# LitSpaR: A Lightweight Spatial Reasoning Model for Indoor Scene Understanding

Hui Han
Dewu
Shanghai, China
clearhanhui@outlook.com

Zaifan Jiang
Dewu
Shanghai, China
jiangzaifan@dewu.com

Chao Wei
Dewu
Shanghai, China
weichao@dewu.com

## Abstract

*Spatial reasoning is an important ability for Multimodal Large Language Models (MLLMs) to understand the physical world. While current MLLMs can generate answers from 2D images, they lack the inherent ability to perceive 3D objects and reason about spatial relationships. In this paper, we demonstrate that point clouds provide rich geometric information for spatial tasks, enabling a lightweight model to achieve competitive performance comparable to Vision-Language Models (VLMs). We introduce Lightweight Spatial Reasoner (LitSpaR), a multimodal model that takes point clouds as input and performs spatial reasoning via 3D visual grounding, serving as the foundation for spatial reasoning. To achieve this, we curate a synthetic dataset and three real-world datasets with the 'grounding-then-reasoning' schema for training. LitSpaR achieves competitive performance on VSI-Bench, where we adapt the benchmark to accept point clouds as input instead of images. Beyond spatial reasoning, LitSpaR retains semantic Question Answering (QA) capabilities and supports semantic object insertion, making it well-suited for AR/VR applications.*

## 1. Introduction

Spatial reasoning capabilities, including perceiving spatial layouts, object sizes, positions, and their relationships, are essential for diverse domains, such as embodied intelligence, autonomous driving, and AR/VR. Recent advanced Vision-Language Models (VLMs) [1, 4, 5, 13], pretrained on image-text and video-text pairs, have made remarkable strides in 2D vision understanding. However, their ability suffer from inherent limitations in 3D spatial comprehension due to the loss of spatial information during 3D-to-2D projection. Although some datasets provide RGB-D images and raw camera parameters for 3D reconstruction, directly utilizing these 3D parameters for reasoning remains challenging. This is because, unlike semantic concepts, these 3D parameters serve primarily as mathematical transformations; their physical semantics are not directly accessible, making them non-intuitive for both humans and VLMs without explicit decoding mechanisms.

Point clouds, consist of a set of sparse points sampled from 3D space, provide an explicit geometry representation of the physical world. Recognizing this advantage, some researchers [12, 16] have proposed models by taking point clouds as multimodal inputs, while they mainly focus on semantic Question Answering (QA) or 3D grounding, lacking the ability to reason in space. Some studies [9, 18, 20] also explored the integration of encoded spatial data as auxiliary inputs for VLMs, demonstrating promising improvements in spatial reasoning tasks. However, these approaches remain reliant on 2D visual inputs, which may incur substantial computational overhead and architectural complexity, and may have restrictions on visual input configuration. In contrast, humans can understand the 3D environment using only spatial cues such as touch and proprioception, even without visual input—a capability that inspires our exploration of point-cloud-only reasoning.

However, for spatial reasoning tasks, we identify two key limitations in existing training recipes. First, most point cloud encoders lack the capacity to perceive relational attributes between objects, such as direction and distance. Second, due to the scarcity and inherent noise in real-world datasets, direct training on such data often yields suboptimal performance. To address the first issue, we design a grounding-reasoning schema that integrates 3D grounding into the reasoning chain, aligning with recent advances [9, 20]. This design emulates human spatial cognition, wherein objects are first localized before their properties and inter-object relationships are reasoned about. For the second issue, we introduce a two-stage curriculum learning framework: the model is initially trained on synthetic datasets to acquire robust foundational representations, and subsequently fine-tuned on real-world data to enhance domain alignment and generalization.

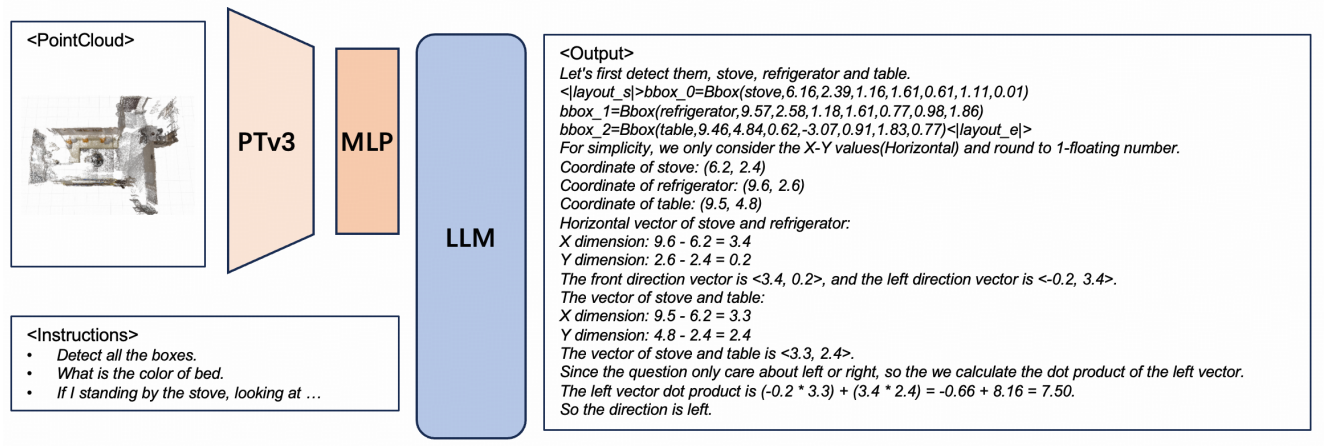



Figure 1. Architecture of LitSpaR. The input point cloud is encoded by a Point Transformer V3 and its features are aligned to the LLM backbone via a two layers MLP adapter. For spatial reasoning tasks, LitSpaR first grounds the objects of interest, then calculates their geometric relationships to get the final answer.

In this work, we present LitSpaR, a multimodal large language model (MLLM) specialized for 3D spatial reasoning, semantic question answering, and semantic object insertion—the task of generating object placement anchors in 3D space from language instructions. Unlike prior approaches [9, 18, 20], LitSpaR relies exclusively on point clouds as multimodal input, eliminating the need for camera calibration and 2D visual encoding. To equip the model with these capabilities, we curate four spatial reasoning datasets following the proposed grounding-reasoning schema. We employ the *SpatialLM Dataset* [16] for the first stage of curriculum training, and a combination of *ScanNet* [11], *ArkitScenes* [6], and *ScanNet++* [24] for the second stage. During training, we apply an exponential moving average (EMA) strategy to the model weights to further improve performance. Finally, despite comprising only 580M parameters, LitSpaR achieves competitive results while offering substantial efficiency advantages.

Our main contributions are summarized as follows:

- **First Point-Cloud Reasoning MLLM:** We propose LitSpaR, a lightweight model that performs spatial reasoning exclusively from point cloud, achieving superior performance on the VSI-Bench benchmark with only 580M parameters.
- **General Abilities:** In addition to maintaining semantic QA ability, we also introduce *Semantic Object Insertion* for LitSpaR, a critical feature for immersive AR/VR interaction experiences.
- **Training Recipe:** We identify key limitations in existing spatial training pipelines and address them through a grounding-reasoning schema and a two-stage curriculum learning strategy.
- **Grounding Evaluation:** We introduce a novel evaluation metric for 3D grounding tasks, designed to align more closely with human spatial intuition than existing metrics.

## 2. Related work

**3D Understanding via 2D Vision.** This represents a mainstream research direction, with an increasing number of models and approaches recently proposed. These models are primarily built upon VLMs, taking videos as the primary input and 3D data as auxiliary information. 3D-LLM [14] employs a pre-trained image encoder to extract 2D features from multi-view images and back-projects them onto the point cloud. It then introduces a Q-Former [15] to align point clouds features with the Large Language Model (LLM) using a fixed number of tokens. LLaVA-3D [26] proposes encoding 3D positional embeddings and injecting them into the 2D features of a CLIP [17] encoder, enabling the VLM to perceive 3D information. Spatial-MLLM [21] enhances spatial reasoning by incorporating geometric features from VGGT [19], a feed-forward neural network that directly infers key 3D attributes of a scene for reconstruction. N3D-VLM [20] adopts an architecture similar to LLaVA-3D and employs the same grounding-then-reasoning paradigm during training. GS-Reasoner [9] reconstructs videos to point clouds and resulting point clouds to geometric features with PTv3 [22] of their dataset. Then, it leverages a cross-attention module to fuse the geometric features and vision features. In contrast, our model is lightweight and takes point clouds as the sole input, yet achieves competitive results.

**3D Understanding via Point Cloud.** These models, like ours, accept point clouds as their exclusive vision input modality. LL3DA [8] adopts a 3D visual encoder pre-trained on ScanNet detection as the scene encoder. 3D-LLaVA [12] uses a 3D U-net [10] to process point clouds

```python
class Bbox:
    name: str
    position_x: float
    position_y: float
    position_z: float
    angle_z: float
    scale_x: float
    scale_y: float
    scale_z: float
```

Figure 2. Definition of our structured representation for bounding box.

into super point features, and uses Omni Superpoint Transformer (OST), composed of self-attention layers, to further extract features for LLM. SceneScript [2] uses a sparse ResNet as the point cloud encoder, then applies a decoder-only transformer to output the scene layout. SpatialLM [12] performs 3D grounding from point clouds and predicts 3D bounding boxes. While it performs grounding, it focuses primarily on this task and lacks spatial reasoning and QA capabilities.

## 3. LitSpaR

### 3.1. Overview

LitSpaR adopts the architecture similar to SpatialLM [16], composed of a point cloud encoder, a two-layer MLP adapter, and an LLM, trained in next token prediction manner. The encoder-only Point Transformer v3 (PTv3) [22] is adopted as the point cloud encoder, with parameters initialized from Sonata [23] to leverage its superior performance. The PTv3 encoder accepts a set of $N$ points as input, where each point feature is formed by concatenating coordinates, colors, and normals (if available). The input points are serialized via several space-filling curves, and partitioned into ordered subsets by a fixed length. Attention is computed within these subsets to reduce computational overhead and preserve locality. Through several transformer and max-pooling layers, these points are finally mapped to $M$ points feature embeddings with dimension $N$, where $M$ is significantly smaller than $N$ in two or three orders of magnitude. The encoder can be formalized as follows:

$$\mathbf{F} = \mathcal{E}(\mathbf{P}), \quad \mathbf{P} \in \mathbb{R}^{N\times\{6,9\}}, \mathbf{F} \in \mathbb{R}^{M\times D}. \tag{1}$$

Figure 2 shows the 3D grounding bounding box representation we defined, which follows SpatialLM [16]. We design the grounding-then-reasoning schema for the LitSpaR, like [9, 20]. It detects all the bounding boxes of objects of interest, and all the grounding outputs are enclosed within `<layout_s>...</layout_e>`. All the numbers are floating-point values with two decimal places, including the positions, scales, and rotations around the z-axis. Then, the model reasons about the spatial relations

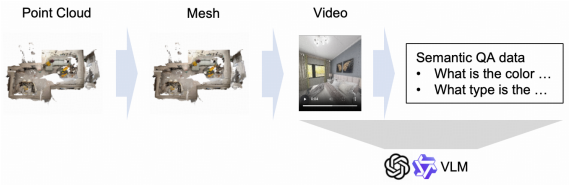



Figure 3. Semantic QA dataset curation pipeline.

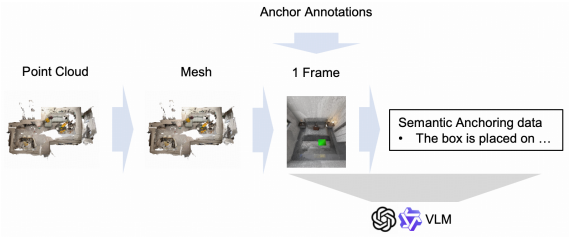
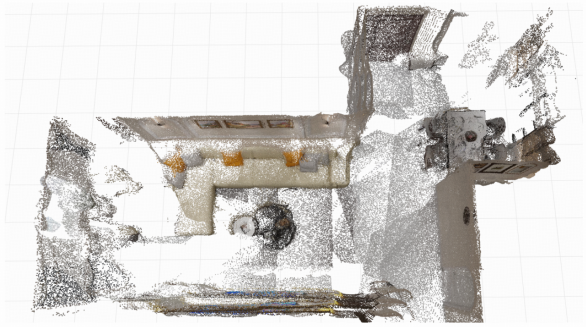



Figure 4. Semantic object insertion dataset curation pipeline.



**Human:**
How is the room?
**GPT:**
The room appears to be a modern, spacious interior with a minimalist design. It features a large open area with a high ceiling, white walls, and a smooth floor. There are multiple seating areas with sofas and tables, arranged in a seating layout. Large windows allow natural light to fill the space, and there are some decorative elements like wall art and plants. The overall atmosphere is clean and contemporary.
**Human:**
What is the color of coffee table?
**GPT:**
The coffee table is dark in color, appearing to be black or a very dark shade.

Figure 5. Two examples of QA ability.

between objects. We present an example of how LitSpaR determines the direction in VSI-Bench in Figure 1. It calculates two dot products: one between the target vector and the front vector, and another between the target vector and the left vector. By comparing these two results, it determines the answer `left`.

### 3.2. Dataset Curation

Our training datasets comprise one synthetic dataset and three real-world datasets, summarized in Table 2. All datasets, including the evaluation set, focus on indoor scenes.

Table 1. Evaluation on VSI-Bench. We cite GS-Reasoner's results as reference, though direct comparison is inequitable by its requirement for both video and point cloud.

| Models | Obj. Cnt. | Abs. Dist. | Obj. Size | Room Size | Rel. Dist. | Rel. Dir. (e./m./h.) |
|---|---|---|---|---|---|---|
| GPT-4o* | 46.2 | 5.3 | 43.8 | 38.2 | 37.0 | 41.3 |
| Gemini-1.5 Pro* | 45.4 | 56.2 | 30.9 | 64.1 | 43.6 | 51.3 |
| LLaVA-NeXT-Video-72B* | 48.9 | 22.8 | 57.4 | 35.3 | 42.4 | 36.7 |
| GS-Reasoner (pred dep.)* | 69.1 | 61.9 | 70.0 | 65.7 | 65.4 | 88.9 |
| GS-Reasoner (gt dep.)* | 70.9 | 73.6 | 77.8 | 81.8 | 70.6 | 90.5 |
| LitSpaR (ours, w/o syn.) | 55.5 | 38.8 | 62.5 | 84.9 | 35.4 | 48.6 (54.8,46.8,44.2) |
| LitSpaR (ours) | 71.6 | 68.9 | 78.3 | 84.8 | 57.5 | 86.8 (89.9,85.4,85.3) |

*: Models that has video input, while our model uses point cloud.
e./m./h.: Abbreviations for easy, medium, and hard difficulty levels, respectively.

Table 2. Dataset statistics.

| Dataset | Type | #Scenes | #Train | #Test |
|---|---|---|---|---|
| SpatialLM [16] | syn. | 12,328 | 12,328 | 0 |
| ScanNet [11] | real | 1,513 | 1,201 | 312 |
| ARKitScenes [6] | real | 5,048 | 4,898 | 150 |
| ScanNet++ [24] | real | 1,006 | 906 | 50 |

For the spatial reasoning task, unlike GCoT [9], which relies on querying a GPT, we employ programmatic generation. This approach is cost-effective while maintaining competitive model performance. Regarding the QA task, the data collection procedures differ between synthetic and real-world datasets. SpatialLM provides only point clouds files and bounding box annotations, whereas QA pairs and video data remain inaccessible. Consequently, we established a pipeline that first renders point clouds into videos and then queries a VLM to generate QA pairs, thereby constructing a semantic QA dataset as illustrated in Figure 3. Conversely, for real-world datasets where video data is readily available, we directly query the VLM to collect the semantic QA dataset. Finally, the semantic object insertion task data is generated via a dedicated pipeline, depicted in Figure 4. Its distinction from the QA pipeline lies in the mesh rendering process; specifically, we incorporate the target bounding box (highlighted in green) into the mesh and render it as a single frame.

Table 3. Processed dataset statistics.

| Task | syn. | real |
|---|---|---|
| 3D Grounding | 1.03M | 29.1K |
| Spatial Reasoning | 3.46M | 119.6K |
| Semantic QA | 200.6K | 99.1K |
| Semantic Object Insertion | 27.8K | - |

Finally, we summarize our processed training dataset statistics in Table 3. For the semantic object insertion task, because the scenes in real-world datasets are not as simple as those in the synthetic dataset, we are still working on it.

We adopt VSI-Bench [7] as our evaluation benchmark, which comprises over 5,000 question-answer pairs sourced from ScanNet [11], ScanNet++ [24], and ARKitScenes [6]. The benchmark originally defines eight tasks focused on video-based spatial and temporal reasoning. However, given that our model processes point clouds, we substitute the original videos input with the corresponding official point clouds. Additionally, we exclude two tasks, *Route Plan* and *Appr. Order*, due to their potential dependence on temporal cues, which are inaccessible in static representations like point clouds.

### 3.3. 3D Grounding Metric $\text{DIoU}_{3D}$

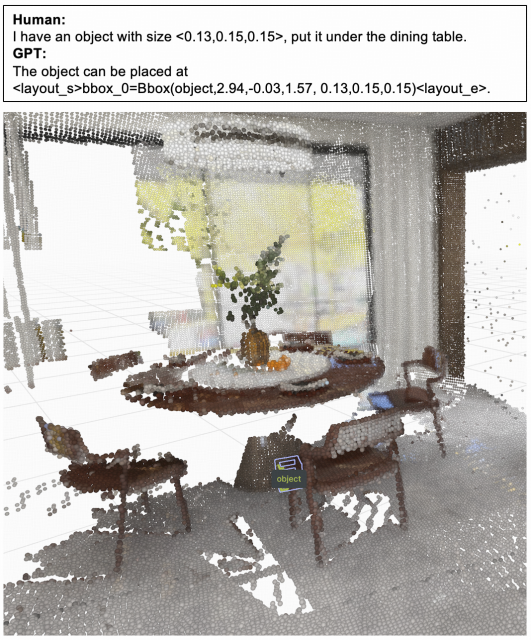
Previous works evaluate the performance of 3D grounding bounding box by determining whether their Intersection over Union (IoU) with the ground truth exceeds a specific threshold [9, 16], e.g., $\text{IoU}_{3D}$@0.25. The $\text{IoU}_{3D}$ score function is defined as follows:

$$\text{IoU}_{3D}(B_{pred}, B_{gt}) = \frac{\text{Vol}(B_{pred} \cap B_{gt})}{\text{Vol}(B_{pred} \cup B_{gt})}, \quad (2)$$

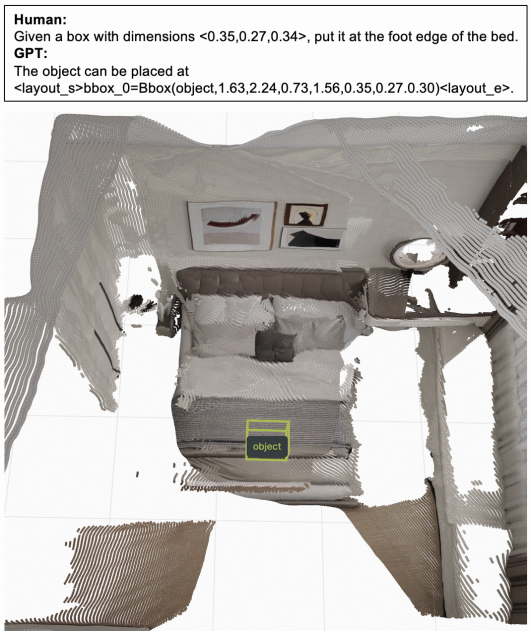
where B denotes the bounding box, and Vol is the volume function.

Although simple, this function has limitations in discrimination and continuity, and tends to assign extremely low scores, especially when encountering objects whose dimensions along the x, y, and z axes differ significantly in scale, such as TVs, carpets, and paintings. As shown in Figure 7, if we apply the $\text{IoU}_{3D}$ function, the predictions in sub-figures (a) and (b) yield the same score, i.e., 0. However, (a) has a slight influence on its spatial relationships, while (b) does not.

Therefore, we require a new evaluation metric that aligns with human intuition. We follow the definition of DIoU [25], introduced $\text{DIoU}_{3D}$, which is defined as follows:




(a) Instruction: "Put it under the dining table".




(b) Instruction: "Put it at the foot edge of the bed".

Figure 6. Two examples of semantic object insertion and their visualization results.

$$\text{DIoU}^*_{3D} = \text{IoU}_{3D} - \frac{\rho^2(B_{pred}, B_{gt})}{c^2}, \tag{3}$$

$$\text{DIoU}_{3D} = \frac{\text{DIoU}^*_{3D} + 1}{2}. \tag{4}$$

The function $\rho^2(\cdot)$ represents the squared Euclidean distance between center points of the two boxes, while $c$ denotes the diagonal length of the smallest enclosing box that covers both $B_{pred}$ and $B_{gt}$. Given that raw $\text{DIoU}^*_{3D}$ typically ranges in $(-1, 1]$, we apply a linear transformation to normalize it to $(0, 1]$, thereby facilitating direct comparison with $\text{IoU}_{3D}$. Regarding the smallest enclosing box, we project the eight corner points of both the predicted and ground truth boxes onto the xy-plane and apply `cv2.minAreaRect` to obtain the smallest 2D bounding box. Since the z-axis is consistently oriented across our dataset, the lower and upper bounds of the smallest enclosing box are directly determined by the minimum and maximum z-coordinates of the two boxes, respectively.

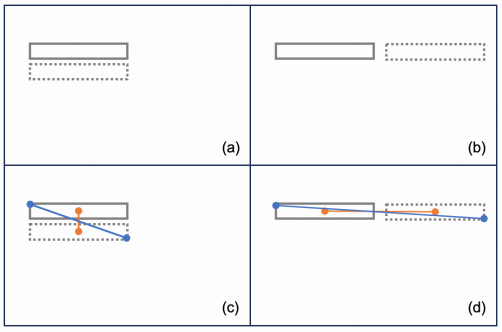



Figure 7. BEV example explains why DIoU-3D is better than IoU-3D. Gray solid boxes denote ground truth TV location, while dashed boxes represent predictions. The red and blue lines represent the center and the smallest enclosing box diagonal distance of the two boxes, respectively.

## 4. Experiment

The main results on VSI-Bench are reported in Section 4.1. We provide two spatial semantic QA and object insertion examples in Section 4.2 to demonstrate the model's general capabilities. Results for 3D Grounding are presented in Section 4.3, evaluated using $\text{DIoU}_{3D}$. Although 3D grounding is not the primary focus of this work, we include these results to ensure experimental completeness and to demonstrate the efficacy of the new evaluation metric.

### 4.1. Spatial Reasoning

We evaluate LitSpaR on an adapted version of the VSI-Bench benchmark [7], where we substitute videos with point clouds as input and select six out of the eight available tasks, as detailed in Section 3.2. We follow the official evaluation metrics of VSI-Bench, employing Accuracy for multiple-choice questions and Mean Relative Accuracy (MRA) for numerical estimation tasks. The results are presented in Table 1. Due to the scarcity of existing spatial reasoning models compatible with our exclusive point clouds setting, a direct and fair comparison is currently unavail-

Table 4. Comparison of F1-Scores (in %) with $\text{DIoU}_{3D}$ and $\text{IoU}_{3D}$. Objects like *painting* and *carpets* get a higher score.

| | $\text{DIoU}_{3D}$ | | $\text{IoU}_{3D}$ | |
|---|---|---|---|---|
| **Objects** | **F1@25** | **F1@50** | **F1@25** | **F1@50** |
| door | 63.27 | 35.03 | 17.44 | 6.50 |
| window | 50.84 | 32.92 | 15.66 | 5.61 |
| curtain | 57.24 | 38.96 | 18.02 | 4.92 |
| nightstand | 70.83 | 67.86 | 44.94 | 9.82 |
| chandelier | 54.44 | 36.80 | 25.42 | 7.84 |
| wardrobe | 43.33 | 38.67 | 26.67 | 10.00 |
| bed | 96.83 | 95.24 | 93.65 | 80.95 |
| sofa | 74.21 | 60.32 | 35.71 | 11.90 |
| chair | 24.71 | 24.71 | 18.10 | 10.34 |
| cabinet | 26.36 | 12.65 | 7.73 | 0.91 |
| dining table | 56.10 | 43.90 | 26.83 | 7.32 |
| plants | 30.00 | 15.48 | 15.48 | 2.38 |
| tv cabinet | 36.36 | 33.33 | 15.15 | 0.00 |
| coffee table | 36.36 | 33.33 | 18.18 | 0.00 |
| side table | 5.88 | 5.88 | 0.00 | 0.00 |
| air cond. | 18.52 | 11.11 | 0.00 | 0.00 |
| dresser | 42.11 | 42.11 | 26.32 | 15.79 |
| stool | 25.00 | 25.00 | 6.25 | 0.00 |
| refrigerator | 0.00 | 0.00 | 0.00 | 0.00 |
| painting | 69.05 | 34.44 | 16.82 | 4.21 |
| carpet | 39.39 | 17.68 | 12.12 | 6.06 |
| tv | 58.33 | 25.00 | 10.42 | 2.08 |
| **Mean** | 44.51 | 33.20 | 20.50 | 8.48 |

able. However, GS-Reasoner [9] applies a similar reasoning paradigm and evaluation tasks like ours; therefore, we report selected results from their work as a reference baseline. Compared to advanced VLMs, LitSpaR outperforms them on all tasks. Compared to GS-Reasoner, which employs a similar paradigm, we also outperform it on several tasks.

However, our model under-performs on tasks including Absolute Distance (Abs. Dist.) and Relative Distance (Rel. Dist.) by a notable margin than GS-Reasoner. We attribute this discrepancy to the data alignment strategy: GS-Reasoner aligns its point clouds data along all three axes (X, Y, and Z), which is crucial for accurately reasoning about distances between closely situated bounding boxes. In contrast, our setting only enforces alignment along the Z-axis (allowing rotation around it) to enhance the model's practical applicability. For instance, in AR/VR applications, aligning point clouds with the gravity vector (Z-axis) is necessary, whereas precise alignment along the X and Y axes is not.

### 4.2. Spatial Semantic QA and Object Insertion

Our model maintains robust capabilities for semantic question answering, and examples illustrating this functionality are presented in Figure 5. We are currently working on conducting quantitative assessments to further validate our model's performance, e.g., evaluations on ScanQA [3].

A novel and important feature of our model is *Semantic Object Insertion*, which enables the model to interpret natural language instructions and precisely localize anchors within the point cloud. Unlike traditional methods that rely on rigid geometric cues (e.g., plane detection and ray-casting), our language-driven mechanism establishes a new interaction paradigm for AR/VR applications, enabling users to convey their intentions through natural language. Figure 6 provides two examples, showing both the predicted insertion coordinates and their visualization in the point cloud.

### 4.3. 3D Grounding

3D Grounding is the basic ability for understanding the space. However, we observe that objects with thin bounding boxes or imbalanced dimensional scales often yield low scores. The introduction of $\text{DIoU}_{3D}$ alleviates this issue. As reported in Table 4, categories such as door, window, and tv, which achieved extremely low scores on $\text{IoU}_{3D}$, show improved performance on $\text{DIoU}_{3D}$.

## 5. Conclusion, Limitations and Future Work

In this paper, we propose LitSpaR, a spatial reasoning model that achieves competitive performance on 3D tasks despite its small parameter count. Additionally, we introduce a novel evaluation metric for 3D grounding that aligns more closely with human spatial perception. A novel feature of LitSpaR is its semantic object insertion capability, which localizes 3D bounding boxes for virtual objects, facilitating AR/VR applications.

Given the early-stage nature of this study and constraints on time and computational resources, certain aspects of our implementation and experiments remain preliminary and require future refinement. For example, trade-offs were made in our training recipe: while we generated millions of synthetic samples, we were limited to sampling at most 200K per task for training. We also acknowledge that, despite our focus on point cloud, 2D vision remains essential for physical world understanding, complementing geometric data by offering vital dense semantic information.

In future work, we plan to perform deeper analyses, including comprehensive ablation studies, and further boost performance. Furthermore, while PTv3 provides a solid foundation for point clouds encoding, exploring more efficient and scalable encoders remains a key direction for future research.

## References


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

[2] Armen Avetisyan, Christopher Xie, Henry Howard-Jenkins, Tsun-Yi Yang, Samir Aroudj, Suvam Patra, Fuyang Zhang, Duncan Frost, Luke Holland, Campbell Orme, Jakob Engel, Edward Miller, Richard Newcombe, and Vasileios Balntas. Scenescript: Reconstructing scenes with an autoregressive structured language model. In *European Conference on Computer Vision (ECCV)*, 2024. 3

[3] 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. 6

[4] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025. 1

[5] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. 1

[6] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. 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)*, 2021. 2, 4

[7] Ellis Brown, Jihan Yang, Shusheng Yang, Rob Fergus, and Saining Xie. Benchmark Designers Should "Train on the Test Set" to Expose Exploitable Non-Visual Shortcuts. *arXiv preprint arXiv:2511.04655*, 2025. 4, 5

[8] 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. 2

[9] Yiming Chen, Zekun Qi, Wenyao Zhang, Xin Jin, Li Zhang, and Peidong Liu. Reasoning in space via grounding in the world, 2025. 1, 2, 3, 4, 6

[10] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016. 2

[11] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 2, 4

[12] Jiajun Deng, Tianyu He, Li Jiang, Tianyu Wang, Feras Dayoub, and Ian Reid. 3d-llava: Towards generalist 3d lmms with omni superpoint transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 1, 2, 3

[13] Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025. 1

[14] 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. *NeurIPS*, 2023. 2

[15] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 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, 2023. 2

[16] Yongsen Mao, Junhao Zhong, Chuan Fang, Jia Zheng, Rui Tang, Hao Zhu, Ping Tan, and Zihan Zhou. Spatiallm: Training large language models for structured indoor modeling. In *Advances in Neural Information Processing Systems*, 2025. 1, 2, 3, 4

[17] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 2

[18] Yanwei Li Jingjia Huang Shen Yan Siyuan Zhou Zhe Liu Xiangtai Li Shuangye Li Wenqian Wang Yi Lin Hengshuang Zhao Rui Yang, Ziyu Zhu. Visual spatial tuning. *arXiv preprint arXiv:2511.05491*, 2025. 1, 2

[19] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 2

[20] Yuxin Wang, Lei Ke, Boqiang Zhang, Tianyuan Qu, Hanxun Yu, Zhenpeng Huang, Meng Yu, Dan Xu, and Dong Yu. N3d-vlm: Native 3d grounding enables accurate spatial reasoning in vision-language models. *arXiv preprint arXiv:2512.16561*, 2025. 1, 2, 3

[21] Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-mllm: Boosting mllm capabilities in visual-based spatial intelligence. *arXiv preprint arXiv:2505.23747*, 2025. 2

[22] 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 *CVPR*, 2024. 2, 3

[23] Xiaoyang Wu, Daniel DeTone, Duncan Frost, Tianwei Shen, Chris Xie, Nan Yang, Jakob Engel, Richard Newcombe, Hengshuang Zhao, and Julian Straub. Sonata: Self-

supervised learning of reliable point representations. In *CVPR*, 2025. 3

[24] 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. 2, 4

[25] Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren. Distance-iou loss: Faster and better learning for bounding box regression. In *Proceedings of the AAAI conference on artificial intelligence*, pages 12993–13000, 2020. 4

[26] Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. *arXiv preprint arXiv:2409.18125*, 2024. 2