiv:2501.01428*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian D Reid, and Niko Suenderhauf. 2023. Sayplan: Grounding large language models using 3d scene graphs for scalable task planning. *CoRR*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, and 1 others. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Johanna Wald, Armen Avetisyan, Nassir Navab, Federico Tombari, and Matthias Nießner. 2019. Rio: 3d object instance re-localization in changing indoor environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7658–7667.

Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Sharon Li, and Neel Joshi. 2025. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. *Advances in Neural Information Processing Systems*, 37:75392–75421.

Zehan Wang, Haifeng Huang, Yang Zhao, Ziang Zhang, and Zhou Zhao. 2023. Chat-3d: Data-efficiently tuning large language model for universal dialogue of 3d scenes. *arXiv preprint arXiv:2308.08769*.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2023. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*.

Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. 2025. Pointllm: Empowering large language models to understand point clouds. In *European Conference on Computer Vision*, pages 131–147. Springer.

Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. 2024. Thinking in space: How multimodal large language models see, remember, and recall spaces. *arXiv preprint arXiv:2412.14171*.

Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. 2022. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19313–19322.

Jiarui Zhang, Mahyar Khayatkhoei, Prateek Chhikara, and Filip Ilievski. 2025a. Mllms know where to look: Training-free perception of small visual details with multimodal llms. *arXiv preprint arXiv:2502.17422*.

Sha Zhang, Di Huang, Jiajun Deng, Shixiang Tang, Wanli Ouyang, Tong He, and Yanyong Zhang. 2025b. Agent3d-zero: An agent for zero-shot 3d understanding. In *European Conference on Computer Vision*, pages 186–202. Springer.

Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. 2025c. Llava-mini: Efficient image and video large multimodal models with one vision token. *arXiv preprint arXiv:2501.03895*.

Xiang Zhang, Zeyuan Chen, Fangyin Wei, and Zhuowen Tu. 2023a. Uni-3d: A universal model for panoptic 3d scene reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9256–9266.

Yiming Zhang, ZeMing Gong, and Angel X Chang. 2023b. Multi3drefer: Grounding text description to multiple 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15225–15236.

Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 2024. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*.

Duo Zheng, Shijia Huang, and Liwei Wang. 2025. Video-3d llm: Learning position-aware video representation for 3d scene understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8995–9006.

Duo Zheng, Shijia Huang, Lin Zhao, Yiwu Zhong, and Liwei Wang. 2024. Towards learning a generalist model for embodied navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13624–13634.

Hongyan Zhi, Peihao Chen, Junyan Li, Shuailei Ma, Xinyu Sun, Tianhang Xiang, Yinjie Lei, Mingkui Tan, and Chuang Gan. 2025. Lscenellm: Enhancing large 3d scene understanding using adaptive visual preferences. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3761–3771.

Yin Zhou and Oncel Tuzel. 2018. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4490–4499.

Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. 2024. Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. *arXiv preprint arXiv:2409.18125*.

Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 2023. 3d-vista: Pre-trained transformer for 3d vision and text alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2911–2921.

A Appendix

A.1 Multimodal Inputs Conversion

To ensure fair evaluation in different combinations of modality, we design a modality transformation pipeline that generates point cloud, RGB, andtext inputs while preserving key semantic information and minimizing information loss.

For *point cloud* generation, we project original multi-view RGB scans into 3D point clouds using pre-recorded camera poses and aligned depth images from 3D benchmarks like ScanNet (Dai et al., 2017) and 3RScan (Wald et al., 2019). Since point clouds originate from multi-view scans, they share the same semantics except for depth.

For *visual* input generation, we uniformly downsample the original RGB scans to 64 frames to reduce redundancy, which is consistent with the input format of the current video LLM (Zheng et al., 2025; Qi et al., 2025; Chen et al., 2024b). Besides, Yang *et al*. (Yang et al., 2024) suggests sequential visual inputs encode spatial semantics, aiding spatial reasoning. We use these RGB frames as the point cloud counterpart.

For *text* input generation, we follow SceneVerse (Jia et al., 2025) and generate object descriptions via image captioning. We first project object point clouds onto the 2D plane to identify their presence in RGB frames. Then, BLIP2 (Li et al., 2023) generates object captions, which GPT-4o (Achiam et al., 2023) refines into concise, structured descriptions. Since these descriptions originate from multi-view images, they capture both egocentric semantics and allocentric spatial relationships.

A.2 Attention Analysis

We adopt the attention measurement method in (Zhang et al., 2025c) shown in Eq. 4. The input prompt to the LLM contains different types of tokens, denoted as T_{sys}, T_{point_cloud}, $T_{instruction}$, and $T_{response}$ respectively. We quantify the attention between different tokens based on the attention map of the transformer layer in the LLM. Given an input token sequence of length L, each transformer layer outputs N attention maps S_i with shape $L \times L$ for N attention heads. The N attention maps are averaged across all heads to obtain an average attention map S. The attention weight from the ith token t_i to the jth token t_j is denoted as S_{ij}. For any two types of tokens T_A and $T_B \in \{T_{sys}, T_{point_cloud}, T_{instruction}, T_{response}\}$, the attention weight from T_A to T_B is given by:

$$\text{Attn}(A \to B) = \sum_{t_i \in T_A} \sum_{t_j \in T_B} S_{ij} \tag{4}$$

Particularly, $\sum_{t_i \in T_A} \sum_{t_j \in T_B} S_{ij}$ sums the all attention weights from T_A to T_B and $\sum_{t_i \in T_A} \mathbf{1}_{\sum_{t_j \in T_B} S_{ij} > 0}$ computes the counts of non-zero attention weight. Thus, Attn represents the average attention score from token A to token B. In this work, we focus on the attention from response tokens to point cloud tokens.

Relative Attention Due to the substantial variation in vanilla attention scores across different cases, directly averaging them may obscure case-specific attention patterns, leading to a distorted overall distribution that fails to accurately reflect the model's true attention behavior during spatial reasoning. Thus, we adopt *relative attention* (Zhang et al., 2025a) that computes the ratio of absolute attention scores to the per-layer average scores. We define $\text{Attn}^{(k)}(A \to B_i)$ as the attention score from T_A to j-th the token of T_B in the k-th layer, and $\text{Attn}^{(k)}(A \to \overline{B})$ as the average attention score from T_A to T_B in the k-th layer. The relative attention score is defined as:

$$\text{Attn}^{(k)}_{rel}(A \to B_j) = \frac{\text{Attn}^{(k)}(A \to B_j)}{\text{Attn}^{(k)}(A \to \overline{B})}, \tag{5}$$

$$\text{Attn}^{(k)}(A \to B_j) = \sum_{t_i \in T_A} S_{ij}, \tag{6}$$

$$\text{Attn}^{(k)}(A \to \overline{B}) = \frac{1}{N_B} \sum_{t_i \in T_A} \sum_{t_j \in T_B} S_{ij}, \tag{7}$$

where N_B is the token number of T_B.

Attention on Multimodal Inputs As depicted in Figure 9, we further analyze the impact of masking different proportions of point cloud tokens on spatial reasoning accuracy (left), and compare the response attention across different input modalities (right). The results show that for ScanQA, masking 90% of point cloud tokens does not significantly affect performance. However, when only 10% of the tokens remain, accuracy drops sharply by 24%, indicating that only a small subset of point cloud tokens plays a critical role in spatial understanding. Additionally, Figure 9(b) shows that the model's response exhibits significantly higher attention to textual inputs than to point cloud tokens, which aligns with similar findings in 2D LLMs (Zhang et al., 2025c).

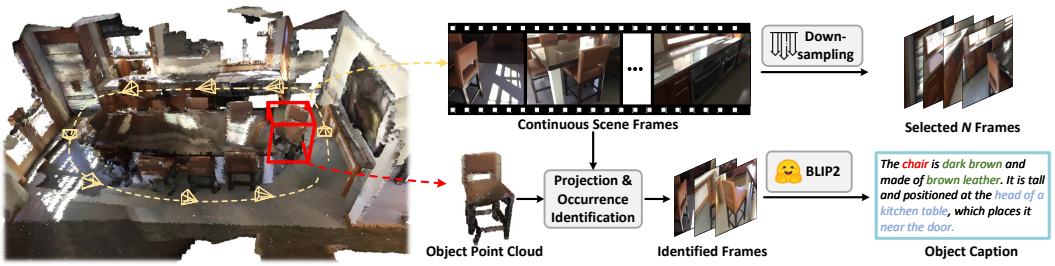

Figure 8: **The multi-modal data generation pipeline.** The 3D scene scan is projected into continuous scene frames which are further uniformly downsampled to N frames as the input for VLMs. We also select frames containing scene objects with object point clouds and leverage caption models to generate a text description of the scene as the input for LLMs.

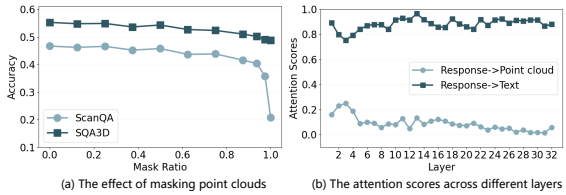

Figure 9: **Visualization of accuracy and attentions scores of LEO.** Left: the results of masking different ratios of point cloud tokens. Right: the attention scores on text and point cloud tokens.

A.3 Logit Lens for 3D LLMs

The logit lens is a mechanistic interpretability technique designed to understand what large language models "know" at intermediate layers. It interprets the model's internal representations by translating hidden states back into the model's output vocabulary space which is human-interpretable.

We exend the technique to 3D LLMs as depicted in Figure 10. In a 3D LLM, each layer produces a hidden representation of the current token or sequence. Normally, only the final layer is passed through the output projection and softmax to produce logits which then determine the next predicted word. The logit lens technique applies this to earlier layers by applying the same output projection matrix to intermediate activations. This effectively shows "what the model would predict if it stopped thinking here."

Formally, the logits of l-th layer is formulates as:

$$\text{logits}_l = W_U \cdot h_l, \tag{8}$$

where h_l is the hidden states at layer l and W_U is the output projection matrix. By examining these layer-wise logits, we can visualize how predictions evolve as the model processes and refines information.

Table 3: **Terms of different models and input modalities.**

Model	Input Modality	Term
3D LLM/VLM/LLM	Text-only	TI
3D LLM/VLM	Vision-only	VI
3D LLM/VLM	Vision-text	VTI
3D LLM	Point cloud-only	PI
3D LLM	Point cloud-vision	PVI
3D LLM	Point cloud-vision-text	PVTI

A.4 Evaluation Metrics

As illustrated in Figure 2, one QA in ScanQA corresponds to a spatial relationship that generates a quadruplet of QAs in RelSpatial QA, and we consider the model to understand the spatial relationship if it correctly answers all QAs in the quadruplet. Therefore, the accuracy of understanding spatial relationships is defined as follows:

$$\text{Accuracy} = \frac{N_{\text{correct}}^{\text{quad}}}{N_{\text{total}}^{\text{quad}}}, \tag{9}$$

Additionally, since the QA pairs in RelSpatialQA and ScanQA partially overlap, we assume that if a model truly understands a specific spatial relationship, it should be able to correctly answer the corresponding questions in both datasets. Therefore, we define the recall metric as follows to evaluate whether the model genuinely understands the spatial relationships:

$$\text{Recall} = \frac{N_{\text{correct}}^{\text{quad}}}{N_{\text{correct}}^{\text{ScanQA}}}. \tag{10}$$

A.5 Ablations of Different Modal Inputs

To investigate the effect of multimodal inputs on 3D LLMs, we conduct controlled experiments by altering the input modalities of multimodal models. Since scenes can be represented in text, vision, or 3D point clouds, our key idea is to replace one modality with another (e.g., substituting point

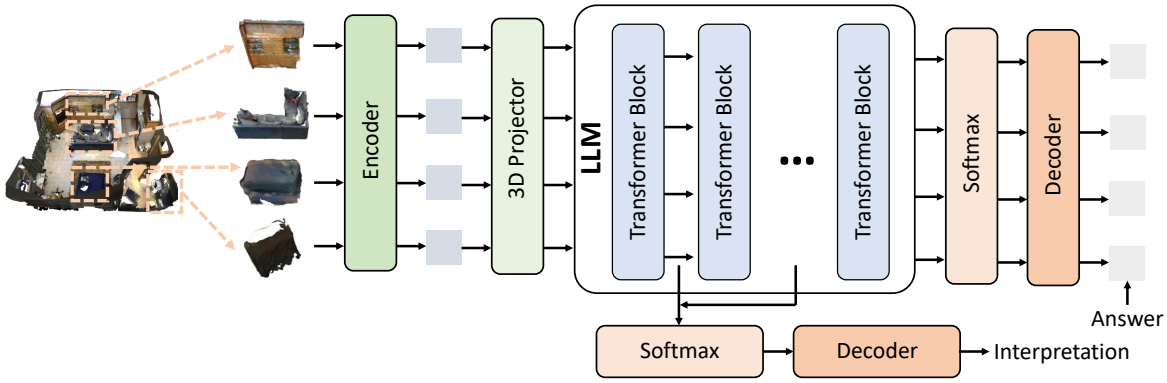

Figure 10: **The mechanism of logit lens**. The point cloud tokens are first encoded by a pretrained point cloud encoder and projected into token space via a 3D projector. After inference in the LLM backbone, the features are passed through a softmax layer and then decoded into the final answer. For simplicity, the text and visual inputs are omitted.

clouds with images or images with text). We design six modality combinations (Table 3) for LLMs, VLMs, and 3D LLMs: 1) TI (Text-only): scene descriptions as input; 2) VI (Vision-only): multi-view images of the scene; 3) VTI (Vision-Text): multi-view images and scene descriptions; 4) PI (Point cloud-only): only point clouds as input, RGB values in point clouds and additional images are removed.; 5) PVI (Point cloud-Vision): Matches the original 3D LLM input, including point clouds with either RGB values or multi-view images; 6) PVTI (Point cloud-Vision-Text): Extends PVI by adding scene descriptions. By default, PVI refers to the 3D LLM's original input.

We evaluate the performance of 3D LLMs on 3D QA tasks by feeding different modality combinations. For 3D LLMs that support visual input, such as LEO (Huang et al., 2023b) and 3D-LLM (Hong et al., 2023), we assess and compare their performance on 3D QA tasks using TQA-3D, VQA-3D, and PQA input. We assess 3D LLMs without visual input capabilities using only TQA-3D and PQA inputs. The results are depicted in Figure 11.

The impact of point clouds and vision on the 3D LLMs depends on their training approach. To validate the effect of point clouds and visual signals on spatial reasoning, we evaluate models with VI and PI on ScanQA and RelSpatialQA datasets by masking the corresponding modality. To obtain PI, we mask the RGB values for models with RGB point clouds as PVI (LL3DA, LEO) and mask image features for models with point clouds and multi-view images as PVI (Chat-Scene, 3D-LLM). To obtain VI, we mask the coordinate values of the point cloud. As depicted in Figure 11 (a) where each vertex on the radar plot denotes a refined EM@1 of a 3D LLM, an interesting result is that after masking the coordinate values of point clouds, the performance of 3D-LLM, Chat-Scene, and Chat3D-v2 have been merely affected, while the performance of LL3DA and LEO have dropped significantly. LL3DA has nearly zero EM@1 with VI. When the point clouds are masked, the Chat-Scene and 3D-LLM suffer a performance drop. Though LEO also has a performance drop with PI, it is not as much as that with VI. This result indicates that 3D LLMs with multi-view images input are more sensitive to visual modality while those with RGB point clouds are more affected by the point cloud modality.

Text-only input could lead to better performance than multi-modal input. We conducted an ablation study evaluating 3D LLM with TI to validate the necessity of multi-modal input for spatial reasoning. As depicted in Figure 11 (b), Chat-Scence and Chat-3D with VI outperform their PVI counterparts while LEO, LL3DA and 3D-LLMs lag behind their PVI performance. The results of Chat-Scene and Chat-3D demonstrate that the model solely relies on text input could even outperform multimodal input on 3D spatial reasoning tasks. In this study, point clouds fail to demonstrate their necessity in achieving the current SOTA performance on 3D spatial reasoning tasks. One possible explanation is that existing SOTA 3D LLMs with EM@1 < 50 can correctly answer fewer than half of the questions in 3D QA, and these questions can be answered merely with visual perception and commonsense reasoning (e.g., identifying the color of a wooden cabinet).

The redundancy in the text modality enhances

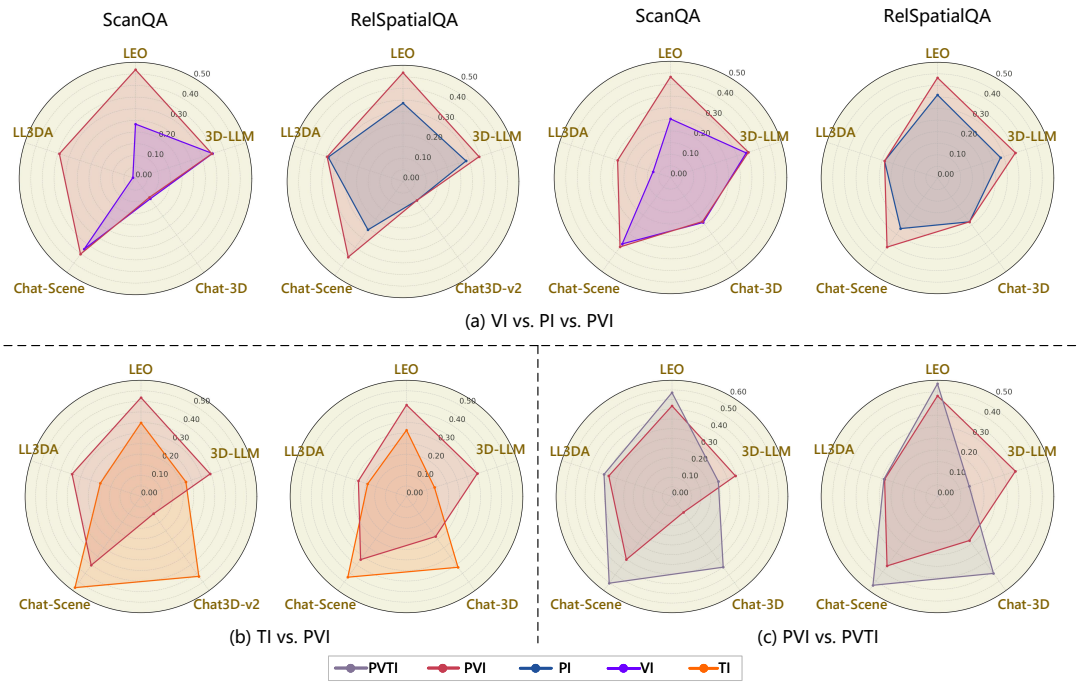

Figure 11: **Ablation studies with different modality inputs.**

the spatial reasoning performance. Current 3D LLMs primarily use point clouds and images as inputs. While these modalities contain sufficient information for spatial reasoning tasks, according to (Wang et al., 2025), VLMs are more effective at leveraging textual descriptions for spatial reasoning. To investigate whether this conclusion extends to 3D LLMs, we evaluate their EM@1 scores using both PVI and PVTI inputs on the ScanQA and RelSpatialQA datasets. The results in Figure 11 (c) clearly show that all 3D LLMs, except 3D-LLM, achieve higher EM@1 scores with PVTI than with PVI. This indicates that redundancy in the text modality enhances spatial reasoning performance, supporting the extension of this conclusion from VLMs to 3D LLMs.

For complete results of models' EM@1 with different combinations of input modalities, please refer to Table 4.

A.6 Data Annotation

We employed five student annotators for data refinement and compensated them at a rate of ¥0.5 per instance. Prior to annotation, we provided three refinement guidelines: (1) filter out irreversible spatial triplets and their corresponding QA pairs, (2) correct inaccurate triplets, and (3) revise grammatical errors in QA cases. After five days of annotation, a total of 5,523 high-quality QA samples were obtained.

Table 4: **EM@1 of different 3D LLMs with multimodal input combinations**.

	RelSpatialQA					ScanQA				
Modality	3D-LLM	LEO	LL3DA	ChatScene	Chat3D	3D-LLM	LEO	LL3DA	ChatScene	Chat3D
Mask All	0.273	0.217	0.079	0.288	0.239	0.305	0.208	0.012	0.294	0.108
TI	0.141	0.315	0.193	**0.473**	**0.414**	0.224	0.349	0.203	**0.555**	0.467
VI	0.342	0.252	0.045	0.353	0.239	**0.348**	0.234	0.012	0.376	0.108
VTI	0.141	0.335	0.234	0.418	**0.414**	0.248	0.381	0.203	0.518	**0.468**
PI	0.285	0.359	**0.243**	0.270	0.234	0.286	0.337	0.339	0.257	0.104
PTI	-	0.442	-	-	0.409	-	-	0.369	-	0.451
PVI	**0.352**	0.435	0.241	0.369	0.231	0.345	0.468	0.344	0.403	0.101
PVTI	0.143	**0.483**	0.241	0.472	0.412	0.252	**0.536**	**0.370**	0.553	0.456