Direction-aware 3D Large Multimodal Models

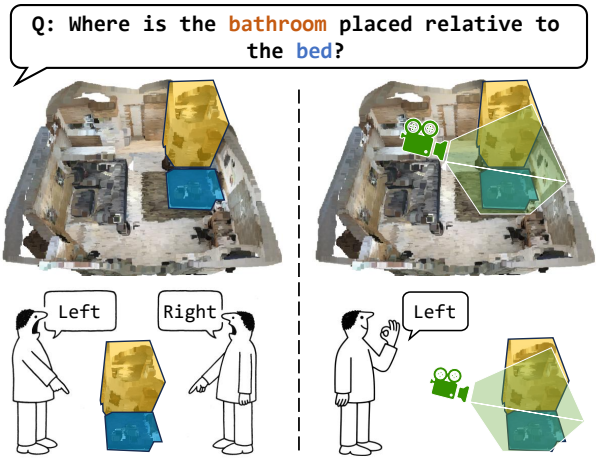
Quan Liu[1] Weihao Xuan[2,3] Junjue Wang[2] Naoto Yokoya[2,3] Ling Shao[4] Shijian Lu[1,†]
[1]Nanyang Technological University [2]The University of Tokyo [3]RIKEN AIP
[4]UCAS-Terminus AI Lab, University of Chinese Academy of Sciences

Abstract

3D large multimodal models (3D LMMs) rely heavily on ego poses for enabling directional question-answering and spatial reasoning. However, most existing point cloud benchmarks contain rich directional queries but lack the corresponding ego poses, making them inherently ill-posed in 3D large multimodal modelling. In this work, we redefine a new and rigorous paradigm that enables direction-aware 3D LMMs by identifying and supplementing ego poses into point cloud benchmarks and transforming the corresponding point cloud data according to the identified ego poses. We enable direction-aware 3D LMMs with two novel designs. The first is PoseRecover, a fully automatic pose recovery pipeline that matches questions with ego poses from RGB-D video extrinsics via object–frustum intersection and visibility check with Z-buffers. The second is PoseAlign that transforms the point cloud data to be aligned with the identified ego poses instead of either injecting ego poses into textual prompts or introducing pose-encoded features in the projection layers. Extensive experiments show that our designs yield consistent improvements across multiple 3D LMM backbones such as LL3DA, LL3DA-SONATA, Chat-Scene, and 3D-LLAVA, improving ScanRefer mIoU by 30.0% *and Scan2Cap LLM-as-judge accuracy by* 11.7%. *In addition, our approach is simple, generic, and training-efficient, requiring only instruction tuning while establishing a strong baseline for direction-aware 3D-LMMs.*

1. Introduction

Generalist point cloud 3D large multimodal models (3D-LMMs) pursue broad competence in 3D scene understanding and reasoning, targeting key tasks such as object grounding [25, 49, 56, 58], referring [1, 5, 57], question answering [2, 55], ego position reasoning [34], object captioning [10, 59], route planning [43], segmentation [18, 44], detection [45], etc. Such enabling represents a key step towards the visual-cognitive core of future embodied agents [14, 22]. In particular, an ideal generalist 3D-LMM should possess persistent awareness of ego poses and spatial directions in the world coordinate for accurate understanding of 3D input data [41, 52], robust and alias-free spatial reasoning [14, 60], and safely navigating in 3D environments [22].



(a) Direction-agnostic 3D LMM (b) Direction-aware 3D LMM

Figure 1. Ego pose is critical in spatial reasoning and understanding. (a) Direction-agnostic 3D LMMs are struggling to reason spatial directions due to the absence of ego-pose information. (b) Incorporating ego pose resolves directional ambiguity, enabling consistent and robust spatial reasoning.

Ego-pose awareness entails distinguishing lateral directions (e.g., 'left', 'right', 'front', 'back') that are defined according to a reference frame [15]. Two forms of directions have been widely adopted: 1) *Egocentric Direction* that defines spatial relations relative to the agent's own body axes (*e.g.*, 'the object is on my left') and 2) *Allocentric Direction* that defines spatial relations relative to an external anchor (*e.g.*, 'the man is on the left of his car'). For indoor scenarios where the anchor has no canonical axes, like a plate or a table, the reference of the allocentric direction falls back to the ego agent [15]. Hence, an ego pose is essential to reconcile the two forms of directions to achieve consistent spatial reasoning and understanding by 3D LMMs.

However, most existing 3D indoor datasets such as ScanRefer [5], Multi3DRefer [61], ScanQA [2], Scan2Cap [10], and Nr3D [1] contain many allocentric directional queries

† Corresponding author: Shijian.Lu@ntu.edu.sg
Code is available at https://github.com/liuQuan98/PoseAlign3D

but provide no explicit ego poses, positioning spatial reasoning ill-posed and hindering the acquisition of persistent ego pose awareness as illustrated in Figure 1. This omission largely stems from their assumption of a global third-person rather than an egocentric first-person perspective of the scene. Prior studies address this omission mostly by creating new datasets instead of rectifying existing ones, requiring 3D LMMs to infer ego poses while responding to spatial queries [34, 60]. Actually, inferring ego poses as a latent variable is conceptually redundant, as ego poses are already available when an embodied agent collects indoor point clouds via simultaneous localization and mapping (SLAM) [36]. To this end, we propose to directly incorporate the readily available pose data as model inputs due to two factors: (1) it rigorously resolves the directional ambiguity in existing 3D indoor benchmarks; (2) it does not affect the application scope of 3D LMMs by leveraging the 'free-lunch' pose in the practical embodied workflow. More discussion is available in section A.1.

For efficient implementation, we enable the new paradigm with two new designs. The first is *PoseRecover*, an automatic pose generation pipeline that addresses the lack of ego poses in existing benchmarks. Specifically, *PoseRecover* retrieves question-related camera poses from ScanNet RGB-D sequences by matching camera frustums with relevant object annotations such as segmentation masks, bounding boxes, and location annotators, leading to a list of candidate camera poses for online training or inference. During the forward pass, candidates undergo various screenings to purge opposite-view errors and ensure authenticity while maintaining a certain degree of variety. The second is *PoseAlign*, a simple solution that incorporates the identified ego pose into point cloud data to enable their interpretation by existing 3D LMMs. Specifically, *PoseAlign* repositions point clouds to align with the identified ego poses, enabling universal direction-awareness across existing 3D LMMs of different architectures. Thanks to the coordinate sensitivity of pretrained point cloud encoders, the direction awareness can be boosted without even tuning the encoder of 3D LMMs.

The contributions of this work can be summarized in three major aspects. First, we identify that most existing 3D LMM benchmarks suffer from ill-posed directions and propose a new paradigm that mitigates this problem effectively by incorporating an ego pose. Second, we design PoseRecover and PoseAlign, the former being a pose generation pipeline that addresses the ill-posed problem in 3D LMM benchmarks by recovering mission-critical pose data, and the latter being a simple yet effective modifier that enables direction-awareness for existing 3D LMMs by injecting pose data into the point cloud. Third, extensive experiments show that our approach improves the direction awareness substantially and consistently across 4 different benchmarks and all 3D LMM architectures.

2. Related Work

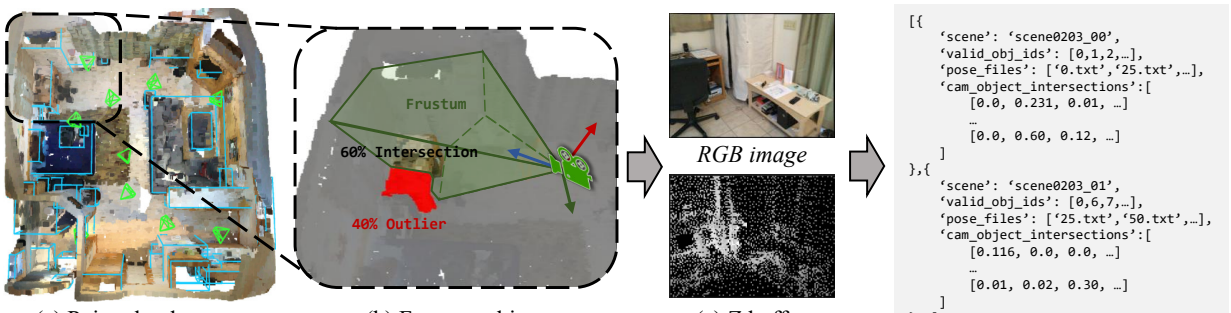
2.1. Indoor 3D LMM Benchmarks

Following the huge success of 2D visual understanding benchmarks [32, 33] and the richly-annotated indoor datasets [4, 12, 49], the first batch of 3D understanding benchmarks [1, 55, 57] take similar approaches to link objects with text descriptions in 3D with expert human annotators, achieving a stunning variety and volume of tasks, *e.g.*, referral [5], captioning [10], and question-answering [2]. While one could argue that their human-centric pipelines implied the viewing direction must be from a human standing inside the room, not explicitly defining such poses causes their questions to be inherently ill-posed [60]. Often completed by crowdsourcing, these benchmarks could never again accurately recover those poses.

Later attempts to improve directional awareness [16, 17, 25, 34, 49, 60] have recognized this problem but failed to fully address it. SQA3D [34] provides a text description of an ego situation and a QA pair that requires spatial reasoning. While the text description implicitly conveys the ego pose, this approach introduces textual ambiguity and is not practically meaningful in real-world applications, where ego pose descriptions are not always available. View2Cap [60] proposes the Situation Grounding module, which takes in a frustum-cropped partial point cloud and regresses the associated camera pose. However, pose regression is far from textual reasoning with direction awareness, thus its effect is limited. Scene-LLM [16] enhances egocentric awareness using a two-step inference scheme, with first an egocentric frustum-cropped point cloud and then a scene-level point cloud. However, this approach is unnecessarily complex and uses proprietary code and data. Furthermore, these methods all set new benchmarks rather than altering existing ones, which still suffer from the problem of ill-posed directional definition.

2.2. 3D LMMs

2D-based 3D LMMs. Based on powerful 2D-pretrained networks, *e.g.*, CLIP [42], DINOv2 [39], and LLAVA-Video [62], this line of research focuses on projecting RGB-D image features back to 3D for understanding [16, 19, 53, 63]. 3D-LLM [19] explored a mixture of projection methods, including direct projection, SLAM [29], and NeRF [35]. Video-3D-LLM [63] and Spatial-MLLM [53] parse the scene as an RGB-D video with pretrained video- or vision-language-models (VLMs). These methods introduce implicit information of ego pose in the images, therefore can be seen as natural implementations of our method.



(a) Point cloud, camera poses and objects (b) Frustum-object intersection check (c) Z buffer visibility check (d) Pose list (saved)

Figure 2. **The offline data generation pipeline for PoseRecover**. (a) Object annotations and camera poses are obtained from ScanNet-v2 [12]. Camera poses and objects are downsampled for visibility. Zoom in for details. (b) PoseRecover exhaustively calculates the intersection rates between objects and camera frustums. (c) Visibility of the intersection is further validated with a z-buffer. (d) These intersection rates are saved and later sampled during training or inference to supplement ego poses to models.

Detector-based 3D LMMs. To reconcile between computational complexity and loss of task-centric information, these methods choose 3D detection [45] or instance segmentation [44] to encode objects as LLM input tokens [9, 18, 21, 22, 50, 64]. For example, Chat-Scene [21] extracts object-centric features via a pretrained detector and passes these object features as tokens using a unified object identifier token. However, these methods are inherently limited by the detector capability and ignore background information, which is essential for reasoning in real scenarios.

3D-backbone-based 3D LMMs. This line of method processes the scene-level point cloud in a single forward pass which is then compressed and sent to LLM for reasoning [8, 14, 23]. LL3DA [8] encodes scene-level features through a Q-Former [31] for natural human interaction. Inspired by superpoint transformers [28, 30, 46], 3D-LLAVA [14] completes point cloud processing and compression simultaneously using the cluster centers, *i.e.*, superpoints, as tokens. Due to natural processing of scene contexts, these methods achieve state-of-the-art performance but are highly contingent on the point cloud encoder design.

3. Data Pipeline

Existing point cloud benchmarks, including ScanQA, ScanRefer, Multi3DRefer, SQA3D, and Scan2Cap, contain 40% to 95% direction-critical queries according to our analysis, yet they omit the camera pose information required to determine the ego agent's directional context. As analyzed in Section 1, this omission renders a significant portion of questions ill-posed with respect to egocentric or allocentric direction understanding. To address this, we introduce PoseRecover, a fully automatic pipeline that recovers ego poses for all text–scene pairs. We first introduce the question analysis to establish our motivation, then introduce the offline PoseRecover pipeline as depicted in Figure 2, which matches the question-specific ground-truth object annotation with the camera frustums, and finally introduce two pose selection strategies for online training or inference.

3.1. Question-Level Analysis

We first quantify the degree of directional ambiguity in existing datasets. An open-source large language model (GPT-OSS-20B [38]) is prompted with a series of question–answer pairs and tasked to determine whether answering the question requires explicit lateral direction reasoning (e.g., "left", "behind", "in front of") [51]. We kindly refer readers to C for the full prompt and B.2 for detailed results. This binary judgment establishes the *direction-critical subsets* of existing datasets, which guide the downstream pose recovery stage. Across datasets, we find that between 40% to 95% of the questions require directional reasoning and are therefore ill-posed without ego-pose supervision.

3.2. Pose Supplementation via PoseRecover

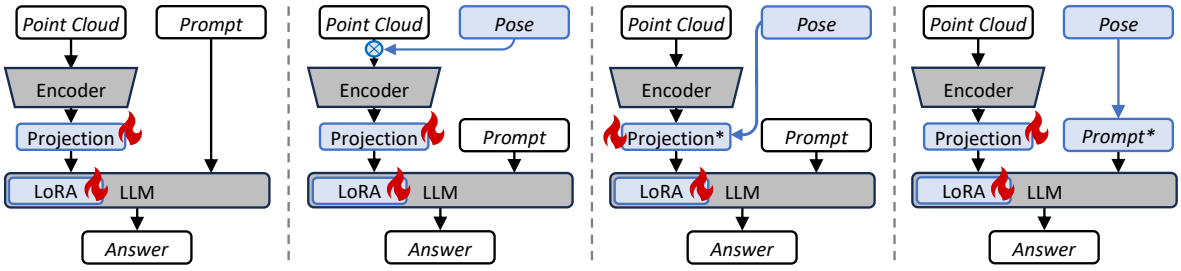
Given the intrinsic and extrinsic parameters of the raw RGB-D sequences in ScanNet-v2, we compute the 3D camera frustums (a visible pyramid with near and far cutoff) and measure their spatial intersection with question-related objects, and save all normalized intersection values into a pose list. Multiple forms of object annotations are supported:

Segmentation-based. For segmentation masks, every point in the point cloud is projected into the image plane using the camera intrinsic and extrinsic, forming a Z-buffer following Equations 1-3:

$$(x_i', y_i', z_i')^\intercal = \mathcal{R}^{-1}(p_i - t), \tag{1}$$

$$(u_i, v_i, 1)^\intercal = \lfloor K(x_i', y_i', z_i')^\intercal / z_i' \rfloor, \tag{2}$$

$$Z_{\mathcal{P}}^{u_i,v_i} = \min_{j|(u_j,v_j)=(u_i,v_i)} (z_j'), \quad \text{s.t. } p_i, p_j \in \mathcal{P}, 0 \le u_i < U, 0 \le v_i < V, \tag{3}$$



(a) Vanilla (b) PoseAlign-Transform (c) PoseAlign-Embed (d) PoseAlign-Prompt

Figure 3. **Three viable designs for PoseAlign**. We explore three mutually exclusive designs to incorporate ego poses into the vanilla model in (a): 1) PoseAlign-Transform that shifts point clouds to the ego reference frame in (b); 2) PoseAlign-Embed that encodes ego poses into point cloud features in (c); 3) PoseAlign-Prompt that integrates ego poses into the text prompt in (d). The projection layer and the LoRA [20] weights of the LLM are trained with instruction-tuning.

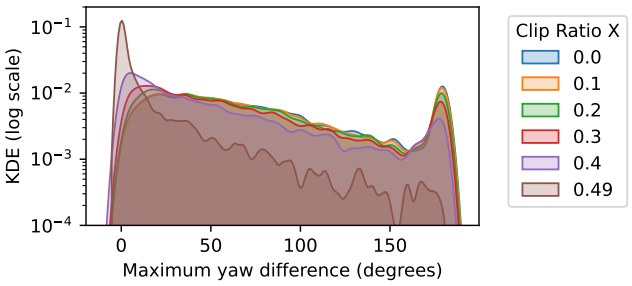


Figure 4. **Effect of the pose clipping.** The KDE [47] of maximum yaw difference among pose candidates rapidly concentrates around zero with increasing clip ratio in ScanQA. Higher clip ratio reduces data variety but boosts pose stability.

where $Z_{\mathcal{P}} \in \mathbb{R}^{U \times V}$ is the Z-buffer, $\mathcal{P} \in \mathbb{R}^{n \times 3}$ is the point cloud, $K, (\mathcal{R}|t)$ are the intrinsic and extrinsic, U, V are the Z-buffer sizes, and $\lfloor \cdot \rfloor$ is the floor operation. A visibility mask is established by depth-buffer comparison, where only the visible points are counted:

$$\phi_{seg} = \frac{1}{|M_{obj}|} \sum_{k \in M_{obj}, 0 \le u_k < U, 0 \le v_k < V} \mathbb{I}\left[z'_k < Z_{\mathcal{P}}^{u_k, v_k} + \delta\right] \tag{4}$$

where M_{obj} is the segmentation mask consisting of point indices for the object, the margin $\delta = 10^{-6}$ tolerates numerical instability, and $\mathbb{I}[\cdot]$ is the Iverson Bracket. The intersection ratio ϕ_{seg} is then defined as the proportion of visible object points lying within the frustum.

Bounding-box-based. As closed-form intersection between bounding box and frustum can be hard to calculate, we apply Monte-Carlo sampling in the box for an estimation of the intersection ratio ϕ_{box} with Equation 5:

$$\phi_{box} = \frac{1}{|\mathcal{P}_{sample}|} \sum_{p \in \mathcal{P}_{sample}} \mathbb{I}[p \in F], \quad \text{s.t. } \mathcal{P}_{sample} = \text{meshgrid}(h, w, l, \Delta) + \epsilon, \tag{5}$$

where F is the frustum region, h, w, l are the bounding box sizes, Δ is a predefined grid size, $\text{meshgrid}(\cdot)$ is a function that generates uniform spatial grid, and $\epsilon \in [0, \Delta)^{\frac{h}{\Delta} \times \frac{w}{\Delta} \times \frac{l}{\Delta}}$ are random noises.

Point-based. For datasets that provide only a single object location [9], we compute the normalized pixel distance ϕ_{point} between the projected object pixel and the image center following Equations 6-7, where smaller distances imply larger intersections:

$$(u_{obj}, v_{obj}, 1)^\intercal = \lfloor K\mathcal{R}^{-1}(p_{obj} - t)/z'_{obj} \rfloor \tag{6}$$

$$\phi_{point} = \begin{cases} 1 - \frac{\left\| (\frac{U}{2} - u_{obj}, \frac{V}{2} - v_{obj}) \right\|_2}{\left\| (\frac{U}{2}, \frac{V}{2}) \right\|_2} & \text{if } 0 \le u_{obj} < U, \\ & \text{and } 0 \le v_{obj} < V \\ 0 & \text{else} \end{cases} \tag{7}$$

where $p_{obj} \in \mathbb{R}^3$ is the object center coordinate, and z'_{obj} is the camera-view depth following Equation 1.

These intersection ratios are exhaustively calculated between all camera poses and objects, yielding a camera-object intersection matrix for each scene as depicted in Figure 2(d). Thanks to vectorized implementation, the visibility check costs less than 40 minutes for ScanNet-v2. We refer readers to section A.1 for more discussions.

3.3. Pose Selection Strategy

During the forward pass, we retrieve the question-specific column of the camera-object intersection matrix using the ground-truth object annotation in existing datasets. Among the candidate camera poses with non-zero intersection, we evaluate two strategies: (1) **Top**: Selecting the pose with the highest intersection ratio, and (2) **Clip**: Random sampling after discarding the top and bottom $X\%$ non-zero scores, where X is the clip ratio. Empirically, the second method with a mediocre X performs better, as it reduces outlier cases like opposite views with 180° yaw difference among candidate poses while still retaining decent variety, as shown in Figure 4. This final camera pose serves as the ego reference frame for the language query, enabling an authentic definition of direction semantics across all datasets.

4. Method

Owing to the absence of ego-pose information in existing benchmarks, current 3D LMMs are not designed to accept pose data as direct input. To bridge this gap, we design and compare three PoseAlign variants as displayed in Figure 3, each augmenting a common backbone (e.g., 3D-LLAVA-style architecture) with ego-pose information through a different pathway: (1) point cloud transformation (PoseAlign-Transform), (2) projection-layer positional embedding (PoseAlign-Embed), and (3) pose prompt injection (PoseAlign-Prompt). All variants finetune the multimodal projector and the LLM (if applicable) jointly under a LoRA-based adaptation scheme.

4.1. PoseAlign-Transform

This variant directly transforms the input point cloud into the recovered camera coordinate frame:

$$(\mathcal{P}_{aligned}|1)^\intercal = \mathcal{U}\mathcal{T}^{-1}(\mathcal{P}|1)^\intercal,$$
$$\text{s.t. } \mathcal{T} = \begin{pmatrix} \mathcal{R} & t \\ 0 & 1 \end{pmatrix}, \mathcal{U} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (8)$$

where $\mathcal{P}$ is the point cloud, $(\mathcal{R}|t)$ is the camera extrinsic matrix, and $\mathcal{U}$ transforms right-down-front camera coordinate to front-left-up coordinate for alignment with the pretrained point cloud encoder. This operation aligns the spatial distribution of scene points with the ego-viewing direction, ensuring that "left" and "right" are consistently expressed in the egocentric frame. Because the transformation preserves all geometric relationships, no architectural modifications are required downstream. This approach achieves the best empirical performance and is therefore used as our default method.

4.2. PoseAlign-Embed

Here, we keep the point cloud unchanged but modulate the projection layer of the multimodal projector with an embedding of the 6-DoF pose:

$$f_{aligned} = f + \text{MLP}(\text{encode}(\mathcal{R}, t, \mathcal{P}_f)), \quad (9)$$

where we encode a series of manual geometric features of the coordinates of the tokens $\mathcal{P}_f \in \mathbb{R}^{\#\text{tokens}\times 3}$ into the original feature vectors $f \in \mathbb{R}^{\#tokens\times d}$ using an MLP. This introduces pose awareness at the feature-projection stage without altering geometry.

4.3. PoseAlign-Prompt

In this variant, the pose is serialized as numeric tokens and prepended to the textual prompt in the following format: `f'position{t}, up{R[:,2]}, front{R[:,0]}, left{R[:,1]}'`. This allows the LLM to reason explicitly about spatial direction in text. Despite its conceptual simplicity, the numeric-token format introduces numeric tokenization overhead and inconsistent positional grounding, resulting in modest performance deterioration. More discussion is available in Section A.2

5. Results

5.1. Experiment Setup

Datasets. We conduct experiments on a series of datasets including ScanRefer [5], Multi3DRefer [61], ScanQA [2], SQA3D [34], and Scan2Cap [10]. These datasets are based on the ScanNet-v2 dataset [12], consisting of 1,513 indoor scenes, among which 1,201 are used for training and 312 are used for validation.

Metrics. We adopt both traditional metrics and new metrics of LLM-as-judge [51] for a comprehensive assessment suite. The LLM-as-judge Accuracy (**L-A**) adopts GPT-OSS-20B [38] or GPT-5-mini [37] (see section B.2) as the base model. The prompt (see section C) is based on the SimpleQA benchmark [51]. L-A is consistent under different paraphrases of the same meaning, focusing on the invariant meanings instead of more random wording choices [51]. We also report traditional metrics including CiDEr (**C**), BLEU-4 (**B-4**), METEOR (**M**), and ROUGE-L (**R**) for QA datasets, where Scan2Cap metrics are restricted to instances with ≥ 0.5 IoU (**@0.5**) except L-A. We follow common practice to use thresholded Accuracy@0.5 (**A@0.5**), F1-score@0.5 (**F1@0.5**), and mean intersection over union (mIoU) for ScanRefer [5] and Multi3DRefer [61].

Baselines and implementation. We adopt four baseline models from an array of distinct architectures to apply PoseAlign modifications, including LL3DA [8], LL3DA-SONATA, Chat-Scene [21], and 3D-LLAVA [14]. LL3DA-SONATA, referred to as LL3DA-S, is a variant of LL3DA whose point cloud encoder is switched to SONATA[54]. All modifications belong to PoseAlign-Transform (*PoseAlign-T*) except for Chat-Scene, which employs PoseAlign-Embed (*PoseAlign-E*) due to its reliance on precomputed point cloud embeddings.

Training specifications. All comparison methods are trained with only the instruction-tuning stage using respective original training schemes and datasets. For Chat-Scene [21] and 3D-LLAVA [14], the LLM LoRA [20] and the projection layer are trained while the 3D encoder is frozen. For LL3DA [8], only the Q-Former [31] is trained. Freezing the encoder prevents the formation of preference over objects on the positive x-axis, ensuring fairness in segmentation- or detection-based assessments. All point

Model	Modality	ScanRef (val)		Multi3DRef (val)		ScanQA (val)					Scan2Cap (val)				
		A@0.5	mIoU	F1@0.5	mIoU	C	B-4	M	R	L-A	C@0.5	B-4@0.5	M@0.5	R@0.5	L-A
Specialist Models:															
ScanQA [2]	PC	-	-	-	-	64.9	10.1	13.1	33.3	-	-	-	-	-	-
3D-VLP [27]	PC	-	-	-	-	67.0	11.2	13.5	34.5	-	54.9	32.3	24.8	51.5	-
3D-VisTA [64]	PC	-	-	-	-	69.6	10.4	13.9	45.7	-	61.6	34.1	26.8	55.0	-
Scan2Cap [10]	PC	-	-	-	-	-	-	-	-	-	39.1	23.3	22.0	44.8	-
MORE [26]	PC	-	-	-	-	-	-	-	-	-	40.9	22.9	21.7	44.4	-
SpaCap3D [48]	PC	-	-	-	-	-	-	-	-	-	44.0	25.3	22.3	45.4	-
D3Net [6]	PC	-	-	-	-	-	-	-	-	-	46.1	30.3	24.4	51.7	-
UniT3D [11]	PC	-	-	-	-	-	-	-	-	-	46.7	27.2	21.9	46.0	-
3DJCG [3]	PC	-	-	-	-	-	-	-	-	-	49.5	31.0	24.2	50.8	-
Vote2Cap-DETR [7]	PC	-	-	-	-	-	-	-	-	-	61.8	34.5	26.2	54.4	-
TGNN [24]	PC	-	27.8	-	-	-	-	-	-	-	-	-	-	-	-
M3DRef-CLIP [61]	PC	-	35.7	-	32.6	-	-	-	-	-	-	-	-	-	-
X-RefSeg3D [41]	PC	-	29.9	-	-	-	-	-	-	-	-	-	-	-	-
3D-STMN [52]	PC	-	39.5	-	-	-	-	-	-	-	-	-	-	-	-
Finetuned Generalists:															
3D-LLM [19]	PC+I	-	-	-	-	69.4	12.0	14.5	35.7	-	-	-	-	-	-
Scene-LLM [16]	PC+I	-	-	-	-	80.0	12.0	16.8	40.0	-	-	-	-	-	-
SegPoint [18]	PC	-	41.7	-	36.1	-	-	-	-	-	-	-	-	-	-
Generalists:															
LEO [22]	PC	-	-	-	-	101.4	13.2	20.0	49.2	-	72.4	**38.2**	**27.9**	**58.1**	-
Scene-LLM [16]	PC	-	-	-	-	80.0	11.7	15.8	35.9	-	-	-	-	-	-
Grounded 3D-LLM [9]	PC	-	-	-	-	72.7	13.4	-	-	-	70.6	35.5	-	-	-
PoseRecover Benchmark:															
LL3DA [8]	PC	-	-	-	-	75.3	13.1	15.2	36.7	34.7	61.6	35.3	25.6	54.3	16.0
LL3DA + PoseAlign-T	PC	-	-	-	-	76.7	13.9	15.6	37.1	35.6	E: Rotated box				
LL3DA-S [8, 54]	PC	-	-	-	-	75.0	12.6	15.1	37.0	34.8	E: No detector				
LL3DA-S + PoseAlign-T	PC	-	-	-	-	76.5	13.2	15.6	36.8	35.9					
Chat-Scene [21]	PC+I	46.4	-	49.1	-	85.6	15.6	17.8	40.4	43.9	74.0	34.5	26.8	56.4	26.6
Chat-Scene + PoseAlign-E	PC+I	46.9	-	**50.2**	-	87.2	15.1	18.1	41.1	44.7	75.5	35.0	27.1	56.7	26.9
3D-LLAVA [14]	PC	41.5	42.6	-	48.1	95.4	16.3	18.9	44.6	45.7	**77.4**	36.4	26.9	57.4	28.1
3D-LLAVA + PoseAlign-T	PC	**58.7**	**55.4**	-	**54.3**	**99.8**	**17.3**	**19.7**	**46.5**	**47.3**	76.1	37.1	27.1	57.6	**31.4**

Table 1. **Cross-dataset performance comparison** on multiple 3D vision-language tasks. 'PC' and 'I' represent point cloud and image modalities, respectively. Major metrics are highlighted with gray background. Performance on PoseRecover benchmark may differ from those in the original papers due to retraining with lower batch sizes. Baselines in PoseRecover benchmark are comparable with all methods because they do not use pose information, while our modifications are comparable within the benchmark due to additional pose input.

cloud data augmentations have been disabled for PoseAlign variants to precisely align the egocentric point cloud coordinates, which include random rotation, flipping, jittering, and scaling. We deem that data augmentation is compensated by the randomness of ego poses during pose selection, where PoseAlign-Transform and -Embed can be seen as exotic implementations of rotation-shift augmentation.

5.2. Main Results

In this experiment, we validate the effectiveness of PoseRecover and PoseAlign designs with an array of specialist and generalist 3D LMMs on the ScanRefer, Multi3DRefer, ScanQA, and Scan2Cap datasets in Table 1. All modified models on the PoseRecover Benchmark get the additional pose input retrieved using the PoseRecover script. The performance of LEO [22] is colored in gray as it utilizes the ground truth object features, which forms a slightly stronger setting as discussed in section A.2. LL3DA has poor compatibility with rotated boxes, and SONATA is a feature extractor with no detector component, so some results on Scan2Cap are omitted. Methods enhanced with PoseAlign receive performance boosts broadly across datasets and metrics, where 3D-LLAVA modified with PoseAlign-Transform hits the highest overall performance of 55.4%(Δ30.0%) ScanRefer mIoU, 54.3%(Δ12.9%) Multi3DRefer mIoU, 47.3%(Δ3.5%) and 31.4%(Δ11.7%) LLM-as-judge accuracy on ScanQA and Scan2Cap, respectively. We conclude that PoseAlign unleashes the full capabilities of current 3D LMMs, which were once bounded by the ill-posed problem definition in existing benchmarks.

Is PoseAlign a good addition to existing tasks? We focus on the PoseRecover benchmark to validate the effectiveness of PoseAlign. Comparing all four baselines with our respective modifications, performance boosts can be widely seen across QA datasets with an average bump of 1% and 0.6% on ScanQA and Scan2Cap, respectively, including CiDEr, BLEU-4, METEOR, and ROUGE-L on all four backbones. The LLM-as-judge accuracy boosts are greater for PoseAlign-T variants (35.6%(Δ2.6%) on

Design	ScanRef mIoU↑	Multi3DRef mIoU↑	ScanQA L-A↑	Scan2Cap L-A↑
Baseline	42.6	48.1	45.7	28.1
Baseline PoseAlign-T(Top)	37.5	41.5	43.0	23.5
Random Pose	39.0	44.3	44.7	25.8
PoseAlign-T(Clip X=0.3)	55.4	54.3	47.3	**31.4**
PoseAlign-T(Top)	**68.5**	**60.2**	**47.5**	29.7
PoseAlign-E(Top)	43.2	49.3	43.4	23.5
PoseAlign-P(Top)	44.2	49.4	44.9	28.1

Table 2. **Ablation experiment on 3D-LLAVA**. 'Baseline PoseAlign-T' is the performance of the baseline model on PoseAlign-T data, where the input point cloud is transformed to the camera location following Equation 8. 'Random Pose' uses random camera poses instead of those found by PoseRecover.

LL3DA, 35.9%(Δ3.2%) on LL3DA-S, 47.3%(Δ3.5%) on 3D-LLAVA) than PoseAlign-E variants (44.7%(Δ1.8%) on Chat-Scene), signaling a better realization of direction awareness. On refer segmentation benchmarks, both Chat-Scene and 3D-LLAVA receive performance boosts among which 3D-LLAVA receives the largest +12.8% and +6.2% mIoU on ScanRefer and Multi3DRefer, respectively. Such performance is astounding given that the point cloud segmentor (*i.e.,* 3D encoder) is frozen, which attributes all improvements to improved '<SEG>' token quality generated by the LLM [14]. We conclude that, given their potential to improve performance, ego-poses are undoubtedly good additions to current 3D LMM pipelines.

Is LLM-as-judge a good metric? As discussed in section 5.1, a good language-task metric should be consistent under verbal randomness as confirmed by Table 1. LLM-as-judge accuracy is consistently higher across all our modifications than the respective baselines, while other metrics tend to fluctuate. It is also robust to the judgment model, as shown in Tables 1 and 6. We conclude that LLM-as-judge is a sufficiently good metric for this task.

5.3. Other Results

Ablation on 3D-LLAVA. We compare the effect of different PoseAlign designs on 3D-LLAVA in Table 2. Incorporating ego-pose information clearly benefits spatial reasoning, as the proposed PoseAlign-Transform consistently outperforms all other variants on refer segmentation tasks, improving mIoU by up to 68.5%(Δ60.8%) on ScanRefer and 60.2%(Δ25.2%) on Multi3DRefer. The Clip strategy ($X = 0.3$) for PoseAlign-Transform further stabilizes performance metrics across benchmarks by purging inconsistent views. In contrast, encoding poses as text (PoseAlign-Prompt) or projection features (PoseAlign-Embed) offers limited or inconsistent gains. Additionally, test performance of a well-trained 3D-LLAVA baseline model on PoseAlign-Transform data (Baseline PoseAlign-T) significantly degrades all metrics, which gives two important insights: 1) the segmentation model never uses the pose data as a shortcut because it is frozen, and 2) the current performance increments are brought purely by better '<SEG>' tokens. Using random poses also decreases model performance slightly below baseline, showcasing the importance of PoseRecover design. We conclude that PoseAlignTransform with the Clip strategy where X = 0.3 is the most balanced and robust way of introducing directional awareness into 3D LMMs.

X	ScanRef mIoU↑	Multi3DRef mIoU↑	ScanQA L-A↑	Scan2Cap L-A↑
0.0	55.1	54.2	**47.3**	30.1
0.1	55.2	**54.3**	46.6	30.8
0.2	54.8	**54.3**	**47.3**	31.2
0.3	**55.4**	**54.3**	**47.3**	**31.4**
0.4	**55.4**	**54.3**	46.9	30.9
0.45	55.2	**54.3**	46.5	**31.4**
0.49	55.1	53.9	46.9	31.1

Table 3. **Parameter tuning experiment** for Clip Ratio X of PoseAlign-Transform on 3D-LLAVA.

Parameter tuning. Table 3 analyzes the sensitivity of the Clip Ratio X in PoseAlign-Transform, which discards the top and bottom $X\%$ of views based on object–frustum intersection during selection of the mission-critical pose. Despite theoretical concerns about opposite-view errors discussed in Figure 4, the model performance remains stable across datasets for $0.0 \leq X \leq 0.49$ with a slight hill in the mediocre Clip Ratio ranging from $0.2 \leq X \leq 0.4$, demonstrating the robustness of the method to this hyperparameter. This is because a smaller variance in viewing directions indeed enhances pose coherence but destroys data diversity, while higher variance in viewing directions adds noise to the model but also serves as effective data augmentation as discussed in section 5.1. Moderate clipping (around $X = 0.3$) yields the best balance, slightly improving results on ScanQA and Scan2Cap to 47.3% and 31.4%, respectively. Therefore, $X = 0.3$ is adopted as the default value for all other experiments.

Performance on direction-critical subsets. Table 4 compares the baseline 3D-LLAVA and its pose-aware variant (PoseAlign-Transform) on full validation sets, direction-critical subsets, and their respective complementary direction-agnostic subsets of ScanQA and Scan2Cap. Firstly, the direction-critical subsets identified via LLMs exhibit a notable performance drop in the baseline model, confirming that existing 3D LMMs struggle with questions requiring spatial orientation. Applying PoseAlign-Transform consistently improves all metrics, especially the LLM-as-judge accuracy, narrowing down the gap between the two subsets. The gains are especially pronounced on ScanQA, where PoseAlign-T achieves 96.1(Δ4.9%) CiDEr

Model	Evaluated on	ScanQA (46.7% Direction-critical)					Scan2Cap (89.7% Direction-critical)				
		C↑	B-4↑	M↑	R↑	L-A↑	C↑	B-4↑	M↑	R↑	L-A↑
3D-LLAVA	Full Val set	95.4	16.3	18.9	44.6	45.7	77.4	36.4	26.93	57.4	28.1
	Direction-critical subset	91.6	18.0	17.8	41.0	37.8	76.8	36.8	26.94	57.4	28.0
	Complementary subset	97.6	11.9	20.1	47.7	52.4	82.7	32.6	26.88	56.9	29.6
3D-LLAVA+ PoseAlign-T	Full Val set	99.8(+4.4)	17.3(+1)	19.7(+0.8)	46.5(+1.9)	47.3(+1.6)	76.1(-1.3)	37.1(+0.7)	27.11(+0.18)	57.6(+0.2)	31.4(+3.3)
	Direction-critical subset	96.1(+4.5)	19.6(+1.6)	18.7(+0.9)	43.1(+2.1)	40.3(+2.5)	75.9(-0.9)	37.5(+0.7)	27.12(+0.18)	57.7(+0.3)	31.3(+3.3)
	Complementary subset	101.8(+4.2)	11.5(-0.4)	20.8(+0.7)	49.4(+1.7)	53.3(+0.9)	77.8(-4.9)	33.4(+0.8)	27.10(+0.22)	56.9(+0.0)	32.6(+3.0)

Table 4. Performance on direction-critical question subset.

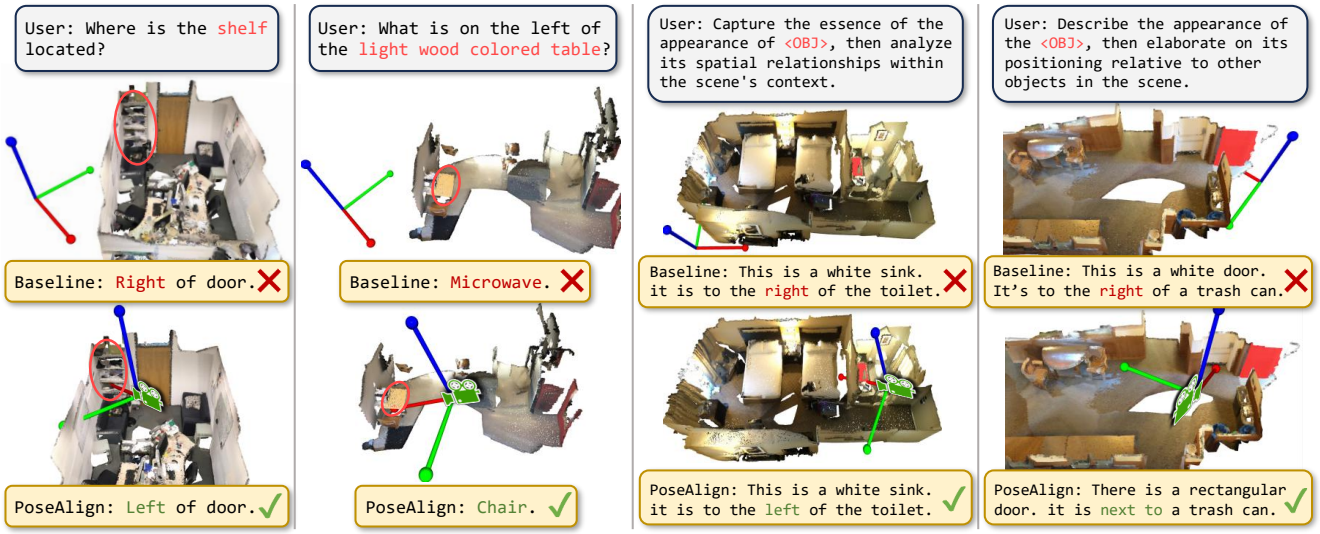


Figure 5. **Qualitative results of direction-critical questions** for 3D-LLAVA baseline (top row) and PoseAlign-Transform (bottom row). The XYZ axes of the world coordinate frame are colored with red, green, and blue, respectively. The baseline paradigm uses default world coordinates of ScanNet-v2, which are non-informative. Instead, the PoseAlign paradigm aligns the coordinate frame to the recovered ego pose, providing an anchor for robust spatial reasoning. Red text highlights wrong answers and green text highlights correct answers.

and 40.3%(Δ6.6%) LLM-as-judge accuracy. While the direction-critical subset is the best-effort judgment of a powerful LLM, some direction-critical questions may still leak into the complementary subset, causing performance to rise slightly there as well. We conclude that our PoseAlign design indeed enhances directional reasoning based on the substantial improvements, especially on the direction-critical subsets.

5.4. Qualitative Results

As displayed in Figure 5, baseline models struggle to perceive spatial relationships, where the default world coordinate only adds confusion and complicates spatial reasoning. In contrast, PoseAlign effectively avoids directional ambiguity thanks to the recovered ego pose, enabling robust and consistent spatial reasoning for 3D LMMs.

5.5. Limitations

Despite being lightweight and generic, our method has several limitations, including assuming SLAM accuracy, the variety of training views, and a static environment. We kindly refer readers to section A.3 for a detailed discussion.

6. Conclusion

We have proposed PoseRecover and PoseAlign, a pair of lightweight yet powerful techniques that transform existing 3D LMMs towards a rigorously defined direction-aware paradigm. To fix existing benchmarks, PoseRecover automatically reconstructs mission-critical ego poses by aligning object annotations with camera frustums derived from ScanNet RGB-D extrinsics, effectively correcting ill-posed direction-critical queries across existing datasets. Building upon these recovered poses, PoseAlign enables persistent directional awareness for 3D LMMs by transforming the input point cloud or encoded point cloud embeddings into the camera reference frame, thereby resolving ambiguity in ego direction and substantially improving spatial reasoning performance. Extensive experiments on multiple benchmarks and model architectures prove that PoseRecover and PoseAlign together unlock the latent potential of current 3D LMM architectures, achieving consistent performance gains without retraining the point cloud encoder. We advocate this framework as a simple, generic, and effective paradigm for advancing direction-aware 3D-language understanding of LMMs.

Acknowledgments

This research is supported by Networked Exchange, United Strength for Stronger Partnerships between Japan and ASEAN (NEXUS), a collaboration program between the Agency for Science, Technology and Research (A*STAR), Singapore (Grant No. R2416IR138), and the Japan Science and Technology Agency (JST), Japan (Grant No. JPMJNX25CA). Weihao Xuan is supported by the RIKEN Junior Research Associate (JRA) Program.

References

[1] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. ReferIt3D: Neural listeners for fine-grained 3D object identification in real-world scenes. In *Proceedings of the European Conference on Computer Vision*, pages 422–440. Springer, 2020. 1, 2

[2] 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. 1, 2, 5, 6

[3] Daigang Cai, Lichen Zhao, Jing Zhang, Lu Sheng, and Dong Xu. 3DJCG: A unified framework for joint dense captioning and visual grounding on 3D point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16464–16473, 2022. 6

[4] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3D: Learning from RGB-D data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017. 2

[5] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. ScanRefer: 3D object localization in RGB-D scans using natural language. In *Proceedings of the European Conference on Computer Vision*, pages 202–221. Springer, 2020. 1, 2, 5

[6] Dave Zhenyu Chen, Qirui Wu, Matthias Nießner, and Angel X Chang. D^3Net: A unified speaker-listener architecture for 3D dense captioning and visual grounding. In *Proceedings of the European Conference on Computer Vision*, pages 487–505. Springer, 2022. 6

[7] Sijin Chen, Hongyuan Zhu, Xin Chen, Yinjie Lei, Gang Yu, and Tao Chen. End-to-end 3D dense captioning with Vote2Cap-DETR. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11124–11133, 2023. 6, 2

[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. 3, 5, 6, 2

[9] Yilun Chen, Shuai Yang, Haifeng Huang, Tai Wang, Runsen Xu, Ruiyuan Lyu, Dahua Lin, and Jiangmiao Pang. Grounded 3D-LLM with referent tokens. *arXiv preprint arXiv:2405.10370*, 2024. 3, 4, 6

[10] Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X Chang. Scan2Cap: Context-aware dense captioning in RGB-D scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3193–3203, 2021. 1, 2, 5, 6

[11] Zhenyu Chen, Ronghang Hu, Xinlei Chen, Matthias Nießner, and Angel X Chang. UNIT3D: A unified transformer for 3D dense captioning and visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18109–18119, 2023. 6

[12] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 5828–5839, 2017. 2, 3, 5

[13] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36:49250–49267, 2023. 3

[14] 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 Computer Vision and Pattern Recognition Conference*, pages 3772–3782, 2025. 1, 3, 5, 6, 7, 2

[15] Christian Freksa, Christopher Habel, and Karl F Wender. *Spatial cognition: An interdisciplinary approach to representing and processing spatial knowledge*. Springer Science & Business Media, 1998. 1

[16] Rao Fu, Jingyu Liu, Xilun Chen, Yixin Nie, and Wenhan Xiong. Scene-LLM: Extending language model for 3D visual understanding and reasoning. *arXiv preprint arXiv:2403.11401*, 2024. 2, 6

[17] Zoey Guo, Yiwen Tang, Ray Zhang, Dong Wang, Zhigang Wang, Bin Zhao, and Xuelong Li. ViewRefer: Grasp the multi-view knowledge for 3D visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15372–15383, 2023. 2

[18] Shuting He, Henghui Ding, Xudong Jiang, and Bihan Wen. SegPoint: Segment any point cloud via large language model. In *Proceedings of the European Conference on Computer Vision*, pages 349–367. Springer, 2024. 1, 3, 6

[19] 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. 2, 6

[20] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *Proceedings of the International Conference on Learning Representations*, 2022. 4, 5

[21] Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, et al. Chat-Scene: Bridging 3D scene and large language models with object identifiers.

Advances in Neural Information Processing Systems, 37: 113991–114017, 2024. 3, 5, 6, 2

[22] Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3D world. *arXiv preprint arXiv:2311.12871*, 2023. 1, 3, 6

[23] Kuan-Chih Huang, Xiangtai Li, Lu Qi, Shuicheng Yan, and Ming-Hsuan Yang. Reason3D: Searching and reasoning 3D segmentation via large language model. In *Proceedings of the International Conference on 3D Vision*, 2025. 3

[24] Pin-Hao Huang, Han-Hung Lee, Hwann-Tzong Chen, and Tyng-Luh Liu. Text-guided graph neural networks for referring 3D instance segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1610–1618, 2021. 6

[25] Shijia Huang, Yilun Chen, Jiaya Jia, and Liwei Wang. Multi-view transformer for 3D visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15524–15533, 2022. 1, 2

[26] Yang Jiao, Shaoxiang Chen, Zequn Jie, Jingjing Chen, Lin Ma, and Yu-Gang Jiang. MORE: Multi-order relation mining for dense captioning in 3D scenes. In *Proceedings of the European Conference on Computer Vision*, pages 528–545. Springer, 2022. 6

[27] Zhao Jin, Munawar Hayat, Yuwei Yang, Yulan Guo, and Yinjie Lei. Context-aware alignment and mutual masking for 3d-language pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10984–10994, 2023. 6

[28] Maxim Kolodiazhnyi, Anna Vorontsova, Anton Konushin, and Danila Rukhovich. OneFormer3D: One transformer for unified point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20943–20953, 2024. 3

[29] Jatavallabhula Krishna, Murthy, Iyer Ganesh, and Paull Liam. gradSLAM: dense SLAM meets automatic differentiation. *arXiv preprint arXiv:1910.10672*, 2019. 2

[30] Xin Lai, Yuhui Yuan, Ruihang Chu, Yukang Chen, Han Hu, and Jiaya Jia. Mask-attention-free transformer for 3D instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3693–3703, 2023. 3

[31] 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 *Proceedings of the International Conference on Machine Learning*, pages 19730–19742. PMLR, 2023. 3, 5

[32] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in Neural Information Processing Systems*, 36:34892–34916, 2023. 2

[33] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 2

[34] Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. SQA3D: Situated question answering in 3D scenes. In *Proceedings of the International Conference on Learning Representations*, 2023. 1, 2, 5

[35] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[36] Riku Murai, Eric Dexheimer, and Andrew J Davison. MASt3R-SLAM: Real-time dense SLAM with 3d reconstruction priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16695–16705, 2025. 2

[37] OpenAI. ChatGPT (GPT-5), 2025. Large language model. 5

[38] OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025. 3, 5

[39] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2

[40] Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, Thomas Funkhouser, et al. OpenScene: 3D scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 815–824, 2023. 1

[41] Zhipeng Qian, Yiwei Ma, Jiayi Ji, and Xiaoshuai Sun. X-RefSeg3D: Enhancing referring 3D instance segmentation via structured cross-modal graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4551–4559, 2024. 1, 6

[42] 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 *Proceedings of the International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2, 1

[43] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied AI research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9339–9347, 2019. 1

[44] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3D: Mask transformer for 3D semantic instance segmentation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 8216–8223. IEEE, 2023. 1, 3, 2

[45] Yichao Shen, Zigang Geng, Yuhui Yuan, Yutong Lin, Ze Liu, Chunyu Wang, Han Hu, Nanning Zheng, and Baining Guo. V-DETR: DETR with vertex relative position encoding for 3D object detection. In *Proceedings of the International Conference on Learning Representations*, 2024. 1, 3

[46] Jiahao Sun, Chunmei Qing, Junpeng Tan, and Xiangmin Xu. Superpoint transformer for 3D scene instance segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2393–2401, 2023. 3

[47] Berwin A Turlach. Bandwidth selection in kernel density estimation: A review. Technical report, Center for Operations Research and Econometrics and Institut de Statistique, 1993. 4

[48] Heng Wang, Chaoyi Zhang, Jianhui Yu, and Weidong Cai. Spatiality-guided transformer for 3D dense captioning on point clouds. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2022. 6

[49] Tai Wang, Xiaohan Mao, Chenming Zhu, Runsen Xu, Ruiyuan Lyu, Peisen Li, Xiao Chen, Wenwei Zhang, Kai Chen, Tianfan Xue, et al. EmbodiedScan: A holistic multi-modal 3D perception suite towards embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19757–19767, 2024. 1, 2

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

[51] Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024. 3, 5

[52] Changli Wu, Yiwei Ma, Qi Chen, Haowei Wang, Gen Luo, Jiayi Ji, and Xiaoshuai Sun. 3D-STMN: Dependency-driven superpoint-text matching network for end-to-end 3D referring expression segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5940–5948, 2024. 1, 6

[53] Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-MLLM: Boosting MLLM capabilities in visual-based spatial intelligence. In *Advances in Neural Information Processing Systems*, 2025. 2

[54] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22193–22204, 2025. 5, 6, 2, 3

[55] Xu Yan, Zhihao Yuan, Yuhao Du, Yinghong Liao, Yao Guo, Shuguang Cui, and Zhen Li. Comprehensive visual question answering on point clouds through compositional scene manipulation. *IEEE Transactions on Visualization and Computer Graphics*, 30(12):7473–7485, 2023. 1, 2

[56] Zhengyuan Yang, Songyang Zhang, Liwei Wang, and Jiebo Luo. SAT: 2d semantics assisted training for 3D visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1856–1866, 2021. 1

[57] Zhihao Yuan, Xu Yan, Yinghong Liao, Ruimao Zhang, Sheng Wang, Zhen Li, and Shuguang Cui. InstanceRefer: Cooperative holistic understanding for visual grounding on point clouds through instance multi-level contextual referring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1791–1800, 2021. 1, 2

[58] Zhihao Yuan, Xu Yan, Zhuo Li, Xuhao Li, Yao Guo, Shuguang Cui, and Zhen Li. Toward explainable and fine-grained 3D grounding through referring textual phrases. *arXiv preprint arXiv:2207.01821*, 2022. 1

[59] Zhihao Yuan, Xu Yan, Yinghong Liao, Yao Guo, Guanbin Li, Shuguang Cui, and Zhen Li. X-Trans2Cap: Cross-modal knowledge transfer using transformer for 3D dense captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8563–8573, 2022. 1

[60] Zhihao Yuan, Yibo Peng, Jinke Ren, Yinghong Liao, Yatong Han, Chun-Mei Feng, Hengshuang Zhao, Guanbin Li, Shuguang Cui, and Zhen Li. Empowering large language models with 3D situation awareness. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19435–19445, 2025. 1, 2

[61] Yiming Zhang, ZeMing Gong, and Angel X Chang. Multi3drefer: Grounding text description to multiple 3D objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15225–15236, 2023. 1, 5, 6

[62] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024. 2

[63] Duo Zheng, Shijia Huang, and Liwei Wang. Video-3D LLM: Learning position-aware video representation for 3D scene understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8995–9006, 2025. 2

[64] Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 3D-VisTA: Pre-trained transformer for 3D vision and text alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2911–2921, 2023. 3, 6