Direction-aware 3D Large Multimodal Models

Supplementary Material

A. Additional Notes

A.1. Explanations

Ill-posed questions without poses. At first glance, it may seem odd that questions involving left/right directions are ill-posed when ego pose is not present, especially to researchers in the fields of image or VLM research. A key distinction between images and 3D point clouds is that images are captured from an ego pose, which inherently defines left/right as the image left/right; However, a room-level point cloud is reconstructed using RGB-D video or even with 360 degree view LiDARs. Once the points are put into free 3D space, the concept of an ego reference frame disappears, now that the ego pose can no longer be trivially inferred from the room-level point cloud data itself. However, existing 3D LMMs receive only such a room-level point cloud and are asked to answer questions containing directional words, which is an ill-posed problem we aim to conquer in this paper.

From implicit ego pose inference to explicit ego pose input. We could infer from past research that they want the 3D LMM to implicitly infer its ego position first, then derive the related answers based on that inferred ego pose. However, this regime is both (1) inherently ill-posed, and (2) adds unnecessary complexity for the 3D LMM to infer its ego position, while the robot's ego pose is trivially available in realistic applications such as SLAM-capable embodied intelligence operating indoors. We therefore conclude that adding such pose data as additional input to the 3D LMMs does not change its application potential at all, and such conversion of regime is both practical and theoretically grounded.

From an ego pose trajectory to a mission-critical pose. In practice, even when the ego trajectory is a free lunch, the agent does not always face the target object of the query and the mission-critical pose has to be assessed on-the-fly. However, we argue that such a mission-critical pose should be retrieved with other techniques rather than with the 3D LMM itself, as the task is essentially a cross-modal alignment problem with well-established solutions. For example, the agent can localize the mission-critical area by comparing the text embedding of the queried object with an offline precomputed semantic map of the indoor point cloud, e.g., using CLIP [42] or OpenScene [40]. After that, using our PoseRecover script, the agent can locate the most probable ego position from the ego pose trajectory, which can then be used to reason consistently and unambiguously. In this work, we are more interested in the prior problem of *whether a mission-critical pose retrieved from ground truth is a necessary and beneficial addition to 3D LMM input, to rectify the ill-posed direction definitions*, rather than a full end-to-end implementation; Our work is purely exploratory and there must exist some better way of integrating those poses into a coherent pipeline than with our PoseAlign.

A.2. Discussions

Compatibility with image-based methods. We deem all image-based methods to have implicitly used ego pose trajectory during the camera projection of the input images. Therefore, these image-based methods are already implicit implementations of the PoseAlign-Transform method and do not need further modifications.

Compatibility with SQA3D. Notably, PoseAlign receives negative results on SQA3D as listed in Tables 7-8. This is because among all training datasets, SQA3D is the only dataset that does not provide any object annotations, which means that PoseRecover pipeline cannot determine authentic poses for the questions, yielding random poses instead. This results in a slightly decreased performance compared to the baseline. We list supplementing object annotations and poses to the SQA3D benchmark as a major future work.

Difference compared to the setting of LEO [22]. While both PoseRecover and LEO can utilize the ground truth object label, we want to point out that they are different settings. PoseRecover aims to rule out the directional ambiguity problem which requires a pose that can only be retrieved using object annotations. PoseAlign only feeds a pose facing the object into the point cloud encoder and the point cloud encoder is not tuned at all, which forbids formation of any direct object feature shortcuts like presented in LEO. We are fully aware that the performance increment of PoseAlign is brought by the normally unavailable pose provided by PoseRecover, but we see this as a testament to our hypothesis that a mission-critical pose derived prior to 3D LMMs is a great tool for ruling out reasoning ambiguity. The system in real action should be composed of two stages, where the agent first 'imagines' itself to be at the mission location and then derives the answers based on that reference frame. Furthermore, the way PoseRecover generates the pose is non-deterministic and the target pose is randomly chosen from all poses that can see the target object, which makes it a lot more realistic and forgiving than

Model	Noise Bound	ScanRefer mIoU↑	Multi3DRef mIoU↑	ScanQA B-4↑	Scan2Cap B-4↑
3D-LLAVA	/	42.6	48.1	16.3	36.4
3D-LLAVA+ PoseAlign-T (Top)	0°, 0m	68.5	60.2	17.3	37.1
	15°, 0.2m	62.5	57.7	17.1	36.9
	30°, 0.5m	53.2	55.1	17.0	36.6

Table 5. Robustness to added uniform pose error.

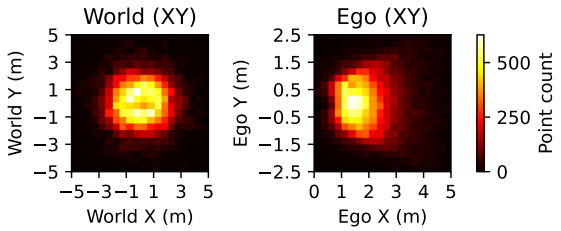

Figure 6. Distribution of objects in world (L) and PoseAlign ego (R) coordinates on X-Y plane. While clustered in front view, objects still follow non-trivial distribution.

LEO setup as shown in Table 1. More explanation can be found in section A.1.

Applicability to RGB-D data without camera extrinsics or LiDAR data. In practical embodied workflows, camera extrinsics are a free lunch from SLAM. For datasets lacking extrinsics, they can be recovered offline using ORB-SLAM2 or similar methods. We simulated SLAM pose noises in Table 5 which reveals the robustness of our method. For single-frame RGB-D or LiDAR data without extrinsics, which are too sparse for SLAM, we operate on single-frame point clouds, which naturally lie in the ego frame and thus constitute a natural application of our method.

Does our setup introduce information leakage? First, we clarify that our benchmark is intentionally defined as an ego-conditioned understanding task, rather than standard free-view understanding. As such, ego alignment is part of the task specification rather than an unintended shortcut. Second, to reduce unintended bias, we avoid selecting the single best-matching viewpoint and instead sample from a diverse candidate set, resulting in non-trivial object spatial distributions in the horizontal plane (see Figure 6). Third, we freeze the point cloud encoder to prevent learning geometric shortcuts, further limiting overfitting to pose priors. Fourth, a PoseAlign-trained 3D-LLAVA model tested on random camera extrinsics results in ScanRefer mIoU dropping from 55.4% to 42.9% but still above vanilla 42.6%. So target facing is crucial to directional awareness but is not a shortcut, otherwise the performance would drop below vanilla.

Interpretation of the performance superiority of PoseAlign-T. The pretrained 3D encoder is trained on cartesian coordinates which internalizes a concept of distance and directions based on the input coordinates. Compared to PoseAlign-T which guarantees consistency at input, PoseAlign-E and PoseAlign-P variants cannot guarantee alignment with (or even fight against) the encoder's concept of direction, causing deterioration.

Design	ScanQA L-A↑	SQA3D L-A↑	Scan2Cap L-A↑
LL3DA	35.2	-	23.3
LL3DA+PoseAlign-T	35.4	-	-
LL3DA-S	36.3	-	-
LL3DA-S+PoseAlign-T	36.5	-	-
Chat-Scene	44.1	56.9	38.6
Chat-Scene+PoseAlign-E	44.7	56.8	39.7
3D-LLAVA	45.9	**58.1**	39.3
3D-LLAVA+PoseAlign-T	**48.3**	57.3	**40.7**

Table 6. Comparison between baselines and respective PoseAlign variants using GPT-5-mini for LLM-as-judge.

A.3. Limitations

While PoseRecover and PoseAlign provide a lightweight and general solution for introducing directional awareness into existing 3D-LMMs, several limitations remain. First, our method relies on the assumption that the 3D LMM is applied on embodied agents, where the availability and accuracy of camera extrinsics are guaranteed. Noise from inaccurate sensor calibration or synchronization is not considered. Second, the recovered poses are limited to the discrete set of views present in ScanNet and may not fully represent all possible viewpoints implied by the language queries, e.g., 'standing on the kitchen counter' or 'crouching under the desk'. Third, although PoseAlign improves directional reasoning without retraining the point cloud encoder, it still assumes the encoder's coordinate sensitivity is preserved, which may not hold for certain architectures employing rotation-invariant designs. Finally, our approach focuses on static indoor environments; extending it to dynamic or outdoor scenarios where ego motion and scene layout change continuously remains an open direction.

B. Additional Experiments

B.1. Other Experimental Setup

Diversity of backbone architectures. We adopt four baseline models, including LL3DA [8], LL3DA-SONATA, Chat-Scene [21] and 3D-LLAVA [14]. Their encoder backbones have distinct architectures ranging from BERT-style Vote2Cap-DETR operating on point patches [7], self-supervised PointTransformerV3 with serialized voxels [54], sparse convolutional neural network with sparse voxels [44], and Omni Superpoint Transformer operating on superpoints [14]. They include both object-wise [8, 21] and

X	ScanQA (val)					SQA3D (test)			Scan2Cap (val)				
	C	B-4	M	R	L-A	EM	EM-R	L-A	C@0.5	B-4@0.5	M@0.5	R@0.5	L-A
None	85.6	15.6	17.8	40.4	43.9	52.8	56.1	56.9	74.0	34.5	26.8	56.4	26.6
0.0	83.8	14.2	17.4	39.7	43.1	53.4	56.2	57.2	37.2	27.9	23.4	52.0	9.76
0.1	86.7	14.9	17.9	40.8	44.3	52.6	55.7	56.6	74.1	34.7	26.9	56.5	27.0
0.2	87.2	15.1	18.1	41.1	44.7	53.3	56.1	57.1	75.5	35.0	27.1	56.7	26.9
0.3	87.5	15.5	17.9	40.9	44.8	53.3	56.2	57.1	74.9	34.7	26.9	56.5	26.9
0.4	82.7	14.3	17.0	39.2	42.7	52.8	55.4	56.1	27.5	25.6	22.4	50.6	6.2
0.45	84.3	14.0	17.5	39.9	42.9	52.8	55.4	56.3	28.2	26.4	22.7	51.1	6.8
0.49	83.6	15.4	17.4	39.5	42.8	52.8	55.5	56.3	60.7	32.2	25.8	54.9	19.0

Table 7. **Full parameter tuning experiment** for Clip Ratio X of PoseAlign-Embed on Chat-Scene.

X	ScanQA (val)					SQA3D (test)			Scan2Cap (val)				
	C	B-4	M	R	L-A	EM	EM-R	L-A	C@0.5	B-4@0.5	M@0.5	R@0.5	L-A
None	95.4	16.3	18.9	44.6	45.7	54.9	57.3	57.9	77.4	36.4	26.9	57.4	28.1
0.0	98.7	16.3	19.3	46.2	47.3	55.1	57.6	56.1	75.5	36.6	26.9	57.2	30.1
0.1	99.4	17.4	19.4	46.1	46.6	52.6	55.0	55.6	76.6	37.0	27.1	57.5	30.8
0.2	99.8	17.3	19.5	46.5	47.3	53.4	56.2	56.6	74.7	36.9	27.0	57.5	31.2
0.3	99.8	17.3	19.7	46.5	47.3	54.2	56.7	57.4	76.1	37.1	27.1	57.6	31.4
0.4	96.9	15.6	19.0	45.6	46.9	53.5	56.1	55.9	73.6	36.4	26.8	57.4	30.9
0.45	97.8	16.0	19.2	45.8	46.5	53.9	56.0	56.6	75.7	36.9	27.0	57.4	31.4
0.49	98.2	16.6	19.3	46.1	46.9	54.2	56.7	55.8	76.3	37.3	27.1	57.5	31.1

Table 8. **Full parameter tuning experiment** for Clip Ratio X of PoseAlign-Transform on 3D-LLAVA.

point-wise [14, 54] feature projection, both with or without a Q-Former [13]. Their base LLMs include OPT-1.3B, Vicuna-v1.5-7B, and the LLM section of LLAVA-v1.5-7B. Consistent performance improvements despite such diversity proves the universal applicability of the PoseAlign modification.

Hardware specification. All comparison methods and respective improved versions are trained on $4\times$ RTX 4090 24G. Batch sizes are trimmed to the same value within the same backbone to enable fair comparison.

PoseAlign-Embed implementation. For the only backbone enhanced with PoseAlign-Embed, Chat-Scene has been a special case among all object-centric 3D LMMs because it passes both the location and the feature of an object into the projection module. In that sense, encoding the ego-pose feature in Chat-Scene reduces to transforming the object locations following Equation 8. For other backbones where the object location is not passed to the LLM, similar modifications can be done by adding a sinusoidal position embedding of the transformed object location onto the projection feature.

LLM-as-judge configuration. For GPT-OSS-20B, we use the officially-recommended vllm package, which packs information into harmony format. The reasoning effort is set to medium in the system prompt. A stronger assessment suite is enabled with GPT-5-mini using minimal reasoning budget.

B.2. Additional Results

PoseRecover time consumption. We down-sample one ego pose every 25 frames for ScanNet-v2, obtaining a reduced yet sufficient set of ego pose trajectories for later processing. Our single-thread program processed 1,513 scenes, each having 1,602 poses and 33 objects on average. Each scene costs 1.7 seconds and total time consumption is 40 minutes on an Intel Xeon CPU. While PoseRecover is already lightweight enough, we provide our PoseRecover results in the code repository for guaranteed reproducibility.

Direction-critical question distribution. We apply the template in section C on the validation sets of three existing point cloud language datasets, ScanQA, Scan2Cap, and SQA3D. The respective portions of direction-critical questions are 46.7% on ScanQA, 89.7% on Scan2Cap, and 95.2% on SQA3D, respectively. The high ratio for the latter two datasets are due to their strong dependence on allocentric spatial relationships to define the caption target or the ego position. We conclude that a significant portion of existing 3D LMM benchmarks contain direction-critical ques-

tions which are ill-posed without ego poses.

Results for LLM-as-judge with GPT-5. In addition to reporting accuracy judged by GPT-OSS-20B, we also provide judgements from the powerful GPT-5 with the same set of prompts, which are listed in Table 6. Similar trends are observed as found in Table 1, where PoseAlign variants improve LLM-as-judge metrics consistently across all baselines and the majority of datasets. Notably, PoseAlign achieves $48.3\%(\Delta 5.0\%)$ and $40.7(+1.4)\%(\Delta 3.6\%)$ accuracy on ScanQA and Scan2Cap, and achieving top-tier accuracy of 57.3% on SQA3D. We also note that the GPT-OSS-20B scores (Table 1) are consistent with the GPT-5 results and both follow the same order, which means that these metrics are consistent and reliable.

Full parameter tuning experiment on Chat-Scene and 3D-LLAVA. We provide all performance metrics of tuning the clip ratio X for two representative backbones, Chat-Scene and 3D-LLAVA, to examine the effect of X on directional alignment. As shown in Tables 7 and 8, performance first improves with moderate clip ratios and then gradually declines beyond $X = 0.3$, exhibiting a clear unimodal trend except on Scan2Cap. Extremely small values ($X<0.1$) cause overfitting to marginal pose deviations, while large values ($X>0.4$) discard too many candidate poses, reducing diversity and robustness. Both models achieve their best balance between stability and discriminability near $X=0.3$, where directional understanding improves across ScanQA, SQA3D, and Scan2Cap without sacrificing general reasoning ability. This consistency across two architectures validates the robustness of the proposed cut-ratio mechanism and supports the choice of $X=0.3$ as the default setting in all main experiments.

C. Prompt Designs

Template for Direction-critical question analysis. The following template is for ScanQA dataset, while the prompts for other datasets are similarly constructed with different examples but the same prompt structure.

```
Your job is to look at a question asking about the
     interior of a room and a bunch of associated ground
      truth answers, and then assign the problem as
     either [A: "NEED_LATERAL_DIRECTION", B: "
     DO_NOT_NEED_LATERAL_DIRECTION"] in order to answer.
First, I will give examples of each class, and then you
     will classify a new question.

The following are examples of
     DO_NOT_NEED_LATERAL_DIRECTION questions.
'''
Question 1: What color is the chair in the kitchen?
Answer 1: "brown", "dark brown".
Question 2: What is the top of the table?
Answer 2: "tv", "television".
Question 3: What is placed next to the fridge?
Answer 3: "door", "the beige door".
Question 4: Where are the two chairs in front of one
     another?
Answer 4: "end table closest to door", "across from each
      other over table"
'''
These questions all DO_NOT_NEED_LATERAL_DIRECTION
     because:
    - Either they ask a different trait of the object
        than its direction (e.g., color, material,
        shape, size, etc.).
    - Or the direction is dependent on an absolute
        direction (e.g., "top", "bottom", "next to", "
        between", "end") which can be implied without
        additional direction information.
    - Or like Question 4 where the reference frame is
        not needed or can be implied despite the
        existence of directional words like "in front
        of", because two chairs facing each other will
        be facing each other no matter where you look
        at them from.
    - While the question may contain spatial words, it
        does not necessarily need the directional
        information to answer.

The following are examples of question that
     NEED_LATERAL_DIRECTION.
'''
Question 1: How many brown chairs are on the left of the
      brown table?
Answer 1: "4", "4"
Question 2: Where is the tall chair on the right of the
     table?
Answer 2: "to left of narrow table tv", "next to 2 other
      chairs as third chair on right"
Question 3: What part of the room has a fridge to its
     right?
Answer 3: "right corner wall adjacent to tv", "corner
     with tv alongside same wall"
Question 4: Where is the beige wooden desk placed?
Answer 4: "up against wall", "at front of class"
Question 5: What is on the front of the brown table?
Answer 5: "tv", "chair"
'''
These predicted answers all NEED_LATERAL_DIRECTION
     because:
    - A relative direction word (e.g., "left", "right",
        "front", "back", "against") is present in the
        question or any of the answers.
    - Even if the question asks about directions
        relative to an object, in some cases the
        reference frame still cannot be implied, e.g.,
        cylindrical or symmetric objects like dishes,
        tables, beds, etc, just like Question 5 above.
    - The question may be asking about a different trait
         of the object than its direction (e.g., color,
         material, shape, size, etc.), but the question
         cannot be answered without the directional
        information.

Also note the following things:
- For classifying the questions where the answers or
     question may seem odd, ignore that and just focus
     on whether the question can be answered without
     directional information.
- The answers may not coincide with each other, but that
      is okay. There can be multiple objects to the left
      or right of something. Just focus on whether the
     question can be answered without directional
     information.

Here is a new question. Simply reply with either
     NEED_LATERAL_DIRECTION or
     DO_NOT_NEED_LATERAL_DIRECTION. Do not ask any more
     questions to the user.
'''
Question: {question}
Answers: {gt_answer}
'''
```

```
Classify the question as one of:
A: NEED_LATERAL_DIRECTION
B: DO_NOT_NEED_LATERAL_DIRECTION

Just return the letters "A" or "B", with no text around
    it.
```

Template for LLM-as-judge. The following template is for LLM-as-judge on the result of ScanQA dataset, while the prompts for other datasets are similarly constructed with different examples but the same prompt structure. The templates can be found in the open-sourced code.

```
Your job is to look at a question, a ground-truth answer
    , and a predicted answer, and then assign a grade
    of either ["CORRECT", "INCORRECT"].
First, I will give examples of each grade, and then you
    will grade a new example.


The following are examples of CORRECT predicted answers.
'''
Question 1: What type of stool is to the left of the
    door?
Ground truth: "bar stool", "black topped wooden stool"
Predicted answer 1: wooden stool.
Predicted answer 2: black topped bar stool.
Predicted answer 3: wooden bar stool.

Question 2: Where is the door of the refrigerator
    located?
Ground truth: "next to microwave", "on refrigerator to
    right of door"
Predicted answer 1: next to the microwave.
Predicted answer 2: on the refrigerator to the right of
    the door.

Question 3: What is in the bin next to the door?
Ground Truth: "trash for recycling", "recycling bins"
Predicted answer 1: trash can.
Predicted answer 2: litter waiting for recycling.
Predicted answer 3: waste.
'''
These predicted answers are all CORRECT because:
    - They fully contain the important information in
        one of the ground truths. Aliases and synonyms
        are acceptable.
    - They do not contain any information that
        contradicts the ground truth.
    - Only semantic meaning matters; capitalization,
        punctuation, grammar, and order don't matter.
    - Hedging and guessing are permissible, provided
        that the ground truth is fully included and the
         response contains no incorrect information or
        contradictions.


The following are examples of INCORRECT predicted
    answers.
'''
Question 1: What is to the right of the refrigerator and
     above the kitchen cabinet?
Ground truth: "microwave", "microwave"
Predicted answer 1: kitchen counter.
Predicted answer 2: kitchen sink.
Predicted answer 3: water tap.

Question 2: Where is the door of the refrigerator
    located?
Ground truth: "next to microwave", "on refrigerator to
    right of door"
Predicted answer 1: on the microwave.
Predicted answer 2: on the door next to the refrigerator
    .
Predicted answer 3: to right of refrigerator.
Predicted answer 4: on the microwave above the
    refrigerator.
Predicted answer 5: on the refrigerator to left of the
    door.

Question 3: How much larger is the blue recycling bin
    compared to the beige trash can?
Ground truth: "twice as wide", "about double size"
Predicted answer 1: larger.
Predicted answer 2: blue recycling bin is larger than
    beige trash can.
Predicted answer 3: blue recycling bin is much larger
    than beige trash can.
'''
These predicted answers are all INCORRECT because:
    - None of the information in the ground truth is
        included in the answer.
    - Or the predicted answer contradicts facts
        contained in the question and the ground truth.
         Pay extra attention to object relations, e.g.,
         the door of the refrigerator cannot be on the
        microwave or another door. Having the correct
        words is not enough; the meaning must be
        correct.
    - The answer shies away from the question and does
        not provide any useful information.


Also note the following things:
- For grading questions where the ground truth contains
    a number, the predicted answer needs to be correct
    to the counting figure in the ground truth, but can
     be relaxed by +-1 when describing rough figures
    like relative size. For example, consider a
    question "How much larger is the brown cabinet
    compared to the purple stool?" with ground truth "5
     times larger".
    - Predicted answers "5", "4", and "6" are all
        CORRECT.
    - Predicted answers "2", "3" and "8" are INCORRECT.
    - Predicted answers "larger" and "more than 2" are
        considered INCORRECT because they provide no
        useful information.
- Do not punish predicted answers if they omit
    information that would be clearly inferred from the
     question.
    - For example, consider the question "What surface
        texture is the wooden cardboard?" and the gold
        target "wooden matte", "brown flat". The
        predicted answer "flat" would be considered
        CORRECT, even though it does not include "
        wooden", since it is clear from the question
        that the cardboard is wooden.


Here is a new example. Simply reply with either CORRECT
    or INCORRECT.
'''
Question: {question}
Ground truth: {ground_truth}
Predicted answer: {predicted_answer}
'''

Grade the predicted answer of this new question as one
    of:
A: CORRECT
B: INCORRECT

Just return the letters "A" or "B", with no text around
    it.
```