Spatial Understanding from Videos: Structured Prompts Meet Simulation Data

Haoyu Zhang[1,2], **Meng Liu**[3,4*], **Zaijing Li**[1,2], **Haokun Wen**[1],
Weili Guan[1], **Yaowei Wang**[1,2], **Liqiang Nie**[1*]

[1]Harbin Institute of Technology (Shenzhen)   [2]Pengcheng Laboratory
[3]Shandong Jianzhu University   [4]Zhongguancun Academy

Abstract

Visual-spatial understanding, the ability to infer object relationships and layouts from visual input, is fundamental to downstream tasks such as robotic navigation and embodied interaction. However, existing methods face spatial uncertainty and data scarcity, limiting the 3D spatial reasoning capability of pre-trained vision-language models (VLMs). To address these challenges, we present a unified framework for enhancing 3D spatial reasoning in pre-trained VLMs without modifying their architecture. This framework combines SpatialMind, a structured prompting strategy that decomposes complex scenes and questions into interpretable reasoning steps, with ScanForgeQA, a scalable question-answering dataset built from diverse 3D simulation scenes through an automated construction process designed for fine-tuning. Extensive experiments across multiple benchmarks demonstrate the individual and combined effectiveness of our prompting and fine-tuning strategies, and yield insights that may inspire future research on visual-spatial understanding.

1 Introduction

Visual-spatial understanding, the ability to infer spatial relationships and the layout of objects from visual input, is a core component of human perception [1, 2, 3]. From a single image, human observers can intuitively estimate distances, relative sizes, and even infer occluded structures. As intelligent systems become increasingly embedded in real-world applications such as autonomous driving [4, 5, 6], robotic navigation [7, 8, 9], and augmented reality [10, 11, 12], it becomes crucial to endow models with similar spatial reasoning capabilities for robust perception and interaction.

Unfortunately, a single image is inherently limited in capturing the complexity of real-world 3D scenes, constraining its utility in practical scenarios [13, 14, 15]. To address this, point clouds have become a mainstream representation for 3D scene understanding due to their ability to encode rich geometric information [16, 17]. Yet, generating high-quality point clouds typically requires expensive sensors and incurs significant computational overhead, limiting scalability and accessibility.

These limitations motivate the pursuit of vision-only solutions that operate on scanning videos or multi-view images of scenes. Such approaches offer a more human-like and scalable pathway to spatial understanding [18]. However, performing 3D spatial reasoning from scanning videos presents two significant challenges: **(1) Spatial Uncertainty.** In the absence of explicit depth information, models must infer 3D structure from inherently limited 2D observations. This process is further complicated by occlusions, perspective distortions, and texture ambiguities, all of which introduce significant spatial uncertainty. Effectively addressing this challenge demands multi-step logical reasoning across frames to reconstruct coherent spatial layouts. **(2) Data Scarcity.** Existing datasets for this task are limited in both scale and diversity, restricting the ability of vision-language models

---

*Corresponding authors.

(VLMs) to acquire robust spatial knowledge and perceptual capabilities. Moreover, these datasets involve scans of real-world scenes, which leads to poor scalability. This highlights the need for scalable and extensible data sources to support effective spatial reasoning in VLMs.

To address these challenges, we propose a dual approach for enhancing 3D spatial reasoning in pre-trained VLMs, without modifying their underlying architecture. First, we introduce **SpatialMind**, a structured Chain-of-Thought (CoT) prompting strategy that guides VLMs through step-by-step reasoning over spatial relationships. Second, we present **ScanForgeQA**, a large-scale synthetic question-answering (QA) dataset constructed from diverse 3D simulation scenes using an automated generation pipeline. Fine-tuning VLMs on this dataset equips them with spatial commonsense knowledge, significantly improving their generalization to unseen spatial layouts. We have validated our approach through extensive experiments across multiple benchmarks. Results demonstrate the individual and combined effectiveness of our prompting and fine-tuning strategies, and yield insights that may inspire future research on visual-spatial understanding.

Our contributions are summarized as follows:

- We introduce SpatialMind, a spatial prompting strategy that decomposes spatial reasoning into structured steps, enabling pre-trained VLMs to perform multi-step inference over spatial relationships from visual input alone.

- We develop a scalable dataset generation pipeline to construct ScanForgeQA, a synthetic spatial question-answering dataset that enables VLMs to acquire spatial commonsense through fine-tuning.

- Experimental results validate the effectiveness and generalizability of both SpatialMind and ScanForgeQA, with their combination achieving further gains and providing valuable insights for future research.

2   Related Work

2D Image Spatial Understanding focuses on modeling spatial relationships among objects within the 2D image. Most existing models are trained on 2D images paired with textual descriptions, which offer limited cues about 3D structure. Consequently, their capacity for spatial reasoning remains constrained. To mitigate this, several approaches, such as SpatialVLM [13], SpatialRGPT [14], and SpatialBot [15], have been proposed. These methods enhance the spatial understanding by fine-tuning models on datasets specifically designed for spatially grounded QA tasks. To enable more comprehensive evaluation, recent studies [19, 20, 21, 22, 23] have introduced hierarchical benchmarks that assess models across varying levels of spatial reasoning complexity. Parallel efforts have explored more explicit forms of spatial interaction [24, 25]. For example, point-based methods [26, 27] interpret spatial instructions by predicting specific target points. Building on this trend, SpatialCoT [28] proposes a two-stage strategy that aligns multimodal inputs with spatial coordinates and incorporates CoT reasoning to better address complex embodied tasks. Despite these advancements, model performance often degrades in complex real-world 3D environments, highlighting the limitations of 2D-based approaches in representing complex 3D scenes.

3D Indoor Spatial Understanding focuses on enabling intelligent agents to identify object positions and infer their spatial relationships within enclosed environments, thereby supporting both object manipulation and interactive scene comprehension. Early 3D models are trained on standard indoor datasets [29, 30, 31, 32, 33, 34, 35] using point clouds to facilitate downstream tasks like 3D object detection and instance segmentation [36, 37, 38, 39] and primarily focus on object-level geometry and appearance features [40, 41, 42, 43]. More recent work extends this focus to complex indoor scenes, emphasizing inter-object spatial relationships and holistic scene-level understanding. To address challenges such as geometric complexity and annotation sparsity, many of these models employ cross-modal strategies that combine point cloud data with auxiliary multi-view 2D images [16, 17, 44]. Inspired by the way humans perceive spatial layouts through vision alone, emerging research [1, 45, 18] has begun to explore purely vision-based approaches to 3D spatial understanding. These methods rely solely on visual inputs, such as scanning videos, without requiring explicit 3D priors like point clouds. This line of work offers a more practical and scalable alternative for real-world deployment. In this context, we further investigate whether purely vision-based inputs can provide a more effective solution for indoor scene understanding.
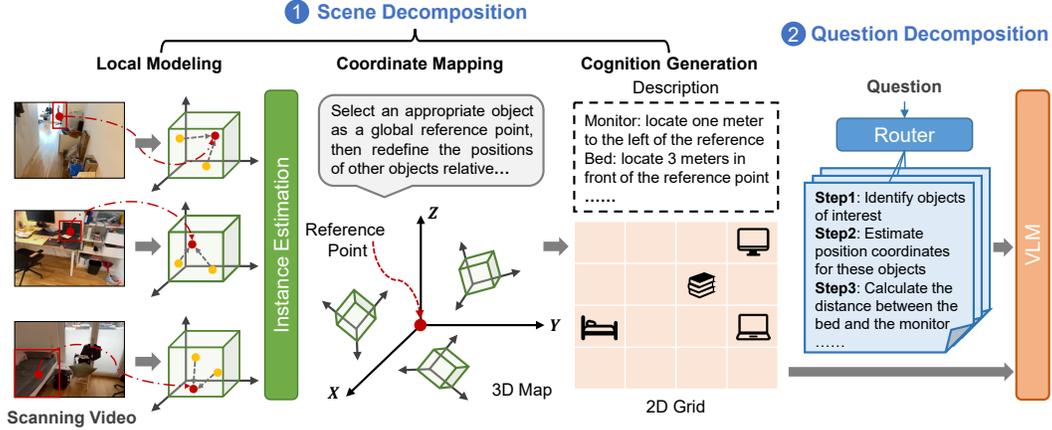
Figure 1: Illustration of our SpatailMind prompting strategy.

3 SpatialMind Prompting Strategy

As shown in Figure 1, our SpatialMind prompting strategy consists of two main components: **1) Scene Decomposition**, where the 3D scene depicted in the video is transformed into multiple different representations; and **2) Question Decomposition**, in which the question is broken down into a sequence of fine-grained reasoning steps. Further details can be found in **Appendix D**.

3.1 Scene Decomposition

The scene decomposition process includes three sequential steps: local modeling, coordinate mapping, and cognition generation.

Local Modeling. The first step processes scanning video frames to extract object instances and their relative spatial configurations within localized coordinate systems. To handle scene complexity and reduce the search space, we leverage GPT-4o[2] to identify all objects mentioned across the questions associated with a given scene, using them as candidate targets. For each frame i, we prompt VLMs to detect a subset of objects $\{c_{ij}\}$ from the candidate targets and estimate their positions $\mathbf{p}_{ij}^{\text{local}} \in \mathbb{R}^3$. These positions are defined relative to a randomly selected reference object (i.e., origin) within the same frame, forming a local 3D map:

$$\mathcal{L}_i = \left\{ (c_{ij}, \mathbf{p}_{ij}^{\text{local}}) \mid j = 1, \dots, n_i \right\}, \tag{1}$$

where n_i denotes the number of objects in frame i. Because each video frame captures only a limited field of view, the same object may appear across multiple frames from different perspectives. Thus, this step focuses on accurate per-frame object detection and spatial localization, laying the foundation for subsequent alignment in a global coordinate system.

Coordinate Mapping. To integrate spatial information across video frames, this step transforms all locally detected object positions into a unified global coordinate system. The global origin is defined by selecting the reference object in the first frame. To estimate motion between frames, we prompt the VLM to infer the relative rotation and translation between adjacent frames. These relative transformations are accumulated sequentially to compute each frame's transformation $\mathbf{T}_i$ with respect to the global coordinate system:

$$\mathbf{T}_i = \prod_{k=1}^{i} \begin{bmatrix} \mathbf{R}_{k,k-1} & \mathbf{t}_{k,k-1} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{2}$$

where $\mathbf{R}_{k,k-1}$ and $\mathbf{t}_{k,k-1}$ denote the relative rotation and translation from frame $k-1$ to frame k, respectively. This accumulated approach provides more stable and accurate alignment than directly estimating each frame's absolute pose. Using these transformations, each object's local coordinates are converted into global coordinates via homogeneous transformation:

$$\begin{bmatrix} \mathbf{p}_{ij}^{\text{global}} \\ 1 \end{bmatrix} = \mathbf{T}_i \cdot \begin{bmatrix} \mathbf{p}_{ij}^{\text{local}} \\ 1 \end{bmatrix}, \tag{3}$$

---

[2]`https://openai.com/index/hello-gpt-4o/`.

where $\mathbf{p}_{ij}^{\text{global}}$ denotes the global coordinates of the object j in the frame i. This step ensures that all detected objects across frames are positioned consistently within the same 3D space. Since objects may appear in multiple frames under different perspectives, we merge duplicate detections based on spatial proximity and semantic consistency via prompting. The result is a global 3D map of the scene:

$$\mathcal{G} = \left\{ \left(c_k, \mathbf{p}_k^{\text{global}} \right) \right\}_{k=1}^{N}, \tag{4}$$

where N is the total number of all object instances in the entire scene. This map serves as a unified spatial abstraction that captures the overall layout from egocentric scanning videos.

Cognition Generation. Beyond constructing a 3D map, we explore two additional formats for representing scene structure: a 2D spatial grid and natural language descriptions. We define a regular 2D grid over the global scene, typically aligned with the XY-plane. Each grid cell corresponds to a fixed real-world area (e.g., 1 meter per cell, denoted by cell size s). Each object c_k is mapped to a discrete grid location (i_k, j_k):

$$(i_k, j_k) = \left(\left\lfloor \frac{x_k}{s} \right\rfloor, \left\lfloor \frac{y_k}{s} \right\rfloor \right), \tag{5}$$

where (x_k, y_k) are the horizontal components of the object's global position $\mathbf{p}_k^{\text{global}}$. In parallel, we generate natural language descriptions of object locations relative to a designated reference point. Using prompting, the model produces statements such as {"monitor": "locate 1 meter to the left of the reference point"}. These descriptions serve as a human-interpretable form of spatial cognition, bridging visual perception and symbolic reasoning.

3.2 Question Decomposition

Different types of spatial questions require distinct reasoning strategies [46]. To accommodate this diversity, we first categorize questions into several types (e.g., object size, relative distance, and relative direction). For each category, we design a dedicated reasoning procedure using GPT-4o, followed by human verification to ensure correctness and interpretability. For instance, consider a question from the "relative distance" category: *Among the refrigerator, window, and microwave, which object is closest to the door?* The reasoning process for this type follows four structured steps: 1) Identify all mentioned objects, 2) Estimate the spatial coordinates of all relevant objects, 3) Compute the pairwise distances between the door and each candidate object, and 4) Select the object with the minimum distance as the answer. During inference, the system correspondingly selects the appropriate reasoning procedure based on the identified question type.

To perform 3D spatial reasoning, we feed the VLMs with the input scanning video, one form of scene representation (e.g., 3D map, 2D grid, or textual position descriptions), the question, and the corresponding step-by-step reasoning plan. To assess which scene representation format is most interpretable for VLMs, we have conducted comparative experiments, as shown in Figure 3.

4 ScanForgeQA Dataset Construction

The construction of the **ScanForgeQA** dataset involves a three-stage pipeline, illustrated in Figure 2. These stages are: **1) Scene Construction**, where single-room 3D environments are created; **2) Scan Creation**, in which egocentric videos are simulated by scanning through the constructed scenes; and **3) QA Generation**, where textual question-answering pairs are automatically generated based on object annotations and the spatial layout of each scene.

4.1 Scene Construction

To ensure diversity and richness in single-room scene collection, we adopt two parallel strategies:

Separation. We modify existing scene datasets to leverage available resources effectively. Specifically, we utilize the 3D-FRONT dataset [47], which contains 6,813 multi-room scenes furnished with diverse 3D objects and annotated with detailed layout semantics and high-quality textures. Since our focus is on single-room environments, we disassemble each multi-room scene into individual rooms. For each scene, we isolate and load one room at a time, along with its corresponding ceiling and walls, and save it as an independent instance. This disassembly process yields 44,427 single-room scenes. We further filter out uncommon room types (e.g., garage, auditorium) and those lacking sufficient
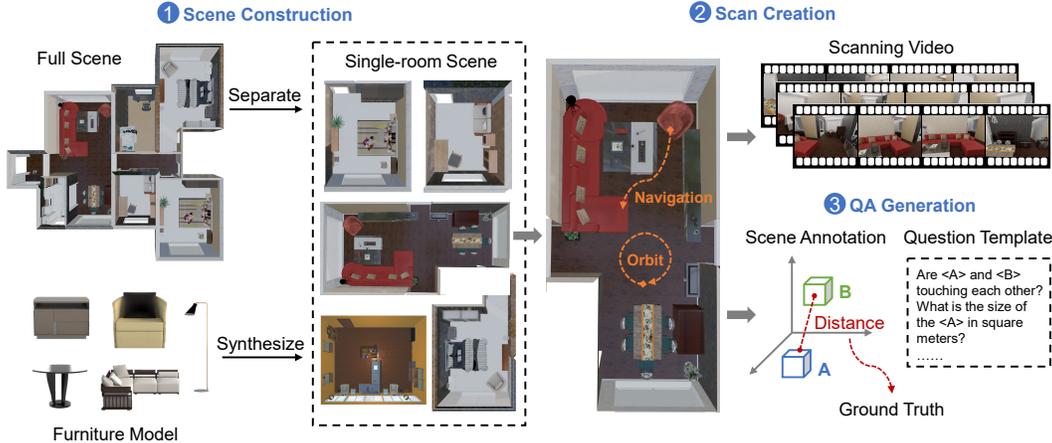
Figure 2: The pipeline of ScanForgeQA data construction.

object content (e.g., aisle, stairwell). The final dataset consists of 34,116 single-room scenes across six common categories: bedroom, kitchen, bathroom, living room, dining room, and storage room.

Synthesis. To introduce additional diversity and originality, we synthesize novel room layouts using a LLM-guided generation approach. Specifically, we adopt HoloDeck [48], a 3D generation framework that leverages LLMs to parse natural language prompts, retrieve matching assets from large-scale 3D object repositories such as Objaverse [49], and optimize their spatial arrangement to form semantically meaningful scenes. To drive the generation process, we first use GPT-4o to create diverse textual descriptions for various room types. For example, a bedroom may be described as: *"A bedroom with a bed, window, armchair, and wardrobe"*. We define eight room categories, including two additional types—office and store—and generate 20 distinct descriptions for each. These prompts are fed into HoloDeck to produce corresponding room layouts, with human verification to ensure spatial plausibility and realism. This synthesis process yields 160 additional single-room scenes.

4.2 Scan Creation

To simulate egocentric scanning videos from the constructed single-room scenes, we implement a scanning procedure using the Unity engine[3]. Each scene is scanned using two complementary strategies designed to emulate natural human visual exploration:

Orbit Scan. We define a circular trajectory centered in the room at a height of approximately 1.5 meters, corresponding to typical adult eye level. The circle's diameter is set to two-thirds of the shorter side of the room. The camera is randomly initialized at a point on this path and moves along the circle either clockwise or counterclockwise. An image is captured every 5 degrees of rotation, resulting in 72 frames per orbit scan. This strategy provides a comprehensive 360-degree panoramic view of the scene.

Navigation Scan. To simulate movement through the environment, we label navigable ground regions based on object categories and generate a navigation mesh using the *NavMesh Baking* API. We randomly select two objects as the navigation start and end points and compute the shortest path between them on the mesh. Among the candidate paths, the two longest are chosen for scanning to achieve a more complete coverage of the scene. For each path, the camera first performs a 360-degree rotation at the starting point, capturing an image every 12 degrees (30 images total). It then traverses the path toward the destination, during which 12 frames are uniformly sampled. Upon arrival, another 360-degree rotation is performed, again capturing 30 images. In total, 72 frames are recorded per path. Due to the limited size of indoor environments, rotational movement yields more visual variation than translation; hence, fewer frames are captured during motion.

Table 1: Comparison of 3D QA Datasets

Dataset	Source	Scenes	Format	QAs	Scalability
SPARTUN3D	3RScan	478	Point Cloud	133K	Hard
MSQA	ScanNet, 3RScan, ARKitScenes	1.7K	Point Cloud	251K	Hard
3D-LLM	ScanNet, HM3D	1.2K	Point Cloud	300K	Hard
ScanForgeQA	Simulation/Synthesis	34K	Scan Video	925K	Easy

4.3 QA Generation

To generate diverse supervised fine-tuning (SFT) data and enhance the 3D spatial reasoning capabilities of existing VLMs, we define three categories of question types: attribute estimation, spatial reasoning, and hypothesis analysis. These categories encompass both quantitative and qualitative dimensions, and cover both open-set and closed-set scenarios. Below, we describe each category in detail, along with the methodology for deriving corresponding ground-truth answers.

Attribute Estimation. This type focuses on static properties of objects and scenes, such as *object count* ("How many chairs are in the room?"), *object size* ("What is the length of the longest side of the refrigerator in meters?"), *room size* ("What is the size of this room in square meters?"), and *room type* ("Based on the object layout, what is the most likely type of room (e.g., kitchen)?"). Ground-truth answers for these questions are directly derived from 3D scene annotations and object metadata provided in the dataset.

Spatial Reasoning. This category targets inter-object spatial relationships, requiring models to infer positional and geometric properties such as distance, orientation, and contact. Representative question types include: *relative distance* ("Which of these objects (refrigerator, couch, ceiling light) is closest to the TV?"), *absolute distance* ("What is the distance between the couch and the table in meters?"), *relative direction* ("If I am standing by sofa and facing the table, which side is the trash can on?"), and *contact relationship* ("Is there a gap between the bed and the headboard?"). For distance-related questions, we compute Euclidean distances between object centroids in the global 3D coordinate space. For contact relationships, object dimensions are also considered to determine physical adjacency. To resolve relative direction,

Figure 3: Effects of different scene expression.

we define an object's front as the side oriented toward the room center. Angular sectors are divided clockwise into four directional categories: right (45°–135°), back (135°–225°), left (225°–315°), and front (315°–45°). For example, an object located at 80° relative to the reference point is classified as being on the right.

Hypothesis Analysis. This category introduces conditional reasoning under hypothetical scenarios, often requiring geometric and commonsense inference. A typical example is *operation feasibility* ("Considering only object dimensions, is it feasible to place the television on the table?"). Feasibility is determined by comparing object dimensions. For stacking, the movable object's length and width must be smaller than those of the supporting surface. For embedding (e.g., fitting an item into a drawer), the object's height must also fall within the bounds of the specific container's volume.

A comparison with existing 3D QA datasets (e.g., SPARTUN3D [50], MSQA [51], and 3D-LLM [17]) is presented in Table 1. The full ScanForgeQA dataset includes 34,276 single-room scenes, 103K simulated video scans, and 925K question-answering pairs for training. Leveraging synthetic environments allows scalable and controlled data generation across diverse spatial scenarios. Additional implementation details are provided in the **Appendix C**.
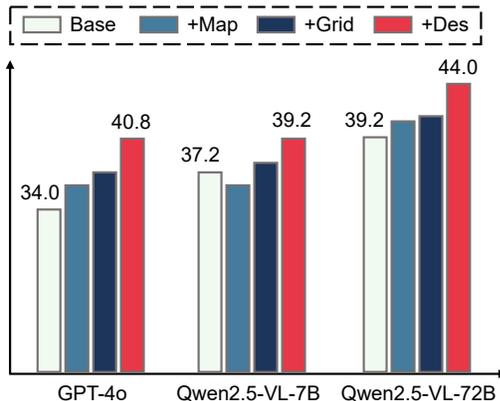
Table 2: Performance comparison on VSI-Bench. † indicates results on VSI-Bench (tiny) set.

Method	Obj. Count	Abs. Dist.	Obj. Size	Room Size	Rel. Dist.	Rel. Dir.	Route Plan	Appr. Order	Avg	Δ
Close-source										
Human Level†	94.3	47.0	60.4	45.9	94.7	95.8	95.8	100.0	79.2	-
Gemini-1.5 Pro†	49.6	28.8	58.6	49.4	46.0	48.1	42.0	68.0	48.8	-
Gemini-1.5 Pro	56.2	30.9	64.1	43.6	51.3	46.3	36.0	34.6	45.4	-
+SpatialMind	63.9	51.8	70.2	47.3	56.3	45.9	42.6	44.3	52.8	↑ **7.4%**
GPT-4o	46.2	5.3	43.8	38.2	37.0	41.3	31.5	28.5	34.0	-
+SpatialMind	40.0	27.1	62.7	40.9	41.0	39.6	37.1	38.5	40.8	↑ **6.8%**
Open-source										
Struct2D	46.0	34.7	56.4	42.6	35.1	44.9	33.5	-	41.9	-
InternVL2-8B	23.1	28.7	48.2	39.8	36.7	30.7	29.9	39.6	34.6	-
+SpatialMind	35.8	28.9	49.7	44.4	37.2	34.8	35.1	45.5	38.9	↑ **4.3%**
+ScanForgeQA	45.3	33.4	54.8	45.0	41.1	36.1	33.4	43.0	41.5	↑ **6.9%**
+Both	47.0	32.8	53.2	46.6	39.8	36.8	37.9	47.5	42.7	↑ **8.1%**
InternVL2-40B	34.9	26.9	46.5	31.8	42.1	32.2	34.0	39.6	36.0	-
+SpatialMind	36.4	30.0	49.1	41.8	43.8	36.1	35.6	50.0	40.4	↑ **4.4%**
+ScanForgeQA	51.0	29.2	52.7	38.1	47.2	36.4	35.9	47.6	42.3	↑ **6.3%**
+Both	52.2	30.5	54.4	41.0	50.5	37.0	40.2	50.3	44.5	↑ **8.5%**
Qwen2.5-VL-7B	40.3	22.2	50.1	38.9	38.0	40.7	31.4	35.9	37.2	-
+SpatialMind	45.1	25.2	52.1	41.4	38.7	41.6	34.7	34.5	39.2	↑ **2.0%**
+ScanForgeQA	53.2	30.5	56.8	44.9	42.3	44.0	37.3	37.7	43.3	↑ **6.1%**
+Both	55.0	29.5	57.3	44.0	43.5	44.3	38.3	39.2	43.9	↑ **6.7%**
Qwen2.5-VL-72B	37.9	28.6	57.4	49.8	45.5	38.4	20.6	35.4	39.2	-
+SpatialMind	42.3	32.0	61.7	53.8	48.2	43.9	30.4	39.3	44.0	↑ **4.8%**
+ScanForgeQA	45.2	32.7	63.3	52.4	50.1	41.7	32.8	40.2	44.8	↑ **5.6%**
+Both	48.6	34.4	68.9	54.7	53.4	43.9	30.1	42.7	47.1	↑ **7.9%**

Table 3: Performance comparison on the EM-EQA subset of OpenEQA and the validation set of ScanQA and SQA3D.

Method	OpenEQA Acc/Score	ScanQA BLEU-1	SQA3D EM-1
Qwen2.5-VL-7B	50.1/3.1	32.5	17.2
+SpatialMind	53.7/3.2	33.1	19.8
+ScanForgeQA	56.2/3.3	34.8	23.3
+Both	58.6/3.5	37.9	24.5
Qwen2.5-VL-72B	53.8/3.2	35.4	34.8
+SpatialMind	55.7/3.2	38.0	39.2
+ScanForgeQA	59.1/3.4	42.5	43.0
+Both	60.4/3.4	44.1	46.3

Table 4: Effects of different fine-tuning data and prompting strategy.

Method	Room Size	Avg
Qwen2.5-VL-7B	38.9	37.2
+SQA3D	38.8	38.9
+ScanQA	38.5	39.1
+ScanForgeQA	44.9	43.3
Qwen2.5-VL-72B	49.8	39.2
+CoT-Question	50.6	41.3
+CoT-Scene	52.1	42.7
+SpatialMind	53.8	44.0

5   Experiments

The experimental settings (including benchmarks, baselines, etc.) and more experimental results can be found in the **Appendix A** and **B**.

5.1   Performace Comparison

We investigated the following five key questions to assess our approach:

Q1: Which scene representation format is most interpretable by VLMs? Figure 3 presents a performance comparison across different representation formats: no additional spatial context (Base), inclusion of a 3D map (+Map), a 2D grid (+Grid), and object-centric textual descriptions (+Des). Across all models, a consistent trend emerges: the +Des variant outperforms others, followed by
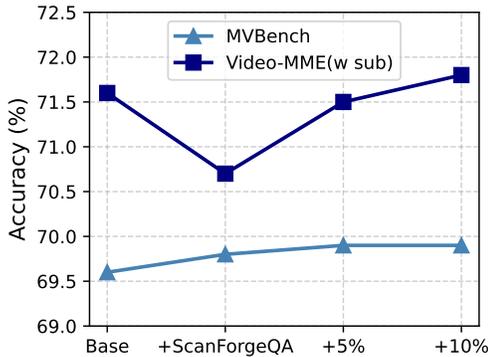
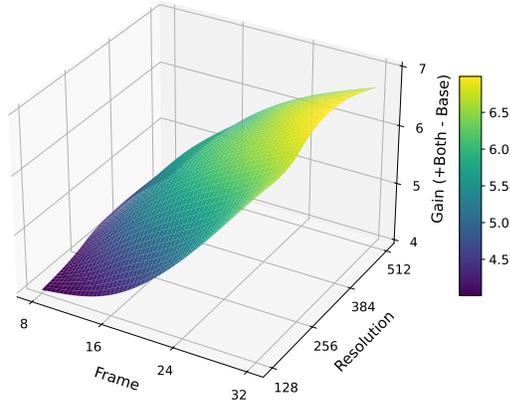Figure 4: Performance of Qwen2.5-VL-7B on MVBench and Video-MME.

Figure 5: Ablation study of Qwen2.5-VL-7B under varying numbers of frames and resolution.

+Grid, while +Map yields the least improvement. These results suggest that current VLMs are more adept at interpreting one-dimensional textual descriptions than high-dimensional structured spatial formats. Consequently, we adopted the textual description format in subsequent experiments as the default scene representation.

How do SpatialMind and ScanForgeQA impact VLM performance? As shown in Table 2, we progressively applied the SpatialMind prompting strategy and the ScanForgeQA fine-tuning data across a range of VLMs, varying in architectures, parameter size, and openness (including both open- and closed-source models). The results reveal three key findings: 1) Both SpatialMind prompting and ScanForgeQA fine-tuning consistently improve visual-spatial understanding across models. This includes large-scale proprietary models such as Gemini-1.5 Pro [4] and GPT-4o, demonstrating the effectiveness and generalizability of our approaches. 2) Model size affects the relative benefit of prompting versus fine-tuning. Larger models (e.g., 72B) benefit more from prompting, which enhances their reasoning capabilities, while smaller models (e.g., 7B) show greater improvements through fine-tuning. For instance, Qwen2.5-VL-7B gains 6.1% from fine-tuning, compared to only 2.0% from prompting. 3) Humans and VLMs exhibit complementary strengths. Human participants excel in qualitative tasks (e.g., achieving 100% accuracy on the *Appearance Order* task) but perform poorly on precise quantitative estimations (e.g., *Object Size*). In contrast, VLMs show strong quantitative reasoning ability and, in some cases, even surpass human-level performance. This contrast underscores the potential of VLMs to complement human perception in spatial tasks.

Can combining prompting and fine-tuning yield further gains? To assess whether SpatialMind and ScanForgeQA provide complementary benefits, we applied the SpatialMind prompting strategy to models that have already been fine-tuned on the ScanForgeQA dataset. The results, reported in the "+Both" row of Table 2, show consistent performance improvements across all evaluated models. These findings confirm that the two approaches are complementary.

Does the improvement generalize to other spatial benchmarks? To assess the generalizability of our framework, we conducted evaluations on multiple benchmarks, including OpenEQA [52], ScanQA [53], and SQA3D [54]. As shown in Table 3, both SpatialMind prompting and ScanForgeQA fine-tuning lead to consistent performance gains across all benchmarks. These results validate the robustness of our approach and confirm its applicability across diverse spatial tasks and datasets.

Does fine-tuning affect performance on other tasks? To investigate whether enhancing visual-spatial capabilities via fine-tuning adversely impacts a model's general performance, we conducted evaluations on MVBench [55] and Video-MME [56], two broad multi-task video benchmarks. As shown in Figure 4, fine-tuning with ScanForgeQA slightly improves performance on MVBench but leads to a marginal drop on Video-MME. This difference likely stems from MVBench containing spatial reasoning tasks, while Video-MME focuses more on event comprehension. To mitigate this trade-off, we further experimented with mixed fine-tuning, combining a small proportion (5% and 10%) of traditional data from ShareGPT4Video [57] with ScanForgeQA. Results show that

---

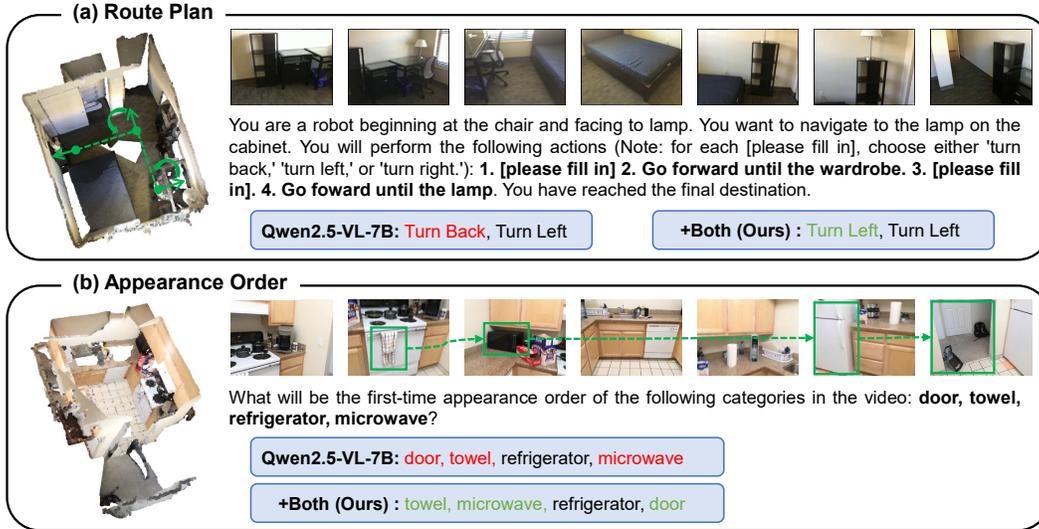[4] https://deepmind.google/technologies/gemini/pro/.

8

Figure 6: Two examples from VSI-Bench comparing predictions from Qwen2.5-VL-7B and Ours.

this strategy achieves improved performance, surpassing the original Qwen2.5-VL-7B baseline, suggesting that spatial fine-tuning can be harmonized with broader capabilities through data balancing.

5.2 Ablation Study

In this section, we explored the impact of various design choices, including prompting strategies, fine-tuning datasets, frame sampling strategies, and input resolution, on the performance of VLMs.

On prompting strategy. To isolate the contributions of each component in the SpatialMind prompting strategy, we evaluated two variants: one containing only the question component (CoT-Question) and another containing only the scene description (CoT-Scene). As shown in Table 4, both variants independently improve spatial reasoning performance, but are less effective than the full combined prompt. Notably, the scene description contributes more significantly to model performance than the reasoning steps, suggesting its central role in facilitating spatial understanding.

On fine-tuning data. To investigate the effectiveness of our proposed ScanForgeQA against existing spatial datasets, we fine-tuned Qwen2.5-VL-7B on SQA3D [54] and ScanQA [53]. As shown in Table 4, fine-tuning on either of these datasets results in lower performance compared to ScanForgeQA, and even reduces accuracy on tasks involving precise spatial estimation (e.g., *Room Size*). This is primarily due to the limited presence of fine-grained spatial estimation samples in the existing datasets. Importantly, both datasets and the VSI-Bench benchmark originate from the same source (i.e., ScanNet [31]), resulting in minimal data discrepancy. This contrast emphasizes the advantage of our simulated data generation pipeline.

On frames and resolution. To evaluate the robustness of our approach, we analyzed performance sensitivity to the number of input frames and image resolution. Figure 5 visualizes the performance of the +Both variant and the baseline Qwen2.5-VL-7B under various configurations. Our method consistently outperforms the baseline across all settings, with performance further improving as the number of frames and resolution increase. This indicates that our approach remains stable and effective under varying visual input conditions.

5.3 Qualitative Analysis

In Figure 6, we presented two illustrative examples from VSI-Bench, comparing predictions from the baseline Qwen2.5-VL-7B and our enhanced variant (+Both). In Case (a), Qwen2.5-VL-7B fails to produce the correct directional prediction, likely due to its limited capacity for 3D spatial reasoning. In contrast, our method successfully identifies the correct answer. Case (b) involves a simpler spatial reasoning task, however, Qwen2.5-VL-7B still fails, potentially due to insufficient object localization.

Our enhanced variant, benefiting from both structured prompting and spatially grounded fine-tuning, demonstrates notable improvements in accuracy and reasoning robustness.

6 Conclusion

In this work, we present an effective framework for enhancing visual-spatial reasoning in VLMs without modifying their underlying architecture. This makes our approach readily adaptable across models of varying scales and types. By integrating the structured prompting strategy (SpatialMind) with an automatically constructed dataset (ScanForgeQA), we enable VLMs to more effectively interpret and reason about 3D spatial relationships in complex visual scenes. Extensive evaluations across multiple spatial reasoning benchmarks demonstrate that our framework consistently improves accuracy, robustness, and generalization. Furthermore, our analysis reveals that prompting and fine-tuning play complementary roles in advancing visual-spatial understanding.

References

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

[2] Yizhang Jin, Jian Li, Jiangning Zhang, Jianlong Hu, Zhenye Gan, Xin Tan, Yong Liu, Yabiao Wang, Chengjie Wang, and Lizhuang Ma. Llava-vsd: Large language-and-vision assistant for visual spatial description. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 11420–11425, 2024.

[3] Zaijing Li, Yuquan Xie, Rui Shao, Gongwei Chen, Dongmei Jiang, and Liqiang Nie. Optimus-2: Multimodal minecraft agent with goal-observation-action conditioned policy. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9039–9049, 2025.

[4] Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, XianPeng Lang, and Hang Zhao. Drivevlm: The convergence of autonomous driving and large vision-language models. In *8th Annual Conference on Robot Learning*.

[5] Kun Wang, Hao Liu, Lirong Jie, Zixu Li, Yupeng Hu, and Liqiang Nie. Explicit granularity and implicit scale correspondence learning for point-supervised video moment localization. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 9214–9223, 2024.

[6] Haoyu Zhang, Meng Liu, Yuhong Li, Ming Yan, Zan Gao, Xiaojun Chang, and Liqiang Nie. Attribute-guided collaborative learning for partial person re-identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):14144–14160, 2023.

[7] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: an embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, pages 8469–8488, 2023.

[8] Haoyu Zhang, Meng Liu, Zixin Liu, Xuemeng Song, Yaowei Wang, and Liqiang Nie. Multi-factor adaptive vision selection for egocentric video question answering. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pages 59310–59328, 2024.

[9] Zaijing Li, Yuquan Xie, Rui Shao, Gongwei Chen, Dongmei Jiang, and Liqiang Nie. Optimus-1: Hybrid multimodal memory empowered agents excel in long-horizon tasks. In *Advances in Neural Information Processing Systems*, volume 37, pages 49881–49913, 2024.

[10] Keshigeyan Chandrasegaran, Agrim Gupta, Lea M Hadzic, Taran Kota, Jimming He, Cristóbal Eyzaguirre, Zane Durante, Manling Li, Jiajun Wu, and Fei-Fei Li. Hourvideo: 1-hour video-language understanding. *Advances in Neural Information Processing Systems*, 37:53168–53197, 2024.

[11] Haoyu Zhang, Qiaohui Chu, Meng Liu, Yunxiao Wang, Bin Wen, Fan Yang, Tingting Gao, Di Zhang, Yaowei Wang, and Liqiang Nie. Exo2ego: Exocentric knowledge guided mllm for egocentric video understanding. *arXiv preprint arXiv:2503.09143*, 2025.

[12] Yisen Feng, Haoyu Zhang, Meng Liu, Weili Guan, and Liqiang Nie. Object-shot enhanced grounding network for egocentric video. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24190–24200, 2025.

[13] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14455–14465, 2024.

[14] An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. Spatialrgpt: Grounded spatial reasoning in vision-language models. *Advances in Neural Information Processing Systems*, 37:135062–135093, 2025.

[15] Wenxiao Cai, Iaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. Spatialbot: Precise spatial understanding with vision language models. *arXiv preprint arXiv:2406.13642*, 2024.

[16] Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. 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, 2024.

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

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

[19] Mengfei Du, Binhao Wu, Zejun Li, Xuan-Jing Huang, and Zhongyu Wei. Embspatial-bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 346–355, 2024.

[20] Wenyu Zhang, Wei En Ng, Lixin Ma, Yuwen Wang, Jungqi Zhao, Allison Koenecke, Boyang Li, and Lu Wang. Sphere: Unveiling spatial blind spots in vision-language models through hierarchical evaluation. *arXiv preprint arXiv:2412.12693*, 2024.

[21] Jianlong Wu, Wei Liu, Ye Liu, Meng Liu, Liqiang Nie, Zhouchen Lin, and Chang Wen Chen. A survey on video temporal grounding with multimodal large language model. *arXiv preprint arXiv:2508.10922*, 2025.

[22] Meng Liu, Xiang Wang, Liqiang Nie, Xiangnan He, Baoquan Chen, and Tat-Seng Chua. Attentive moment retrieval in videos. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 15–24, 2018.

[23] Zaijing Li, Yuquan Xie, Rui Shao, Gongwei Chen, Weili Guan, Dongmei Jiang, and Liqiang Nie. Optimus-3: Towards generalist multimodal minecraft agents with scalable task experts. *arXiv preprint arXiv:2506.10357*, 2025.

[24] Chenyang Ma, Kai Lu, Ta-Ying Cheng, Niki Trigoni, and Andrew Markham. Spatialpin: Enhancing spatial reasoning capabilities of vision-language models through prompting and interacting 3d priors. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

[25] Jun Yu, Yunxiang Zhang, Zerui Zhang, Zhao Yang, Gongpeng Zhao, Fengzhao Sun, Fanrui Zhang, Qingsong Liu, Jianqing Sun, Jiaen Liang, et al. Rag-guided large language models for visual spatial description with adaptive hallucination corrector. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 11407–11413, 2024.

[26] Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction in robotics. In *8th Annual Conference on Robot Learning*.

[27] Chan Hee Song, Valts Blukis, Jonathan Tremblay, Stephen Tyree, Yu Su, and Stan Birchfield. Robospatial: Teaching spatial understanding to 2d and 3d vision-language models for robotics. *arXiv preprint arXiv:2411.16537*, 2024.

[28] Yuecheng Liu, Dafeng Chi, Shiguang Wu, Zhanguang Zhang, Yaochen Hu, Lingfeng Zhang, Yingxue Zhang, Shuang Wu, Tongtong Cao, Guowei Huang, et al. Spatialcot: Advancing spatial reasoning through coordinate alignment and chain-of-thought for embodied task planning. *arXiv preprint arXiv:2501.10074*, 2025.

[29] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Yuri Feigin, Peter Fu, Thomas Gebauer, Daniel Kurz, Tal Dimry, Brandon Joffe, Arik Schwartz, et al. Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

[30] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *International Conference on 3D Vision (3DV)*, 2017.

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

[32] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Procthor: Large-scale embodied ai using procedural generation. *Advances in Neural Information Processing Systems*, 35:5982–5994, 2022.

[33] Yongsen Mao, Yiming Zhang, Hanxiao Jiang, Angel Chang, and Manolis Savva. Multiscan: Scalable rgbd scanning for 3d environments with articulated objects. *Advances in neural information processing systems*, 35:9058–9071, 2022.

[34] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

[35] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12–22, 2023.

[36] Thang Vu, Kookhoi Kim, Tung M Luu, Thanh Nguyen, and Chang D Yoo. Softgroup for 3d instance segmentation on point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2708–2717, 2022.

[37] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler faster stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4840–4851, 2024.

[38] Phuc Nguyen, Tuan Duc Ngo, Evangelos Kalogerakis, Chuang Gan, Anh Tran, Cuong Pham, and Khoi Nguyen. Open3dis: Open-vocabulary 3d instance segmentation with 2d mask guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4018–4028, 2024.

[39] David Rozenberszki, Or Litany, and Angela Dai. Unscene3d: Unsupervised 3d instance segmentation for indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19957–19967, 2024.

[40] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, et al. Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. *arXiv preprint arXiv:2309.00615*, 2023.

[41] Zekun Qi, Runpei Dong, Shaochen Zhang, Haoran Geng, Chunrui Han, Zheng Ge, Li Yi, and Kaisheng Ma. Shapellm: Universal 3d object understanding for embodied interaction. In *European Conference on Computer Vision*, pages 214–238. Springer, 2024.

[42] Zhe Liu, Jinghua Hou, Xinyu Wang, Xiaoqing Ye, Jingdong Wang, Hengshuang Zhao, and Xiang Bai. Lion: Linear group rnn for 3d object detection in point clouds. *Advances in Neural Information Processing Systems*, 37:13601–13626, 2024.

[43] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *European Conference on Computer Vision*, pages 131–147. Springer, 2024.

[44] Yunze Man, Shuhong Zheng, Zhipeng Bao, Martial Hebert, Liangyan Gui, and Yu-Xiong Wang. Lexicon3d: Probing visual foundation models for complex 3d scene understanding. *Advances in Neural Information Processing Systems*, 37:76819–76847, 2024.

[45] Zhenyi Liao, Qingsong Xie, Yanhao Zhang, Zijian Kong, Haonan Lu, Zhenyu Yang, and Zhijie Deng. Improved visual-spatial reasoning via r1-zero-like training. *arXiv preprint arXiv:2504.00883*, 2025.

[46] Haoyu Zhang, Meng Liu, Zan Gao, Xiaoqiang Lei, Yinglong Wang, and Liqiang Nie. Multimodal dialog system: Relational graph-based context-aware question understanding. In *Proceedings of the 29th ACM international conference on multimedia*, pages 695–703, 2021.

[47] Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, et al. 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10933–10942, 2021.

[48] Yue Yang, Fan-Yun Sun, Luca Weihs, Eli VanderBilt, Alvaro Herrasti, Winson Han, Jiajun Wu, Nick Haber, Ranjay Krishna, Lingjie Liu, et al. Holodeck: Language guided generation of 3d embodied ai environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16227–16237, 2024.

[49] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13142–13153, 2023.

[50] Yue Zhang, Zhiyang Xu, Ying Shen, Parisa Kordjamshidi, and Lifu Huang. Spartun3d: Situated spatial understanding of 3d world in large language models. *arXiv preprint arXiv:2410.03878*, 2024.

[51] Xiongkun Linghu, Jiangyong Huang, Xuesong Niu, Xiaojian Shawn Ma, Baoxiong Jia, and Siyuan Huang. Multi-modal situated reasoning in 3d scenes. *Advances in Neural Information Processing Systems*, 37:140903–140936, 2024.

[52] Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, et al. Openeqa: Embodied question answering in the era of foundation models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16488–16498, 2024.

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

[54] Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. Sqa3d: Situated question answering in 3d scenes. In *The Eleventh International Conference on Learning Representations*.

[55] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22195–22206, 2024.

[56] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024.

[57] Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Zhenyu Tang, Li Yuan, et al. Sharegpt4video: Improving video understanding and generation with better captions. *Advances in Neural Information Processing Systems*, 37:19472–19495, 2024.

[58] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

[59] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

[60] Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. Videollama 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025.

[61] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[62] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[63] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our abstract and introduction clearly state four main contributions made in the paper. Our experimental results also support our claims.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have a dedicated section to discuss the limitations of our work. Please refer to Appendix E.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: Our paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We have included detailed information on the methodology (Appendix D), data construction (Appendix C), and experimental settings (Appendix A).

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have provided instructions on data access and preparation. Code files for prompting can be found in the Supplementary Material. The data pipeline, data, and model weights will be publicly available upon paper publication

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We provide detailed experimental settings in Appendix A.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: : Performing multiple repetitive experiments in order to compute error bars is labor-intensive and has significant overhead for the large language model development.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the computational resources needed in order to reproduce the experiments in Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research conforms in every respect with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have a dedicated section to discuss the societal impacts of our work. Please refer to Appendix F.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our research builds upon existing datasets rather than creating new ones from the internet. So our work does not pose such risks to the best of our knowledge.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All datasets used in our work are commonly used datasets with open access. We have adhered to their licenses and provided citations to give them credit.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.

19

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: The data pipeline, data, model weights, and benchmark will be publicly available upon paper publication.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our research does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our research does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Experimental Settings

A.1 Benchmarks

To validate the effectiveness of our SpatialMind strategy and ScanForgeQA data in enhancing the visual-spatial understanding capabilities of VLMs, we conducted evaluations on four comprehensive benchmarks: VSI-Bench [1], OpenEQA [52], ScanQA [53], and SQA3D [54]. The comparisons are shown in Table 5.

Table 5: Comparison of different benchmarks. Note that the statistics for the ScanQA and SQA3D datasets are reported on their respective validation sets.

Dataset	#Questions	#Scenes	Source	Question Types
VSI-Bench	>5,000	288	ScanNet, ScanNet++, ARKitScenes	Configuration, Measurement Estimation, Spatio-temporal Reasoning
OpenEQA	1,899	>180	ScanNet, HM3D	Object Recognition, Attribute Reasoning, Spatial Understanding, Functional Reasoning
ScanQA	9,353	71	ScanNet	Object, Color, Quantity, Location
SQA3D	3,261	65	ScanNet	Spatial Relations, Commonsense, Navigation, Multi-hop Reasoning

A.1.1 VSI-Bench

VSI-Bench [1] is a comprehensive evaluation benchmark designed to assess the visual-spatial reasoning capabilities of VLMs in dynamic 3D environments. It comprises over 5,000 high-quality question–answer pairs grounded in 288 diverse indoor video scenes, sourced from real-world 3D reconstruction datasets such as ScanNet [31], ScanNet++ [35], and ARKitScenes [29]. The benchmark defines eight distinct tasks organized into three major categories: **configuration-based tasks** (e.g., object counting, relative direction, and path planning), **measurement estimation tasks** (e.g., object size, absolute distance, room size), and **spatiotemporal reasoning tasks** (e.g., object appearance order). These tasks collectively assess VLMs' abilities to reason about spatial relationships, estimate scale, and understand temporal information. Evaluation is primarily based on accuracy, with tolerance-based error metrics applied to numerical estimation tasks to account for minor deviations. All evaluations follow the official protocols defined in VSI-Bench [1].

A.1.2 OpenEQA

OpenEQA [52], introduced by Meta AI in 2024, is a benchmark designed to evaluate the spatial understanding and reasoning capabilities of VLMs in real-world indoor environments. The dataset comprises over 1,600 human-annotated question–answer pairs spanning 180+ diverse indoor scenes and supports two primary evaluation tasks. In our experiment, we focused on the **Episodic Memory** (EM-EQA) setting, where models must answer questions based solely on a previously observed egocentric video trajectory, without access to external spatial priors. Questions cover a wide range of reasoning types, including object attributes, spatial relationships, object locations, and functional reasoning. To evaluate model perfromance on open-ended responses, we employed GPT-4o [5] as an automatic evaluator to assess semantic alignment between predicted and ground-truth answers. This enables consistent and reliable scoring across diverse question formats.

A.1.3 ScanQA

ScanQA [53] is a large-scale benchmark designed to evaluate spatial question answering in real-world indoor environments, grounded in richly annotated 3D scans. Built on top of the ScanNet [31] dataset, it contains over 41,000 human-curated question–answer pairs across 800 RGB-D scenes, with each question grounded in the 3D geometry and semantics of the scene. The dataset covers a broad range of open-ended questions involving object attributes, spatial relations, and scene-level understanding,

---

[5] https://github.com/mbzuai-oryx/Video-ChatGPT/tree/main/quantitative_evaluation.

Table 6: Performance comparison on VSI-Bench. † indicates results on VSI-Bench (tiny) set and LLaVA-NV denotes the LLaVA-NeXT-Video model.

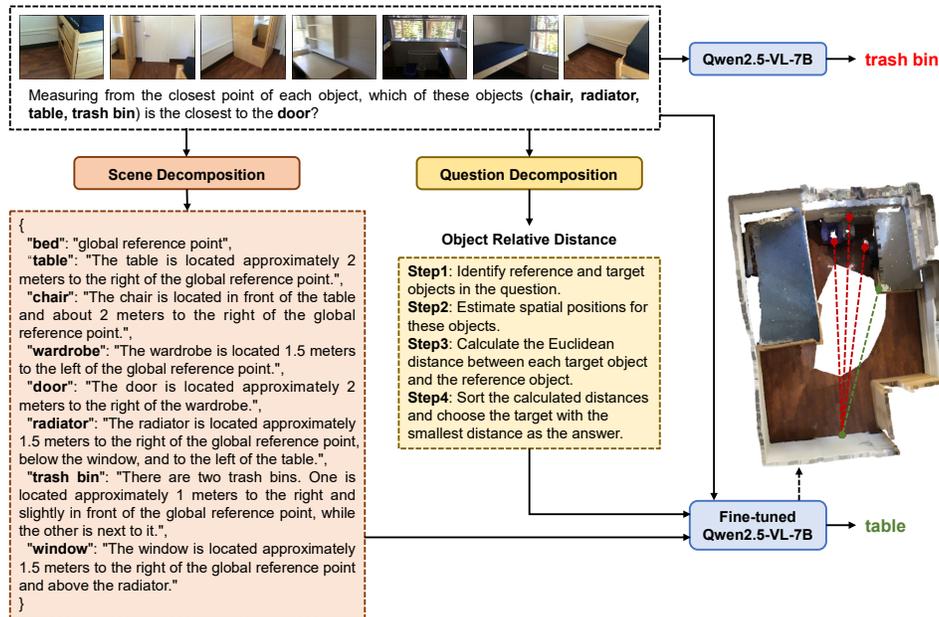| Method	Obj. Count	Abs. Dist.	Obj. Size	Room Size	Rel. Dist.	Rel. Dir.	Route Plan	Appr. Order	Avg	Δ
Close-source										
Human Level†	94.3	47.0	60.4	45.9	94.7	95.8	95.8	100.0	79.2	-
Gemini-2.0 Flash†	52.4	30.6	66.7	31.8	56.0	46.3	24.5	55.1	45.4	-
Gemini-1.5 Pro†	49.6	28.8	58.6	49.4	46.0	48.1	42.0	68.0	48.8	-
Gemini-1.5 Pro	56.2	30.9	64.1	43.6	51.3	46.3	36.0	34.6	45.4	-
+SpatialMind	63.9	51.8	70.2	47.3	56.3	45.9	42.6	44.3	52.8	↑**7.4%**
GPT-4o	46.2	5.3	43.8	38.2	37.0	41.3	31.5	28.5	34.0	-
+SpatialMind	40.0	27.1	62.7	40.9	41.0	39.6	37.1	38.5	40.8	↑**6.8%**
Open-source										
InternVL2-8B	23.1	28.7	48.2	39.8	36.7	30.7	29.9	39.6	34.6	-
+SpatialMind	35.8	28.9	49.7	44.4	37.2	34.8	35.1	45.5	38.9	↑**4.3%**
+ScanForgeQA	45.3	33.4	54.8	45.0	41.1	36.1	33.4	43.0	41.5	↑**6.9%**
+Both	47.0	32.8	53.2	46.6	39.8	36.8	37.9	47.5	42.7	↑**8.1%**
InternVL2-40B	34.9	26.9	46.5	31.8	42.1	32.2	34.0	39.6	36.0	-
+SpatialMind	36.4	30.0	49.1	41.8	43.8	36.1	35.6	50.0	40.4	↑**4.4%**
+ScanForgeQA	51.0	29.2	52.7	38.1	47.2	36.4	35.9	47.6	42.3	↑**6.3%**
+Both	52.2	30.5	54.4	41.0	50.5	37.0	40.2	50.3	44.5	↑**8.5%**
InternVL2.5-8B	7.0	34.1	43.0	42.4	38.0	40.1	23.2	35.9	33.0	-
+SpatialMind	18.7	29.6	46.5	45.2	40.3	40.9	27.8	46.6	37.0	↑**4.0%**
+ScanForgeQA	56.5	33.2	50.7	43.4	39.0	33.1	28.4	34.3	39.8	↑**6.8%**
+Both	52.8	30.7	53.4	44.8	41.1	38.7	29.6	47.0	42.3	↑**9.3%**
InternVL2.5-38B	43.6	33.0	53.0	48.8	53.5	35.7	34.5	34.0	42.0	-
+SpatialMind	56.1	38.3	59.3	52.3	59.7	43.6	39.0	32.9	47.7	↑**5.7%**
LLaVA-NV-7B	48.5	14.0	47.8	24.2	43.5	42.4	34.0	30.6	35.6	-
+SpatialMind	49.0	22.6	47.9	24.6	41.3	43.0	37.1	29.8	36.9	↑**1.3%**
LLaVA-NV-72B	48.9	22.8	57.4	35.3	42.4	36.7	35.0	48.6	40.9	-
+SpatialMind	52.5	24.5	60.2	37.7	42.7	39.5	37.3	51.0	43.2	↑**2.3%**
VideoLLaMA3-7B	36.8	22.2	34.7	24.9	44.6	41.7	36.1	28.8	33.7	-
+SpatialMind	38.6	23.9	36.2	36.6	42.3	40.9	35.9	33.2	36.0	↑**2.3%**
Qwen2-VL-7B	39.4	25.0	25.8	43.2	32.6	30.9	27.8	32.6	32.2	-
+SpatialMind	43.7	29.0	29.4	46.5	33.9	32.8	32.0	31.9	34.9	↑**2.7%**
Qwen2.5-VL-7B	40.3	22.2	50.1	38.9	38.0	40.7	31.4	35.9	37.2	-
+SpatialMind	45.1	25.2	52.1	41.4	38.7	41.6	34.7	34.5	39.2	↑**2.0%**
+ScanForgeQA	53.2	30.5	56.8	44.9	42.3	44.0	37.3	37.7	43.3	↑**6.1%**
+Both	55.0	29.5	57.3	44.0	43.5	44.3	38.3	39.2	43.9	↑**6.7%**
Qwen2.5-VL-72B	37.9	28.6	57.4	49.8	45.5	38.4	20.6	35.4	39.2	-
+SpatialMind	42.3	32.0	61.7	53.8	48.2	43.9	30.4	39.3	44.0	↑**4.8%**
+ScanForgeQA	45.2	32.7	63.3	52.4	50.1	41.7	32.8	40.2	44.8	↑**5.6%**
+Both	48.6	34.4	68.9	54.7	53.4	43.9	30.1	42.7	47.1	↑**7.9%**

Figure 7: A complete example illustrating the visual prompting process with intermediate outputs.

such as "What is between the sofa and the table?" or "Where is the lamp located?". To evaluate model performance, we adopted BLEU-1 as the primary metric. Given that the answers are typically short and factual, unigram precision offers a reliable measure of response quality.

A.1.4 SQA3D

SQA3D [54] is a reasoning-centric benchmark designed to evaluate situational understanding within 3D indoor scenes. Built on 650 scenes from ScanNet [31], it comprises approximately 6,800 annotated situations and over 33,000 questions that span a diverse range of reasoning types, including spatial relations, commonsense reasoning, and multi-hop logical reasoning. Each question is grounded in a specific 3D context, requiring precise and spatially informed answers. To evaluate model performance, we used the Exact Match at 1 (EM-1) metric. This metric is well-suited for the benchmark, as answers are typically concise, specific, and context-dependent.

A.2 Baselines

As shown in Tables 6, our evaluation includes a diverse set of VLM baselines spanning a range of architectures and scales. These include open-source models such as InternVL2 [58] (8B, 40B), InternVL2.5 [58] (8B, 38B), LLaVA-NeXT-Video [59] (7B, 72B), VideoLLaMA3 [60] (7B), Qwen2-VL [61] (7B), and Qwen2.5-VL [62] (7B, 72B). We also evaluated closed-source models, including Gemini-2.0 Flash [6], Gemini-1.5 Pro [7] and GPT-4o [63]. For open-source models, we adopted each model's default parameter settings, including learning rate, number of frames, and input resolution. For closed-source models, GPT-4o processes 16 frames per video, while all Gemini models operate at a fixed sampling rate of 1 frame per second (1 FPS). All experiments are conducted on 8 NVIDIA H20 GPUs.

B   More Experimental Results

B.1   Performance Comparison

In Table 6, we extended our evaluation to additional VLMs with varying architectures and parameter scales, applying both the SpatialMind prompting and ScanForgeQA fine-tuning. The results

---

[6]`https://deepmind.google/technologies/gemini/flash/`.
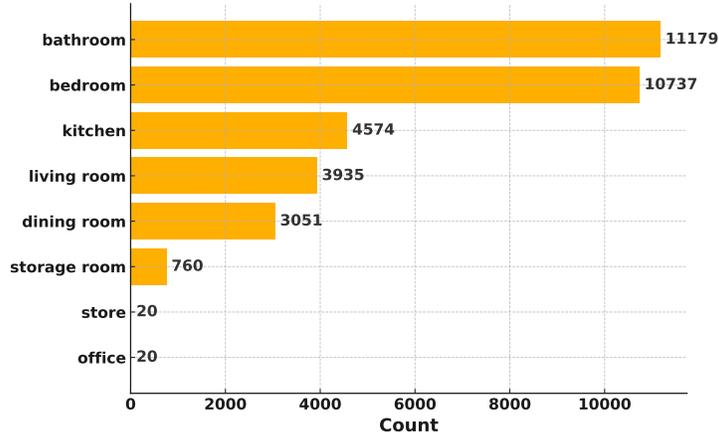[7]`https://deepmind.google/technologies/gemini/`.

Figure 8: Distribution of room types in the ScanForgeQA dataset.

are consistent with those reported in our main analysis, further reinforcing the effectiveness and generalizability of our proposed methods. Moreover, the combined use of prompting and fine-tuning continues to yield superior performance across models, highlighting their complementary strengths and demonstrating the robustness of our framework.

B.2 Case Study

Figure 7 presents a detailed example that illustrates the full prompting process along with intermediate outputs. The scene decomposition module produces approximate spatial descriptions of object locations, effectively capturing the overall layout of the 3D scene. Meanwhile, the question decomposition module identifies the question type as "object relative distance" and selects the corresponding reasoning steps to guide inference. By combining this structured information and feeding it into the Qwen2.5-VL-7B model fine-tuned with ScanForgeQA, the correct target object is successfully identified. This example demonstrates the interpretability and effectiveness of our framework in performing multi-step spatial reasoning grounded in visual context.

C More details for ScanForgeQA

C.1 Scene Distribution

Figure 8 presents the distribution of the eight room categories included in ScanForgeQA. The dataset exhibits a clear long-tailed distribution in which categories such as *bathroom* and *bedroom* are heavily represented, whereas *store* and *office* types are comparatively rare. This imbalance is primarily due to the inherent distribution of room types in the 3D-FRONT dataset [47].

C.2 Scan Example

We provided two frames captured by the camera during the creation of the scanning video, as shown in Figure 9, to visually illustrate this process. Additionally, we further provided scanning creation scripts as well as more video demos in the Supplemental Material:

- "**codes/nav_script.py**" is the script that creates the navigation scan, presenting the exact implementation.
- "**Creation of Scanning Video.mp4**" illustrates the creation process of the scanning video from both first-person and third-person perspectives.
- "**Scanning Video.mkv**" presents the rendered scene scan resulting from the aforementioned scanning process.
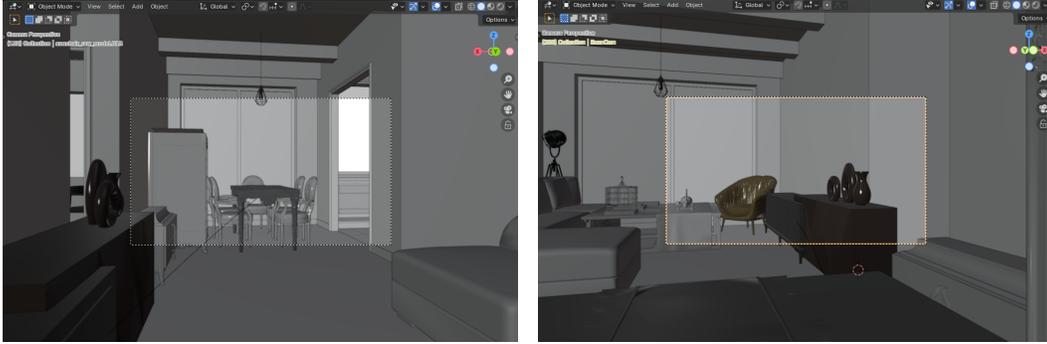
25

Figure 9: Screenshots from the scan creation process.

C.3 QA Definition

Table 7 provides representative templates and corresponding answer sources for each question type in ScanForgeQA. This information clarifies the coverage of different reasoning categories and indicates which scene annotations are used to derive ground-truth answers.

D More details for SpatialMind

D.1 Prompt for Object Description

As a representative example, Figure 10 illustrates the prompt used to generate textual descriptions of object positions. This prompt guides the model to produce structured spatial layouts in natural language based on the visual input. Similar prompts are used for other reasoning components, following a consistent design pattern. Other prompts are provided in the "**codes/gen_scene_exp.py**" file of Supplementary Material for reference and reproducibility.

D.2 Reasoning Steps for Different Questions

Figure 7 provides an example of the detailed reasoning steps used for the *relative distance* question type. Each question type is paired with a concise, structured reasoning process that offers a generalizable solution strategy for VLMs. All reasoning steps for the various question types are included in the "**codes/reason_steps.py**" file of Supplementary Material for reference and reproducibility.

E Limitations

This paper demonstrates that existing VLMs tend to rely more heavily on understanding textual descriptions when performing spatial reasoning. However, textual descriptions alone may not capture spatial semantics as intuitively as visual input, potentially limiting the upper bound of performance for such methods. Moreover, we primarily focus on basic single-room 3D scene. Although strong performance has been achieved on OpenEQA [52], which includes some multi-room scenes, more complex settings, such as large-scale indoor layouts and outdoor environments, remain underexplored and represent important directions for future work.

F Broader Impacts

Our approach improves visual-spatial understanding in VLMs, enabling better interpretation of object layouts and spatial relationships. This enhancement offers potential benefits in areas such as embodied AI and AR/VR applications, where spatial reasoning is essential. However, similar to other prompting and fine-tuning methods, our framework may inherit biases from pre-trained models or training data, potentially leading to unfair or inaccurate spatial interpretations. While our method reduces reliance on expensive 3D sensors, evaluating spatial accuracy remains a key challenge, especially for

downstream tasks that demand high robustness and trustworthiness. Despite these limitations, we believe that releasing our framework and dataset will foster transparency, support reproducibility, and accelerate progress toward trustworthy and robust spatial reasoning systems.

Table 7: Overview of question templates and their corresponding answer sources.

Type	Question Template	Answer Source
Attribute Estimation		
Object Count	How many <A> are there in the room? What is the total number of <A>?	Name of object instance
Object Size	What is the length of the longest side of <A> in meters? What is the size of <A> in square meters? What is the length of the shortest side of <A> in meters? How tall is <A> in meters?	Dimension of the object model
Room Size	What is the size of the room in square meters?	Scene boundaries and scale
Room Type	Based on object layout, what is the most likely type of this room? Is this space a living room, a kitchen, or something else?	Room labels
Spatial Reasoning		
Relative Distance	Which of these objects (<A>, <B>, <C>) is the closest to <R>? Among the listed objects (<A>, <B>, <C>), which one is closest to <R>?	The minimum distance between the centroids of objects
Absolute Distance	What is the distance between <A> and <B> in meters? Measure the distance from <A> to <B> in meters. How far is <A> from <B> in meters?	Euclidean distance between objects
Relative Direction	If I am standing by <A> and facing <B>, which side is object <R> on? From the viewpoint at <A> facing <B>, where is <R>?	The direction of connections between objects
Contact Relationship	Is there a gap between <A> and <B>? Are <A> and <B> touching each other?	Comparison of distances and sizes between objects
Hypothesis Analysis		
Operation Feasibility	Considering only object sizes, is there enough space to put <A> in <B>? Considering only object dimensions, is it feasible to place <A> on <B>?	Comparison of sizes between objects

Prompt

[Task]
You are given a video (multiple frames) capturing an indoor scene. Your goal is to recognize {categories_of_interest} objects, analyze the spatial layout of the scene, and describe the relative position of each object.

[Instructions]
1. Per-frame analysis:
- For each frame, choose one object as a **local reference point**.
- Predict the relative position of all other objects with respect to this local reference point.
- Express relative positions using simple terms like "left", "right", "front", "behind", "above", "below", and approximate distances (e.g., "2 meters to the right").

2. Global scene layout:
- Take the **local reference point from the first frame** as the **global reference point** for the whole video.
- Use overlapping objects between frames to align frames together.
- Gradually build the spatial descriptions for all objects relative to the global reference point.

[Rules]
- If a category has multiple instances (e.g., two chairs), describe each instance separately.
- Preserve the real-world spatial relationships and distances as accurately as possible.
- Use clear and consistent directional and distance terms.

[Output Format]
ONLY Return the result as a JSON dictionary following STRICTLY this format:
{
"category name": "global reference point",
"another category name": "Position description relative to the global reference point",
...
}

Example:
{
"chair": "The chair is located 1.5 meters to the left and 0.5 meters behind the global reference point",
"table": "The table is located 2 meters to the right of the global reference point",
"lamp": "The lamp is located 1 meter above and 1 meter behind the global reference point"
}

Figure 10: Example prompt used to generate textual descriptions of object positions.