SceneScaffold: Active Scene-State Construction for Unified 3D Scene Understanding

Xiangqi Li[1,2] **Libo Huang**[1*] **Jiarui Zhao**[2] **Weilun Feng**[1,2]
Chuanguang Yang[1] **Zhulin An**[1*] **Yongjun Xu**[1]

[1]State Key Laboratory of AI Safety, Institute of Computing Technology, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences
{lixiangqi24s,huanglibo,fengweilun24s,yangchuanguang,anzhulin,xyj}@ict.ac.cn
zhaojiarui231@mails.ucas.ac.cn
Code: https://github.com/lixiangqi707/SceneScaffold

Abstract

Recent 3D large multimodal models (3D-LMMs) rely on a visual bottleneck to compress complex 3D scene evidence into a limited number of visual tokens compatible with large language models (LLMs). Current visual bottlenecks, however, often passively compress heterogeneous 3D evidence into a homogeneous object-centric token sequence, leaving the spatial organization of the scene under-represented. This under-representation forces the LLM to recover spatial relations from a flattened token sequence, leading to unstable reasoning in relation-intensive and spatially ambiguous scenes. To address this issue, we propose **SceneScaffold**, an active scene-state construction framework for unified 3D scene understanding. SceneScaffold reformulates the visual bottleneck from a passive feature compressor into an active scene organizer, constructing a role-aware spatial scaffold before language reasoning. Specifically, SceneScaffold organizes superpoint-level visual evidence into scene-state components with distinct structural roles: entity states preserve core object semantics, scene-frame states maintain spatial references via boundary and region anchors, relation states encode object-environment interaction cues, and a global summary provides compact context. Through this role-aware construction, SceneScaffold provides the LLM with a spatially organized scene representation before language reasoning. Experiments on unified 3D scene understanding tasks, including 3D visual grounding, question answering, and dense captioning, demonstrate the effectiveness of SceneScaffold, while diagnostic results further show its applicability to relation-intensive and spatially ambiguous cases.

1 Introduction

Large Language Models (LLMs) have emerged as the central interface for general semantic reasoning and human-agent interaction [7, 46, 2, 31, 26, 19]. Extending this formidable capability to 3D physical environments is a critical step toward embodied intelligence and real-world spatial interaction [3, 28, 34, 36]. Consequently, recent 3D Large Multimodal Models (3D-LMMs) connect 3D scene representations with natural language within a unified LLM-based architecture [40, 24, 13]. These models are designed to handle diverse scene-level tasks simultaneously, including 3D visual grounding [1, 59], question answering [5, 44], and dense captioning [16, 14]. Since raw 3D scenes contain a massive number of points and candidate regions, a visual bottleneck is indispensable to compress complex 3D representations into a strictly limited number of visual tokens for the LLM [18, 53].

*Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

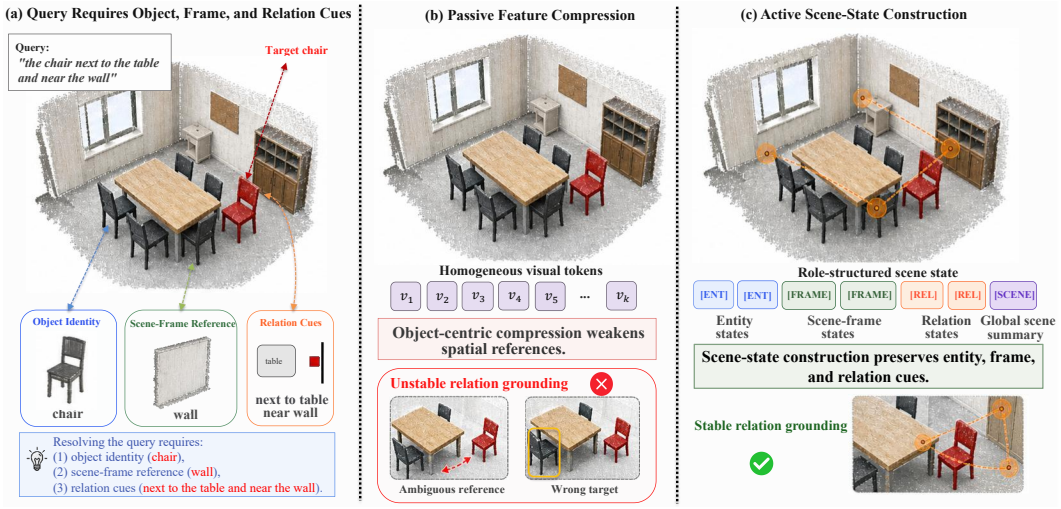


Figure 1: **Motivation of SceneScaffold.** Relation-intensive 3D queries require object identity, scene-frame references, and relation cues. Under a fixed visual bottleneck, passive feature compression tends to preserve object-centric fragments while weakening spatial references, which can lead to unstable relation grounding. In contrast, SceneScaffold organizes visual evidence into a role-structured scene state before language reasoning, explicitly preserving entity evidence, scene-frame references, relation cues, and global scene context.

This bottleneck is especially consequential for 3D scenes, which are continuous physical environments characterized by metric scale, spatial extents, boundaries, and object-environment interactions [17, 9, 15]. As illustrated in Figure 1, resolving a relation-intensive 3D query requires object identity to be grounded together with scene-frame references and relation cues. These spatial references define object locations and their relations to the surroundings, and are essential for reliable 3D understanding [5, 44, 63, 47, 10, 61, 37]. However, current 3D-LMMs typically instantiate the visual bottleneck as a homogeneous feature compressor [13, 18, 53]. Large-scale 3D inputs are selected or pooled into a flat token sequence biased toward semantically salient object-centric evidence [18, 53]. Consequently, critical spatial references, such as scene boundaries, occupied regions, and coarse layout cues, can be under-represented in the limited interface [61, 43, 57]. The object-centric compression process degrades the coherent scene state into a set of isolated object fragments. It also disrupts the collaborative division of labor between the visual and language modalities. When the visual side provides only weakly structured fragments, the language model has to recover scene references, object positions, and spatial relations from a flattened token sequence. This shifted responsibility creates a modal role mismatch: the visual side mainly filters visual evidence, while the language side is forced to perform implicit compensatory reasoning over a one-dimensional sequence to reconstruct missing 3D relations. Such a modal role mismatch is particularly detrimental to unified 3D scene understanding, where multi-task reasoning shares the same physical environment but demands multi-granularity spatial states. For instance, question answering requires global layout comprehension [5, 44], whereas visual grounding requires fine-grained spatial disambiguation among densely distributed, visually similar entities [10, 1, 59, 61]. Forcing the LLM to simultaneously shoulder the dual burden of semantic reasoning and compensatory spatial-structure recovery tends to cause relational confusion, reference drift, and unstable localization as scene complexity increases.

To address these limitations, we argue for a paradigm shift: the visual bottleneck should evolve from a passive feature compressor into an *active scene organizer*. It should explicitly clarify the structural roles of different types of visual evidence and assemble a coarse spatial-reference scaffold before the features enter the LLM. Based on this view, we propose **SceneScaffold**, an active scene-state construction framework for unified 3D scene understanding. Under a fixed cross-modal token budget, SceneScaffold organizes superpoint-level visual evidence into four role-grounded components: *entity states* preserve core object semantics and local details; *scene-frame states* maintain spatial references via boundary and region anchors; *relation states* explicitly encode object-object and object-environment interaction cues; and a *global scene summary* distills compact context for multi-task reuse. Through this role-aware construction and budget-preserving serialization mechanism, the

downstream language model receives a shared scene representation whose basic spatial organization has already been formed on the visual side.

Our main contributions are summarized as follows. First, we analyze the visual bottleneck of unified 3D-LMMs and reveal a modal role mismatch: object-centric compression under-represents spatial references and shifts implicit compensatory spatial reasoning to the language side. Second, we reformulate the visual bottleneck from passive feature compression to active scene-state construction, organizing entity evidence, scene-frame references, and relation cues into a role-structured scene-state interface under a fixed cross-modal token budget. Third, we design **SceneScaffold**, a unified framework that constructs entity states, scene-frame states, relation states, and a global scene summary as a shared scene representation for multiple 3D tasks. Finally, we evaluate SceneScaffold on 3D visual grounding, question answering, and dense captioning, demonstrating its effectiveness for relation-intensive and spatially ambiguous 3D scene understanding.

2 Related Work

2.1 Unified 3D Scene Understanding and 3D LMMs

3D vision-language understanding has evolved from task-specific modeling toward unified scene-level understanding [53, 18, 24, 20]. Early works study individual tasks, including 3D visual grounding in ScanRefer [10] and ReferIt3D [1], dense captioning in Scan2Cap [16], and question answering in ScanQA [5] and SQA3D [44]. Unified models such as 3DJCG [8] and UniT3D [11] jointly handle grounding and captioning, but still rely heavily on task-specific heads. Recent 3D large multimodal models connect 3D scene representations to LLMs for open-ended interaction. 3D-LLM [24] injects lifted 3D features into LLMs, LL3DA [13] introduces visual interactive instruction tuning, PointLLM [58] aligns point clouds with LLMs, Chat-Scene [27] uses object identifiers for scene-level interaction, and Scene-LLM [20] extends language models to 3D reasoning. More recent methods further improve the 3D-LMM interface: 3D-LLaVA [18] integrates visual token selection, prompt encoding, and mask decoding with an Omni Superpoint Transformer; PerLA [45] enhances local-global visual representations; LSceneLLM [62] adaptively selects task-relevant regions; Scenes as Tokens [53] constructs holistic scene tokens with a multi-scale NDT tokenizer. Although these models advance unified 3D scene understanding, most still formulate the cross-modal interface as a finite set of tokens, proposals, or selected features [4]. Their interfaces are typically designed around feature selection, alignment, or tokenization, while the internal role structure of compressed 3D evidence is less explicitly studied. In contrast, we revisit the visual bottleneck itself and argue that unified 3D scene understanding requires a role-preserving shared scene state rather than a homogeneous collection of visual fragments.

2.2 Visual Bottlenecks and Scene Representation

Visual bottlenecks are widely used to reduce the dimensionality of visual inputs before language reasoning [18, 39, 60]. In video LLMs, DynFocus encodes question-relevant frames in greater detail under a limited token budget [21]. DyCPO distinguishes visual exploration from answer-relevant reasoning tokens during optimization [22]. Visual-enhanced depth scaling allocates additional reasoning steps to complex tokens [23]. In 2D multimodal models, LLaVA [42] uses projection layers to align visual features with LLMs, while BLIP-2 [35] adopts query-based connectors. Multimodal Wasserstein barycenter alignment provides a shared representation space across multiple modalities [52]. In all-in-one image restoration, barycenter and residual spaces separate degradation-invariant content from degradation-specific information [51]. Efficient vision transformers further reduce token redundancy through learned token selection in TokenLearner [50], dynamic token sparsification in DynamicViT [49], token reorganization in EViT [38], and token merging in ToMe [6]. Image-restoration transformers also use adaptive sparse attention to suppress irrelevant interactions [66]. Hierarchical attention diversifies feature learning and reduces head redundancy [64], while high-frequency injection helps retain local detail [65]. However, 3D scene evidence is inherently heterogeneous: object surfaces, scene boundaries, layout regions, object-environment references, and object-object relations play different roles. Spatially grounded 3D grounding methods, such as 3DVG-Transformer [61], BUTD-DETR [30], 3D-SPS [43], and EDA [57], model spatial context for grounding, but their relation modeling is usually tied to specific task objectives rather than serving as a reusable interface for LLM-based multi-task understanding.

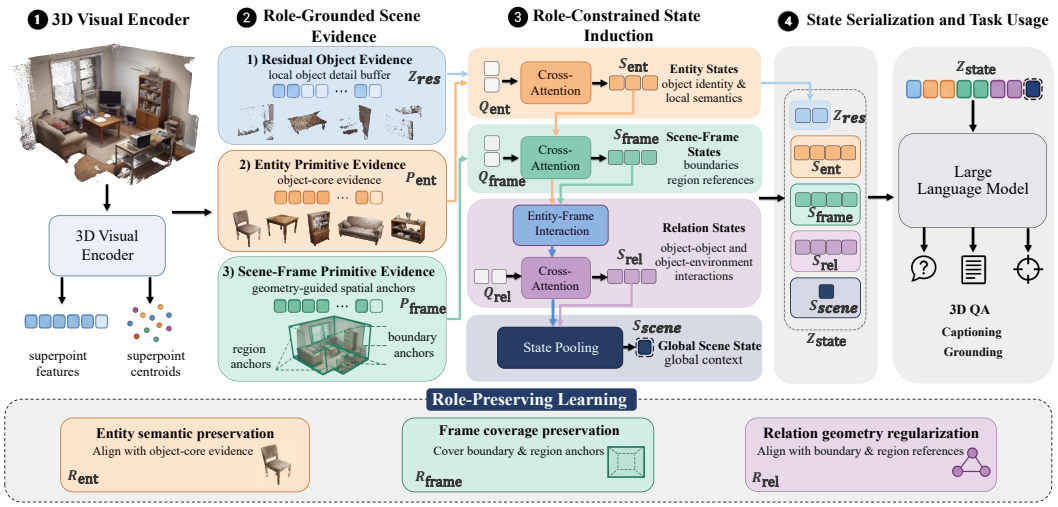


Figure 2: **Overview of SceneScaffold.** Given a 3D scene, the visual encoder extracts superpoint features and centroids. SceneScaffold first organizes them into a residual object buffer, entity primitive evidence, and geometry-guided scene-frame evidence. It then induces entity states, scene-frame states, relation states, and a global scene summary through role-constrained state readers, and serializes the resulting states as a fixed-budget visual interface for the LLM. Role-preserving learning regularizes entity, frame, and relation states with object semantics, spatial coverage, and coarse relation geometry, respectively.

Our work differs from prior token compression and task-specific relation modeling. We do not simply select more salient object tokens under a fixed budget. Instead, we formulate the 3D visual bottleneck as role-preserving scene-state construction, where object-core evidence, scene-frame anchors, and relation-bearing cues are explicitly organized before language reasoning. This provides LLMs with a reusable 3D scene state for visual grounding, question answering, and dense captioning.

3 Method

To address the modal role mismatch caused by homogeneous object-centric compression, we enable the visual side to organize entity evidence, scene-frame references, and relation cues into a compact scene state before language reasoning. We propose SceneScaffold, a role-constrained scene-state construction framework under a fixed visual-token budget. Given visual evidence from a 3D scene, SceneScaffold organizes the visual bottleneck into a multi-task shared scene-state interface. The resulting visual input exposes object evidence, spatial references, and relational cues to the LLM.

As illustrated in Figure 2, the forward pipeline of SceneScaffold consists of four stages: 3D visual encoding, role-grounded scene evidence construction, role-constrained state induction, and state serialization for task usage. Role-preserving learning is applied during training to maintain the structural responsibilities of the induced states.

3.1 Scene-State Interface under a Fixed Visual Budget

Given an input point cloud X, a 3D visual encoder first extracts superpoint-level features and their physical centroids:

$$(H, U) = \mathcal{E}_{\text{3D}}(X), \tag{1}$$

where $H = \{h_j\}_{j=1}^{M}$ denotes M superpoint features, and $U = \{u_j\}_{j=1}^{M}$ denotes their corresponding 3D centroids. Unified 3D scene understanding requires this dense superpoint-level evidence to be transformed into a length-limited visual input for the LLM. SceneScaffold formulates this limited visual interface as a fixed-length sequence with K visual tokens:

$$Z_{\text{state}} = \Pi\left(Z_{\text{res}}, S_{\text{ent}}, S_{\text{frame}}, S_{\text{rel}}, S_{\text{scene}}\right), \quad |Z_{\text{state}}| = K. \tag{2}$$

Here, Z_{res} is a residual object buffer that preserves fine-grained object details from the original visual connector. S_{ent} denotes entity states that encode core object identity and semantics. S_{frame} denotes scene-frame states that maintain spatial references such as boundaries and regions. S_{rel} denotes relation states that model spatial interactions between object evidence and scene-frame references. S_{scene} denotes a global scene summary, and $\Pi(\cdot)$ is the final serialization function.

The fixed budget serves as a constraint on the cross-modal interface. Under this constraint, SceneScaffold grounds different components of the visual interface in distinct evidence sources and induction pathways: residual tokens preserve local object details, entity states read object-core evidence, frame states read geometric spatial anchors, and relation states are induced from entity-frame interactions. This design organizes the limited interface into a role-structured scene-state representation before it is consumed by the LLM.

3.2 Role-Grounded Scene Evidence

SceneScaffold first constructs scene evidence from different sources. The role of each state is grounded at the evidence level: entity states read object-core evidence, scene-frame states read geometric spatial evidence, and the residual object buffer preserves fine-grained object information from the original visual connector.

Entity evidence. For entity states, we use a lightweight entity scoring head to extract object-core evidence from the global superpoint features. Let f_{ent} be the entity scoring function. The entity evidence is defined as

$$\mathcal{P}_{\text{ent}} = \text{TopK}\left(H, f_{\text{ent}}(H), K_{\text{ent}}\right). \tag{3}$$

In our implementation, f_{ent} is a lightweight shared MLP that maps each superpoint feature to a scalar entity score, without language conditioning. Here, K_{ent} denotes the number of selected entity evidence tokens. $\mathcal{P}_{\text{ent}}$ gathers representative evidence for object identity and local semantics in the current scene. It serves as the evidence source used by the entity reader to induce a fixed number of entity states.

Scene-frame evidence. For scene-frame states, we construct geometric spatial evidence from the superpoint centroids U. Specifically, we generate two types of spatial anchors from the horizontal scene extent. The first type consists of anchors pooled from the left, right, front, and back boundary bands. The second type consists of region anchors aggregated from occupied spatial grid cells. Let $\mathcal{A}_{\text{bd}}$ and $\mathcal{A}_{\text{reg}}$ denote the boundary anchors and region anchors, respectively. The scene-frame evidence is written as

$$\mathcal{P}_{\text{frame}} = \mathcal{A}_{\text{bd}} \cup \mathcal{A}_{\text{reg}}. \tag{4}$$

Since this evidence source is determined by geometric locations, it enables scene-frame states to cover spatial references such as boundaries and global occupancy patterns.

In parallel, SceneScaffold keeps a subset of the original visual connector outputs as Z_{res} in the final visual sequence. This residual object buffer preserves local details and helps retain discriminative appearance cues, especially for small objects. Therefore, SceneScaffold forms three complementary inputs at the evidence level: residual object evidence for detail preservation, object-core evidence for entity induction, and scene-frame evidence for spatial organization.

3.3 Role-Constrained State Induction

SceneScaffold applies learnable state readers on top of role-grounded evidence, so that each state is conditioned by the current scene, its evidence source, and its induction path. Given an evidence set $\mathcal{P}$ and a learnable role reader Q, the state induction operator is abstracted as

$$S = \Phi(Q, \mathcal{P}), \tag{5}$$

where $\Phi(\cdot)$ is implemented with a lightweight cross-attention reader, followed by a residual connection and layer normalization. We provide the concrete reader structure in Appendix A.1. The learnable reader Q acts as a role-specific evidence reader, while the output state S is conditioned on the current scene evidence.

Entity states and scene-frame states are induced from their corresponding evidence sources:

$$S_{\text{ent}} = \Phi_{\text{ent}}\left(Q_{\text{ent}}, \mathcal{P}_{\text{ent}}\right), \tag{6}$$

$$S_{\text{frame}} = \Phi_{\text{frame}}\left(Q_{\text{frame}}, \mathcal{P}_{\text{frame}}\right). \tag{7}$$

The former aggregates core object information, while the latter aggregates spatial boundary and region references. Relation states are induced after entity states and scene-frame states have been formed:

$$S_{\text{rel}} = \Phi_{\text{rel}}\left(Q_{\text{rel}}, [S_{\text{ent}}; S_{\text{frame}}]\right). \tag{8}$$

This induction pathway targets the compositional nature of 3D scene relations, where object identity is coupled with scene references such as boundary proximity, inter-object adjacency, and regional localization. By reading $[S_{\text{ent}}; S_{\text{frame}}]$, S_{rel} aggregates interaction cues among objects and between objects and the environment.

Finally, the global scene summary is obtained by aggregating the constructed role-specific states:

$$S_{\text{scene}} = \phi_{\text{scene}}\left([\text{Pool}(S_{\text{ent}}); \text{Pool}(S_{\text{frame}}); \text{Pool}(S_{\text{rel}})]\right). \tag{9}$$

This summary provides compact global context for downstream reasoning. In this way, SceneScaffold constructs role-structured scene states from superpoint-level visual evidence.

3.4 State Serialization and Task Usage

After state construction, SceneScaffold projects the residual object buffer and all role-specific states into the LLM embedding space and serializes them as a unified visual interface:

$$Z_{\text{state}} = [\psi(Z_{\text{res}}) + r_{\text{res}}; \psi(S_{\text{ent}}) + r_{\text{ent}}; \psi(S_{\text{frame}}) + r_{\text{frame}}; \psi(S_{\text{rel}}) + r_{\text{rel}}; \psi(S_{\text{scene}}) + r_{\text{scene}}]. \tag{10}$$

Here, $\psi(\cdot)$ is the cross-modal projector, and r_{res}, r_{ent}, r_{frame}, r_{rel}, and r_{scene} are lightweight role embeddings. The role embeddings expose the identity of each serialized block to the LLM, while the state semantics are grounded by the evidence sources, induction pathways, and role-preserving objectives introduced in Sec. 3.5. We provide serialization details in Appendix A.1.

For unified 3D tasks such as question answering, scene description, and visual grounding, Z_{state} is used as the shared visual interface for the LLM. Different tasks therefore reason over the same scene-state foundation and share object, spatial, and relational evidence.

For tasks that involve dense mask decoding, we keep the standard [SEG]-based query generation and use the constructed states as a lightweight query condition. Let q^0_{seg} be the initial segmentation query. The state-conditioned query is computed as

$$q_{\text{seg}} = q^0_{\text{seg}} + \alpha W_{\text{state}}\left[\text{Pool}(S_{\text{ent}}); \text{Pool}(S_{\text{frame}}); \text{Pool}(S_{\text{rel}})\right], \tag{11}$$

where α is a learnable scaling factor and W_{state} maps the pooled states into the query space. This condition allows the mask decoder to access the same object, spatial, and relational summaries used by the LLM, providing additional context for spatially constrained referring expressions.

3.5 Training Objective

The training objective of SceneScaffold combines task supervision with role-preserving supervision:

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \mathcal{R}_{\text{role}}. \tag{12}$$

Here, $\mathcal{L}_{\text{task}}$ includes the standard language modeling loss and the segmentation mask loss. The role-preserving term maintains the structural responsibilities of different states:

$$\mathcal{R}_{\text{role}} = \lambda_{\text{ent}}\mathcal{R}_{\text{ent}} + \lambda_{\text{frame}}\mathcal{R}_{\text{frame}} + \lambda_{\text{rel}}\mathcal{R}_{\text{rel}}. \tag{13}$$

We keep the role-preserving objectives compact in the main text and provide the detailed construction of the coverage score c_g, the boundary label y_{bd}, and the region label y_{reg} in Appendix A.

Entity semantic preservation. Entity states are encouraged to remain consistent with the object-core evidence they read. We construct an entity target by weighted aggregation over entity evidence and align the pooled entity state with this target in the feature space:

$$\mathcal{R}_{\text{ent}} = 1 - \cos\left(g_{\text{ent}}(\text{Pool}(S_{\text{ent}})), \text{Pool}_w(\mathcal{P}_{\text{ent}})\right). \tag{14}$$

Here, g_{ent} is a lightweight projection layer, and Pool_w denotes score-weighted pooling over entity evidence. This constraint stabilizes the object identity and local semantics preserved by entity states.

Frame coverage preservation. Scene-frame states are encouraged to cover diverse boundary and region anchors. Let A_{frame} be the attention distribution when frame states read frame evidence, and let $\mathcal{G}$ denote the set of boundary and region anchor groups. For each group g, we define its coverage as c_g. The frame coverage term is

$$\mathcal{R}_{\text{frame}} = -\frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \log(c_g + \epsilon). \tag{15}$$

This term encourages frame states to attend to multiple boundary and region groups, so that the frame representation preserves spatial coverage rather than concentrating on a small subset of anchors.

Relation geometry regularization. The relation states model spatial interactions between object evidence and scene-frame references. We apply a coarse geometry-aware regularization that aligns relation states with the boundary and region references derived from entity evidence:

$$\mathcal{R}_{\text{rel}} = \text{CE}(\hat{y}_{\text{bd}}, y_{\text{bd}}) + \text{CE}(\hat{y}_{\text{reg}}, y_{\text{reg}}). \tag{16}$$

Here, y_{bd} and y_{reg} are boundary and region labels generated from the weighted center of entity evidence. This term provides lightweight geometric supervision for relation states, encouraging them to carry coarse object-environment references.

4 Experiments

4.1 Experimental Setting

Datasets and benchmarks. We evaluate SceneScaffold on five representative 3D vision-language benchmarks, including ScanRefer [10], Multi3DRefer [59], ScanQA [5], SQA3D [44], and Scan2Cap [16]. ScanRefer evaluates single-target visual grounding, where a model is required to predict a point-level mask according to a natural-language expression. Multi3DRefer extends this setting to zero-target, single-target, and multi-target expressions, and is therefore more suitable for evaluating multi-object disambiguation in complex 3D scenes. ScanQA evaluates question answering over 3D scenes, while SQA3D further introduces situated question answering under embodied scene contexts. Scan2Cap evaluates dense 3D captioning, where the model describes target objects together with their attributes, locations, and surrounding context. For ScanRefer and Multi3DRefer, we report mean Intersection-over-Union (mIoU). For ScanQA and Scan2Cap, we report CIDEr (C), BLEU-4 (B-4), METEOR (M), and ROUGE-L (R). For SQA3D, we report exact match (EM) and its refined version (EM-R). These benchmarks jointly cover point-level grounding, language-based reasoning, and object description, allowing us to evaluate whether a shared scene-state interface can support heterogeneous 3D scene understanding tasks.

Implementation details. For a fair comparison with recent unified 3D LMMs, we follow the unified instruction-tuning protocol of 3D-LLaVA [18]. SceneScaffold uses LLaVA-1.5-7B [41] as the language-model backbone. The point-cloud encoder adopts a 3D U-Net architecture and is initialized from pretrained weights on large-scale indoor 3D scenes [18]. Given an input point cloud, the point-cloud encoder extracts superpoint-level visual features together with their 3D centroids. SceneScaffold then reorganizes the fixed visual interface into residual detail tokens, entity states, scene-frame states, relation states, and a global scene state. Unless otherwise specified, the total number of visual tokens inserted into the language model is fixed to $K = 100$. The default allocation consists of 75 residual detail tokens, 8 entity states, 8 scene-frame states, 8 relation states, and 1 global scene state. During instruction tuning, we freeze the point-cloud encoder and the main body of the LLM, and optimize only the SceneScaffold state-construction modules, cross-modal projection layers, segmentation-query projection layers, and LoRA parameters [25]. We train the model on 8 NVIDIA A800 GPUs using AdamW with a cosine learning-rate schedule. The initial learning rate is set to 2×10^{-4}. The batch size, number of epochs, and gradient accumulation steps are set to 2, 1, and 8, respectively. The training objective includes language modeling loss, mask prediction loss, and role-preserving regularization, including entity semantic preservation, frame coverage preservation, and relation geometry regularization. Source code is included in the supplementary material and will be made publicly available upon publication.

Table 1: **Overall comparison on five 3D vision-language benchmarks.** *PC* and *I* denote point cloud and image modality (Mod.), respectively. Specialist models and PC+I are included for reference. The best and second-best results among point-cloud-only generalist models are marked in **bold** and underline, respectively.

Method	Mod.	ScanRefer	Multi3D	ScanQA (val)				SQA3D (test)		Scan2Cap (val)			
		mIoU↑	mIoU↑	C↑	B-4↑	M↑	R↑	EM↑	EM-R↑	C@.5↑	B-4@.5↑	M@.5↑	R@.5↑
Specialist Models													
ScanQA [5]	PC	–	–	64.9	10.1	13.1	33.3	46.6	–	–	–	–	–
3D-VLP [33]	PC	–	–	67.0	11.2	13.5	34.5	–	–	54.9	32.3	24.8	51.5
3D-VisTA [67]	PC	–	–	69.6	10.4	13.9	45.7	48.5	–	61.6	34.1	26.8	55.0
Scan2Cap [16]	PC	–	–	–	–	–	–	41.0	–	39.1	23.3	22.0	44.8
MORE [32]	PC	–	–	–	–	–	–	–	–	40.9	22.9	21.7	44.4
SpaCap3D [54]	PC	–	–	–	–	–	–	–	–	44.0	25.3	22.3	45.4
D3Net [12]	PC	–	–	–	–	–	–	–	–	46.1	30.3	24.4	51.7
UniT3D [11]	PC	–	–	–	–	–	–	–	–	46.7	27.2	21.9	46.0
3DJCG [8]	PC	–	–	–	–	–	–	–	–	49.5	31.0	24.2	50.8
Vote2Cap-DETR [14]	PC	–	–	–	–	–	–	–	–	61.8	34.5	26.2	54.4
TGNN [29]	PC	27.8	–	–	–	–	–	–	–	–	–	–	–
M3DRef-CLIP [59]	PC	35.7	32.6	–	–	–	–	–	–	–	–	–	–
X-RefSeg3D [48]	PC	29.9	–	–	–	–	–	–	–	–	–	–	–
3D-STMN [56]	PC	39.5	–	–	–	–	–	–	–	–	–	–	–
Generalist Models													
3D-LLM [24]	PC+I	–	–	69.4	12.0	14.5	35.7	–	–	–	–	–	–
LEO [28]	PC+I	–	–	101.4	13.2	20.0	49.2	50.0	52.4	72.4	38.2	27.9	58.1
Scene-LLM [20]	PC+I	–	–	80.0	11.7	15.8	35.9	53.6	–	–	–	–	–
Chat-Scene [27]	PC+I	–	–	87.7	14.3	18.0	41.6	54.6	57.5	77.2	36.4	28.0	58.1
LL3DA [13]	PC	–	–	76.8	13.5	15.9	37.3	–	–	65.2	36.8	26.0	55.1
Spatial 3D-LLM [55]	PC	–	–	82.5	13.9	16.8	39.1	46.2	–	72.2	34.6	23.1	54.3
Grounded 3D-LLM [15]	PC	–	–	72.7	13.4	–	–	–	–	70.6	35.5	–	–
LSceneLLM [62]	PC	–	–	88.2	–	18.0	40.8	–	–	–	–	–	–
3D-LLaVA [18]	PC	43.3	42.7	92.6	**17.1**	18.4	43.1	54.5	56.6	78.8	36.9	27.1	57.7
NDTokenizer3D [53]	PC	–	46.0	**98.6**	17.0	**19.4**	44.9	54.4	57.1	79.0	36.7	27.1	57.7
SceneScaffold (ours)	PC	**47.0**	**47.9**	95.8	16.8	19.0	**45.0**	**55.5**	**58.3**	**79.1**	**37.1**	**27.3**	**57.8**

4.2 Main Results

Table 1 compares SceneScaffold with specialist models and generalist 3D models on five 3D vision-language benchmarks. SceneScaffold uses point clouds as visual input and constructs a role-factored scene-state interface under the same fixed visual-token budget. Specialist models and PC+I methods are included for reference, while the best and second-best results are highlighted among point-cloud-only generalist models.

3D visual grounding. On 3D visual grounding, SceneScaffold achieves 47.0 mIoU on ScanRefer and 47.9 mIoU on Multi3DRefer, improving over 3D-LLaVA by 3.7 and 5.2 mIoU, respectively. It also outperforms NDTokenizer3D on Multi3DRefer by 1.9 mIoU. These results show that SceneScaffold brings clear gains on localization-oriented tasks, where the model must resolve object identity, spatial references, and multiple similar candidates. This improvement is consistent with the design of SceneScaffold, which organizes the fixed visual interface into entity, frame, and relation states rather than a homogeneous set of object-centric visual tokens.

3D question answering. On ScanQA, SceneScaffold obtains 95.8 CIDEr, 16.8 BLEU-4, 19.0 METEOR, and 45.0 ROUGE-L. Compared with 3D-LLaVA, it improves CIDEr, METEOR, and ROUGE-L by 3.2, 0.6, and 1.9, respectively. On SQA3D, SceneScaffold achieves 55.5 EM and 58.3 EM-R, outperforming both 3D-LLaVA and NDTokenizer3D. These results indicate that the constructed scene states also provide useful context for language-based reasoning over 3D scenes.

3D dense captioning. On Scan2Cap, SceneScaffold achieves 79.1 CIDEr@0.5, 37.1 BLEU-4@0.5, 27.3 METEOR@0.5, and 57.8 ROUGE-L@0.5. Compared with 3D-LLaVA, it obtains modest improvements across all four metrics. Overall, SceneScaffold shows the strongest gains on

Table 2: **Diagnostic evaluation on relation-heavy and spatially ambiguous samples.** "Overall" denotes the full evaluation split, while "Rel.+Amb." denotes the subset requiring both spatial-reference understanding and multi-instance disambiguation.

Method	ScanRefer		Multi3DRefer		ScanQA		Scan2Cap	
	Overall mIoU↑	Rel.+Amb. mIoU↑	Overall mIoU↑	Rel.+Amb. mIoU↑	Overall C↑	Rel.+Amb. C↑	Overall C@0.5↑	Rel.+Amb. C@0.5↑
3D-LLaVA	43.3	37.6	42.7	40.7	92.6	84.9	78.8	76.0
SceneScaffold	**47.0**	**40.6**	**47.9**	**43.5**	**95.8**	**88.6**	**79.1**	**77.9**

Table 3: **Ablation on scene-state components.** All variants keep the same final visual-token budget. Removed state slots are reallocated to residual detail tokens.

Variant	Grounding		ScanQA				SQA3D		Scan2Cap			
	ScanRefer mIoU↑	Multi3DRefer mIoU↑	C↑	B-4↑	M↑	R↑	EM↑	EM-R↑	C@.5↑	B-4@.5↑	M@.5↑	R@.5↑
w/o S_{ent}	45.0	46.2	91.1	14.3	18.2	43.6	52.7	56.2	77.5	36.5	27.0	57.6
w/o S_{frame}	45.9	46.2	95.3	16.7	18.9	44.8	55.3	58.3	77.9	36.9	27.1	57.8
w/o S_{rel}	45.1	46.4	92.6	15.1	18.4	43.8	56.4	57.8	75.5	35.9	26.9	57.3
w/o S_{scene}	45.8	46.3	93.0	15.1	18.3	44.2	54.8	57.8	76.1	35.5	27.2	57.7
SceneScaffold Full	47.0	47.9	95.8	16.8	19.0	45.0	55.5	58.3	79.1	37.1	27.3	57.8

grounding-oriented tasks, while maintaining competitive performance on question answering and dense captioning.

4.3 Diagnostic Evaluation on Relation-Heavy and Ambiguous Scenes

To examine whether SceneScaffold benefits samples that require spatial references and multi-instance disambiguation, we evaluate it on a diagnostic Rel.+Amb. subset. This subset contains samples involving both explicit spatial-reference expressions and multiple same-category candidates; its construction details and statistics are provided in Appendix B.1.

As shown in Table 2, the Rel.+Amb. subset is consistently harder than the full split, especially on ScanRefer, ScanQA, and Scan2Cap. SceneScaffold still improves over 3D-LLaVA on all diagnostic metrics, with gains of 3.0 mIoU on ScanRefer, 2.8 mIoU on Multi3DRefer, 3.7 CIDEr on ScanQA, and 1.9 CIDEr@0.5 on Scan2Cap. These results indicate that the proposed scene-state interface remains effective when spatial references and multiple same-category candidates must be jointly resolved, supporting our motivation of organizing entity, frame, and relation evidence before language reasoning.

4.4 Ablation Study

We ablate different scene-state components to examine their contributions to unified 3D scene understanding. All variants keep the same final visual-token budget. When one state component is removed, its slots are reallocated to residual detail tokens, so that the comparison focuses on the organization of the visual interface rather than the number of tokens.

Table 3 reports the ablation results of different scene-state components under the same final visual-token budget. The full SceneScaffold model achieves the best grounding performance, reaching 47.0 mIoU on ScanRefer and 47.9 mIoU on Multi3DRefer. It also obtains the strongest ScanQA results and the most balanced Scan2Cap performance among the ablated variants, showing that the complete state composition provides a more stable visual interface across tasks. Different components contribute complementary roles. Entity states provide broad object-centric evidence: without S_{ent}, ScanRefer drops from 47.0 to 45.0, Multi3DRefer from 47.9 to 46.2, and ScanQA CIDEr from 95.8 to 91.1. Scene-frame states mainly support spatial anchoring. Removing S_{frame} keeps several language metrics close to the full model, but consistently weakens grounding performance, which is consistent with its role in preserving boundary and region references. Relation states appear to support relational reasoning and spatially grounded description: removing S_{rel} reduces ScanQA and Scan2Cap performance, with Scan2Cap CIDEr decreasing from 79.1 to 75.5. The global scene

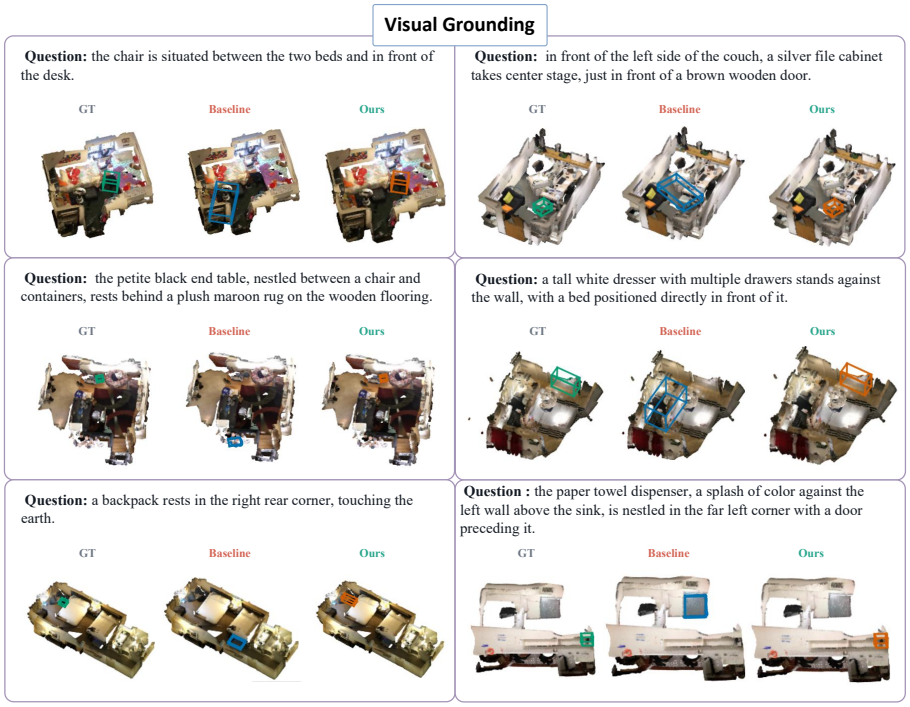


Figure 3: **Qualitative results on visual grounding.** SceneScaffold more accurately grounds relation-heavy and spatially ambiguous expressions by organizing entity, frame, and relation states, whereas the baseline tends to select semantically similar but spatially incorrect instances.

state mainly affects language-oriented tasks; without S_{scene}, ScanQA CIDEr decreases to 93.0 and Scan2Cap CIDEr to 76.1, reflecting its function as a compact summary of entity, frame, and relation states.

Overall, replacing any state component with additional residual detail tokens cannot recover the full model, indicating that SceneScaffold benefits from role-factored scene-state organization rather than merely from local detail preservation. Figure 3 further illustrates this behavior on relation-heavy and spatially ambiguous samples, complementing the diagnostic results in Table 2: the selected cases contain multiple same-category candidates and explicit spatial references, such as objects located between beds, near a wall, in front of another object, or in a room corner. The baseline often identifies the correct object category but drifts to a spatially inconsistent instance, whereas SceneScaffold more reliably grounds the target satisfying the relation constraint. Additional qualitative results on question answering and dense captioning are provided in Appendix B.5. Appendix B.6 additionally shows that removing role-preserving objectives weakens performance while keeping the architecture and token budget unchanged, supporting their role in stabilizing the intended state functions.

5 Conclusion

In this paper, we revisit the visual bottleneck in unified 3D large multimodal models from the perspective of modal role allocation. Instead of treating the bottleneck as a passive feature compressor that selects a homogeneous set of visual fragments, we argue that the visual side should organize a compact 3D scene state before language reasoning. To this end, we propose SceneScaffold, an active scene-state construction framework that reallocates a fixed visual-token budget into residual detail tokens, entity states, scene-frame states, relation states, and a global scene summary. These states are grounded in distinct evidence sources and induction pathways, allowing object semantics, spatial references, and interaction cues to be exposed to the language model in a structured form. Experiments across grounding, question answering, and dense captioning show that SceneScaffold improves unified 3D scene understanding, with larger gains on relation-heavy and spatially ambiguous samples. These results suggest that active scene-state construction is an effective direction for building more reliable 3D-LMM interfaces under limited visual budgets.

Acknowledgments and Disclosure of Funding

This work was supported by Beijing Natural Science Foundation (No. QG26011), National Natural Science Foundation of China (No. 62606510, No. 62476264, No. 62406312, and No. 62676389), and the Youth Key Project of the Chinese Academy of Sciences (GFQN-2026-34).

References

[1] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas J. Guibas. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *European Conference on Computer Vision*, pages 422–440, 2020.

[2] Zhaochong An, Guolei Sun, Yun Liu, Runjia Li, Min Wu, Ming-Ming Cheng, Ender Konukoglu, and Serge Belongie. Multimodality helps few-shot 3D point cloud semantic segmentation. In *International Conference on Learning Representations*, 2025.

[3] Zhulin An, Xinqiang Yu, Chu Wang, Yinlong Zhang, and Chunhe Song. Embodied intelligence: Recent advances and future perspectives. *The Innovation Informatics*, 1(1):100008, 2025.

[4] Zijia An, Boyu Diao, Ruiqi Liu, Libo Huang, Chuanguang Yang, Fei Wang, Zhulin An, and Yongjun Xu. Parameterized prompt for incremental object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

[5] 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] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. In *International Conference on Learning Representations*, 2023.

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

[8] 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.

[9] Angel X. 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. In *International Conference on 3D Vision*, pages 667–676, 2017.

[10] Dave Zhenyu Chen, Angel X. Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *European Conference on Computer Vision*, pages 202–221, 2020.

[11] Dave 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.

[12] Dave Zhenyu Chen, Qirui Wu, Matthias Nießner, and Angel X. Chang. D3net: A unified speaker-listener architecture for 3d dense captioning and visual grounding. In *European Conference on Computer Vision*, pages 487–505, 2022.

[13] 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.

[14] 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.

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

[16] 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.

[17] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5828–5839, 2017.

[18] 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*, pages 3772–3782, 2025.

[19] Weilun Feng, Chuanguang Yang, Haotong Qin, Yuqi Li, Xiangqi Li, Zhulin An, Libo Huang, Boyu Diao, Fuzhen Zhuang, Michele Magno, Yongjun Xu, Yingli Tian, and Tingwen Huang. MPQ-DMv2: Flexible residual mixed precision quantization for low-bit diffusion models with temporal distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

[20] Rao Fu, Jingyu Liu, Xilun Chen, Yixin Nie, and Wenhan Xiong. Scene-LLM: Extending language model for 3D visual reasoning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2195–2206, 2025.

[21] Yudong Han, Qingpei Guo, Liyuan Pan, Liu Liu, Yu Guan, and Ming Yang. DynFocus: Dynamic cooperative network empowers LLMs with video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8512–8522, 2025.

[22] Yudong Han, Yong Wang, Zaiquan Yang, Liang Lin, Chongyang Tao, Xiangxiang Chu, and Liyuan Pan. Beyond entropy: Self-diagnostic multi-role token optimization for video reasoning, 2026.

[23] Yudong Han, Yong Wang, Zaiquan Yang, Zhen Qu, Liyuan Pan, and Xiangxiang Chu. Visual enhanced depth scaling for multimodal latent reasoning. In *Advances in Neural Information Processing Systems*, 2026. Accepted at NeurIPS 2026.

[24] 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. In *Advances in Neural Information Processing Systems*, volume 36, pages 20482–20494, 2023.

[25] 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 *International Conference on Learning Representations*, 2022.

[26] Chengsong Huang, Wenhao Yu, Xiaoyang Wang, Hongming Zhang, Zongxia Li, Ruosen Li, Jiaxin Huang, Haitao Mi, and Dong Yu. R-Zero: Self-evolving reasoning LLM from zero data. In *International Conference on Learning Representations*, 2026.

[27] Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, and Zhou Zhao. Chat-scene: Bridging 3d scene and large language models with object identifiers. In *Advances in Neural Information Processing Systems*, volume 37, pages 113991–114017, 2024.

[28] 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. In *Proceedings of the International Conference on Machine Learning*, pages 20413–20451, 2024.

[29] 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*, volume 35, pages 1610–1618, 2021.

[30] Ayush Jain, Nikolaos Gkanatsios, Ishita Mediratta, and Katerina Fragkiadaki. Bottom up top down detection transformers for language grounding in images and point clouds. In *European Conference on Computer Vision*, pages 417–433, 2022.

[31] Gangwei Jiang, Caigao Jiang, Zhaoyi Li, Siqiao Xue, Jun Zhou, Linqi Song, Defu Lian, and Ying Wei. Unlocking the power of function vectors for characterizing and mitigating catastrophic forgetting in continual instruction tuning. In *International Conference on Learning Representations*, 2025.

[32] 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 *European Conference on Computer Vision*, pages 528–545, 2022.

[33] 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.

[34] Kento Kawaharazuka, Tatsuya Matsushima, Andrew Gambardella, Jiaxian Guo, Chris Paxton, and Andy Zeng. Real-world robot applications of foundation models: A review. *Advanced Robotics*, 38(18):1232–1254, 2024.

[35] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. 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, 2023.

[36] Xiangqi Li, Libo Huang, Zhulin An, Weilun Feng, Chuanguang Yang, Boyu Diao, Fei Wang, and Yongjun Xu. Geometric feature embedding for effective 3d few-shot class incremental learning. In *International Conference on Machine Learning*, 2025.

[37] Xiangqi Li, Libo Huang, Jiarui Zhao, Weilun Feng, Chuanguang Yang, Zhulin An, and Yongjun Xu. Point-uq: An uncertainty-quantification paradigm for point cloud few-shot class incremental learning. In *International Conference on Learning Representations*, 2026.

[38] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Evit: Expediting vision transformers via token reorganizations. In *International Conference on Learning Representations*, 2022.

[39] Junyan Lin, Haoran Chen, Yue Fan, Yingqi Fan, Xin Jin, Hui Su, Jinlan Fu, and Xiaoyu Shen. Multi-layer visual feature fusion in multimodal llms: Methods, analysis, and best practices. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 4156–4166, 2025.

[40] Dingning Liu, Xiaoshui Huang, Yuenan Hou, Zhihui Wang, Zhenfei Yin, Yongshun Gong, Peng Gao, and Wanli Ouyang. Uni3d-llm: Unifying point cloud perception, generation and editing with large language models. *arXiv preprint arXiv:2402.03327*, 2024.

[41] 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.

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

[43] Junyu Luo, Jiahui Fu, Xianghao Kong, Chen Gao, Haibing Ren, Hao Shen, Huaxia Xia, and Si Liu. 3d-sps: Single-stage 3d visual grounding via referred point progressive selection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16454–16463, 2022.

[44] Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. Sqa3d: Situated question answering in 3d scenes. In *International Conference on Learning Representations*, 2023.

[45] Guofeng Mei, Wei Lin, Luigi Riz, Yujiao Wu, Fabio Poiesi, and Yiming Wang. Perla: Perceptive 3d language assistant. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14369–14379, 2025.

[46] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[47] Zhangyang Qi, Ye Fang, Zeyi Sun, Xiaoyang Wu, Tong Wu, Jiaqi Wang, Dahua Lin, and Hengshuang Zhao. Gpt4point: A unified framework for point-language understanding and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26417–26427, 2024.

[48] 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*, volume 38, pages 4551–4559, 2024.

[49] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. In *Advances in Neural Information Processing Systems*, volume 34, pages 13937–13949, 2021.

[50] Michael S. Ryoo, A. J. Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. Token-Learner: Adaptive space-time tokenization for videos. In *Advances in Neural Information Processing Systems*, volume 34, pages 12786–12797, 2021.

[51] Xiaole Tang, Xiaoyi He, Jiayi Xu, Xiang Gu, and Jian Sun. Learning continuous wasserstein barycenter space for generalized all-in-one image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(7):8395–8410, 2026.

[52] Xiaole Tang, Jiayi Xu, Xiang Gu, Yan Yang, and Jian Sun. Binding multiple modalities via multimodal wasserstein barycenter, 2026.

[53] Yutao Tang, Cheng Zhao, Gaurav Mittal, Rohith Kukkala, Rama Chellappa, Cheng Peng, and Mei Chen. Scenes as tokens: Multi-scale normal distributions transform tokenizer for general 3D vision-language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 38584–38594, 2026.

[54] 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*, pages 1393–1400, 2022.

[55] Xiaoyan Wang, Zeju Li, Yifan Xu, Jiaxing Qi, Zhifei Yang, Ruifei Ma, Xiangde Liu, and Chao Zhang. Spatial 3d-llm: exploring spatial awareness in 3d vision-language models. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2025.

[56] 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*, volume 38, pages 5940–5948, 2024.

[57] Yanmin Wu, Xinhua Cheng, Renrui Zhang, Zesen Cheng, and Jian Zhang. Eda: Explicit text-decoupling and dense alignment for 3d visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19231–19242, 2023.

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

[59] 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.

[60] Jiarui Zhao, Libo Huang, Xiangqi Li, Zhulin An, Chuanguang Yang, Yu Wang, Boyu Diao, and Yongjun Xu. Representation-steered incremental adapter-tuning for class-incremental learning with pre-trained models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18010–18020, 2026.

[61] Lichen Zhao, Daigang Cai, Lu Sheng, and Dong Xu. 3dvg-transformer: Relation modeling for visual grounding on point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2928–2937, 2021.

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

[63] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3D: Exploring unified 3D representation at scale. In *International Conference on Learning Representations*, 2024.

[64] Shihao Zhou, Dayu Li, Jinshan Pan, Juncheng Zhou, Jinglei Shi, and Jufeng Yang. Devil is in the uniformity: Exploring diverse learners within transformer for image restoration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12307–12317, 2025.

[65] Shihao Zhou, Jinshan Pan, Duosheng Chen, Yaopeng Dong, and Jufeng Yang. Rethinking the importance of high-frequency components in transformers for image restoration. *IEEE Transactions on Image Processing*, 35:4744–4758, 2026.

[66] Shihao Zhou, Jinshan Pan, and Jufeng Yang. Learning an adaptive sparse transformer for efficient image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(11):10344–10360, 2025.

[67] 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.

Appendix for *SceneScaffold: Active Scene-State Construction for Unified 3D Scene Understanding*

A Implementation Details of Scene-State Construction and Learning

This section provides implementation details for state reader construction, serialization, role-grounded evidence construction, and role-preserving learning in SceneScaffold. In the main paper, we present compact forms of entity semantic preservation, frame coverage preservation, and relation geometry regularization. We also summarize the default hyperparameter settings used in all main experiments. Here, we further specify the construction of the weighted entity target, frame group coverage c_g, boundary label y_{bd}, and region label y_{reg}.

A.1 State Reader, Serialization, and Hyperparameter Settings

We instantiate each state induction operator Φ as a one-layer cross-attention reader with residual connection and layer normalization. Given learnable role queries Q and evidence tokens P, the reader is implemented as

$$\Phi(Q, P) = \mathrm{LN}\left(Q + \mathrm{MHA}(Q, P, P)\right), \tag{17}$$

where $\mathrm{MHA}(\cdot)$ denotes multi-head cross-attention. Entity, frame, and relation states use independent query parameters and independent reader blocks. The reader uses four attention heads, attention dropout 0.0, and temperature 1.0.

All serialized visual tokens are mapped by the same cross-modal projector $\psi(\cdot)$. The serialized order is fixed for all scenes:

$$[Z_{\mathrm{res}}, S_{\mathrm{ent}}, S_{\mathrm{frame}}, S_{\mathrm{rel}}, S_{\mathrm{scene}}]. \tag{18}$$

After projection, we add one learnable role embedding to each block type, corresponding to residual, entity, frame, relation, and scene-summary tokens. These role embeddings are initialized to zero and learned during training. We do not introduce an additional role-specific projector; all state blocks share the same visual-to-language projector before role embeddings are added.

In our implementation, the total visual budget is $K = 100$. We use 8 entity state slots, 8 scene-frame state slots, 8 relation state slots, and one global scene summary token. Unless otherwise specified, the residual object buffer uses the remaining budget, resulting in 75 residual object tokens. For role-grounded evidence construction, the entity primitive budget is 128. The scene-frame evidence uses four boundary anchors and a 3×3 region grid, with a boundary-band ratio of 0.15 and a minimum of four superpoints required for a valid anchor. The frame evidence budget is therefore determined by the valid boundary and occupied region anchors, up to the configured frame-anchor budget. The state-conditioned segmentation query uses a learnable scale initialized to 0.0. The role-preserving loss weights are set to 0.05 for entity semantic preservation, frame coverage preservation, relation boundary regularization, and relation region regularization. The residual object dropout rate is set to 0.2 during training and disabled during inference.

A.2 Entity Target for Semantic Preservation

Given superpoint features $H = \{h_j\}_{j=1}^{M}$, the entity scoring head predicts an entity score for each superpoint:

$$a_j = f_{\mathrm{ent}}(h_j). \tag{19}$$

The scoring head f_{ent} is implemented as a lightweight MLP:

$$f_{\mathrm{ent}}(h_j) = W_2 \sigma\left(W_1 \mathrm{LN}(h_j)\right), \tag{20}$$

where $\mathrm{LN}(\cdot)$ is layer normalization and $\sigma(\cdot)$ is GELU. The output is a scalar entity score for each superpoint. We select the top-scored superpoints as entity evidence:

$$\mathcal{I}_{\mathrm{ent}} = \mathrm{TopK}(a, K_{\mathrm{ent}}), \quad \mathcal{P}_{\mathrm{ent}} = \{h_j \mid j \in \mathcal{I}_{\mathrm{ent}}\}. \tag{21}$$

We use hard TopK selection in this step. Gradients are propagated through the selected feature paths, while the discrete index selection itself is not relaxed. The selected entity scores are normalized as

$$w_j = \frac{\exp(a_j)}{\sum_{k \in \mathcal{I}_{\mathrm{ent}}} \exp(a_k)}, \quad j \in \mathcal{I}_{\mathrm{ent}}. \tag{22}$$

The score-weighted entity target is

$$t_{\text{ent}} = \text{Pool}_w(\mathcal{P}_{\text{ent}}) = \sum_{j \in \mathcal{I}_{\text{ent}}} w_j h_j. \tag{23}$$

The pooled entity state is

$$\bar{S}_{\text{ent}} = \text{Pool}(S_{\text{ent}}). \tag{24}$$

The entity semantic preservation term used in the main paper is implemented as

$$\mathcal{R}_{\text{ent}} = 1 - \cos\left(g_{\text{ent}}(\bar{S}_{\text{ent}}), t_{\text{ent}}\right), \tag{25}$$

where g_{ent} is a lightweight projection layer. This term aligns entity states with the object-core target aggregated from entity evidence.

A.3 Construction of Frame Anchors

Scene-frame evidence consists of boundary anchors and region anchors. For each superpoint centroid

$$u_j = (x_j, y_j, z_j), \tag{26}$$

we compute the horizontal scene extent:

$$x_{\min}, x_{\max}, y_{\min}, y_{\max}. \tag{27}$$

Boundary anchors. Given a boundary-band ratio ρ, we define four horizontal boundary bands:

$$\mathcal{B}_{\text{left}} = \{j \mid x_j \le x_{\min} + \rho(x_{\max} - x_{\min})\}, \tag{28}$$

$$\mathcal{B}_{\text{right}} = \{j \mid x_j \ge x_{\max} - \rho(x_{\max} - x_{\min})\}, \tag{29}$$

$$\mathcal{B}_{\text{front}} = \{j \mid y_j \le y_{\min} + \rho(y_{\max} - y_{\min})\}, \tag{30}$$

$$\mathcal{B}_{\text{back}} = \{j \mid y_j \ge y_{\max} - \rho(y_{\max} - y_{\min})\}. \tag{31}$$

For each non-empty boundary band $q \in \{\text{left}, \text{right}, \text{front}, \text{back}\}$, its boundary anchor is obtained by average pooling:

$$a_q^{\text{bd}} = \frac{1}{|\mathcal{B}_q|} \sum_{j \in \mathcal{B}_q} h_j. \tag{32}$$

All boundary anchors form

$$\mathcal{A}_{\text{bd}} = \{a_q^{\text{bd}}\}. \tag{33}$$

Region anchors. For region anchors, we divide the horizontal plane into a $G_x \times G_y$ grid. We first normalize the horizontal coordinates:

$$\tilde{x}_j = \frac{x_j - x_{\min}}{x_{\max} - x_{\min} + \epsilon}, \quad \tilde{y}_j = \frac{y_j - y_{\min}}{y_{\max} - y_{\min} + \epsilon}. \tag{34}$$

The region index of each superpoint is

$$r_j = \lfloor G_x \tilde{x}_j \rfloor + G_x \lfloor G_y \tilde{y}_j \rfloor. \tag{35}$$

For each occupied region cell r, we define

$$\mathcal{B}_r^{\text{reg}} = \{j \mid r_j = r\}. \tag{36}$$

The corresponding region anchor is

$$a_r^{\text{reg}} = \frac{1}{|\mathcal{B}_r^{\text{reg}}|} \sum_{j \in \mathcal{B}_r^{\text{reg}}} h_j. \tag{37}$$

All region anchors form

$$\mathcal{A}_{\text{reg}} = \{a_r^{\text{reg}}\}. \tag{38}$$

The final scene-frame evidence is

$$\mathcal{P}_{\text{frame}} = \mathcal{A}_{\text{bd}} \cup \mathcal{A}_{\text{reg}}. \tag{39}$$

In implementation, boundary anchors are constructed first, and the remaining frame-evidence budget is assigned to occupied region anchors. When the number of occupied region cells exceeds the remaining budget, we keep the cells with larger superpoint counts. With the default setting, we use four boundary anchors and a 3×3 region grid; only anchors satisfying the minimum-point requirement are used for state induction.

A.4 Frame Coverage Term

The frame coverage term encourages S_{frame} to cover diverse boundary and region anchors. Let

$$A_{\text{frame}} \in \mathbb{R}^{M_{\text{frame}} \times |\mathcal{P}_{\text{frame}}|} \tag{40}$$

be the cross-attention map produced when frame states read frame evidence. $A^{m,j}_{\text{frame}}$ denotes the attention weight from the m-th frame state slot to the j-th frame primitive.

Each frame primitive is assigned a group id. Boundary anchors are grouped by their directions, i.e., left, right, front, and back. Region anchors are grouped by their spatial grid indices. We denote the set of all valid groups as

$$\mathcal{G} = \mathcal{G}_{\text{bd}} \cup \mathcal{G}_{\text{reg}}. \tag{41}$$

For a group $g \in \mathcal{G}$, its coverage is defined as

$$c_g = \max_m \sum_{j: g_j = g} A^{m,j}_{\text{frame}}, \tag{42}$$

where g_j is the group id of the j-th frame primitive. This definition considers a group as covered if at least one frame state slot assigns sufficient attention to primitives in that group.

The frame coverage regularizer is

$$\mathcal{R}_{\text{frame}} = -\frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \log(c_g + \epsilon). \tag{43}$$

This term increases the probability that different spatial anchors are read by frame states, allowing S_{frame} to maintain boundary and region coverage.

A.5 Geometry Labels for Relation Regularization

Relation geometry regularization uses the center of entity evidence to generate coarse boundary and region labels. We first compute the weighted entity center:

$$\mu_{\text{ent}} = \sum_{j \in \mathcal{I}_{\text{ent}}} w_j u_j, \tag{44}$$

where

$$\mu_{\text{ent}} = (\mu_x, \mu_y, \mu_z). \tag{45}$$

Boundary label. We compute the distances from the entity center to four horizontal boundaries:

$$d_{\text{left}} = \mu_x - x_{\min}, \quad d_{\text{right}} = x_{\max} - \mu_x, \tag{46}$$

$$d_{\text{front}} = \mu_y - y_{\min}, \quad d_{\text{back}} = y_{\max} - \mu_y. \tag{47}$$

The boundary label is defined as the nearest boundary direction:

$$y_{\text{bd}} = \arg\min\{d_{\text{left}}, d_{\text{right}}, d_{\text{front}}, d_{\text{back}}\}. \tag{48}$$

This label provides a coarse directional reference from the entity evidence to the scene boundary.

Region label. The region label is generated according to the grid cell containing the entity center. We normalize the entity center coordinates as

$$\tilde{\mu}_x = \frac{\mu_x - x_{\min}}{x_{\max} - x_{\min} + \epsilon}, \quad \tilde{\mu}_y = \frac{\mu_y - y_{\min}}{y_{\max} - y_{\min} + \epsilon}. \tag{49}$$

The region label is

$$y_{\text{reg}} = \lfloor G_x \tilde{\mu}_x \rfloor + G_x \lfloor G_y \tilde{\mu}_y \rfloor. \tag{50}$$

In implementation, $\tilde{\mu}_x$ and $\tilde{\mu}_y$ are clamped to valid ranges to avoid numerical boundary issues.

A.6 Relation Geometry Regularization

Relation states are induced from the interaction between entity states and scene-frame states. To encourage them to carry coarse spatial references between object evidence and spatial anchors, we apply boundary and region classification losses to the pooled relation state:

$$\bar{S}_{\mathrm{rel}} = \mathrm{Pool}(S_{\mathrm{rel}}). \tag{51}$$

The boundary and region predictions are

$$\hat{y}_{\mathrm{bd}} = g_{\mathrm{bd}}(\bar{S}_{\mathrm{rel}}), \quad \hat{y}_{\mathrm{reg}} = g_{\mathrm{reg}}(\bar{S}_{\mathrm{rel}}), \tag{52}$$

where g_{bd} and g_{reg} are lightweight classification heads. The relation geometry regularizer is

$$\mathcal{R}_{\mathrm{rel}} = \mathrm{CE}(\hat{y}_{\mathrm{bd}}, y_{\mathrm{bd}}) + \mathrm{CE}(\hat{y}_{\mathrm{reg}}, y_{\mathrm{reg}}). \tag{53}$$

This is a coarse geometry-aware regularization. It encourages S_{rel} to encode relative references between object evidence, scene boundaries, and occupied regions.

A.7 Residual Object Dropout

SceneScaffold keeps Z_{res} to preserve the local detail capability of the original visual connector. During training, we apply lightweight dropout to the residual object buffer:

$$\tilde{Z}_{\mathrm{res}} = m \odot Z_{\mathrm{res}}, \quad m_i \sim \mathrm{Bernoulli}(1-p). \tag{54}$$

$\tilde{Z}_{\mathrm{res}}$ replaces Z_{res} during serialization in training. This serves as a state-usage regularizer, encouraging the model to use entity, frame, relation, and global states while preserving local object details. Dropout is disabled during inference.

A.8 Overall Training Objective

The main paper writes the training objective as

$$\mathcal{L} = \mathcal{L}_{\mathrm{task}} + \mathcal{R}_{\mathrm{role}}, \tag{55}$$

where

$$\mathcal{R}_{\mathrm{role}} = \lambda_{\mathrm{ent}}\mathcal{R}_{\mathrm{ent}} + \lambda_{\mathrm{frame}}\mathcal{R}_{\mathrm{frame}} + \lambda_{\mathrm{rel}}\mathcal{R}_{\mathrm{rel}}. \tag{56}$$

In implementation, the task loss contains a language modeling loss and a segmentation loss. The complete objective can be written as

$$\mathcal{L} = \mathcal{L}_{\mathrm{lm}} + \lambda_{\mathrm{seg}}\mathcal{L}_{\mathrm{seg}} + \lambda_{\mathrm{ent}}\mathcal{R}_{\mathrm{ent}} + \lambda_{\mathrm{frame}}\mathcal{R}_{\mathrm{frame}} + \lambda_{\mathrm{rel}}\mathcal{R}_{\mathrm{rel}}. \tag{57}$$

For visual grounding, the segmentation loss includes BCE, Dice, and decoder auxiliary losses:

$$\mathcal{L}_{\mathrm{seg}} = \lambda_{\mathrm{bce}}\mathcal{L}_{\mathrm{BCE}} + \lambda_{\mathrm{dice}}\mathcal{L}_{\mathrm{Dice}} + \mathcal{L}_{\mathrm{aux}}^{\mathrm{dec}}. \tag{58}$$

The overall objective combines task supervision, role-preserving supervision, and residual-object dropout as a lightweight state-usage regularizer.

B Additional Experimental Details

B.1 Diagnostic Subset Construction

We construct a relation-heavy and spatially ambiguous diagnostic subset to analyze whether SceneScaffold improves samples that require both spatial-reference understanding and multi-instance disambiguation. We first select samples whose expressions or questions contain explicit spatial or scene-reference words, including *left*, *right*, *front*, *behind*, *near*, *next to*, *beside*, *between*, *under*, *below*, *above*, *on top of*, *against*, *wall*, *window*, *door*, *floor*, *corner*, *closest*, *second*, and *third*. We then keep samples where multiple same-category candidates appear in the same scene. Samples satisfying both criteria form the Rel.+Amb. subset. The resulting subset statistics are summarized in Table B.1 (left).

Table B.1: **Additional experimental statistics.** Left: statistics of the diagnostic Rel.+Amb. subsets. Right: token-budget allocation for state-component ablations.

Diagnostic subset statistics

Dataset	Total	Rel.+Amb.
ScanRefer	9508	6329
Multi3DRefer	11120	6235
ScanQA	4675	2272
Scan2Cap	2007	1631

Token-budget allocation

Variant	Detail	Entity	Frame	Relation	Scene
Full	75	8	8	8	1
w/o S_{ent}	83	0	8	8	1
w/o S_{frame}	83	8	0	8	1
w/o S_{rel}	83	8	8	0	1
w/o S_{scene}	76	8	8	8	0

Table B.2: **Control baselines for role-factored state construction.** All variants keep the same final visual-token budget. Each baseline modifies one construction factor while preserving the overall state-interface scale.

Variant	Grounding		ScanQA				SQA3D		Scan2Cap			
	ScanRefer mIoU↑	Multi3DRefer mIoU↑	C↑	B-4↑	M↑	R↑	EM↑	EM-R↑	C@.5↑	B-4@.5↑	M@.5↑	R@.5↑
Generic query states	45.7	46.5	93.6	14.9	18.5	44.4	53.4	56.1	77.7	36.7	27.0	57.5
Shared-evidence role states	45.9	46.4	92.4	14.3	18.4	44.1	53.7	56.3	77.6	36.2	27.1	57.5
Object-centric frame evidence	45.8	46.7	94.2	15.2	18.7	44.7	54.1	56.7	77.5	36.1	27.0	57.3
Relation from primitive evidence	45.7	46.6	92.2	14.6	18.2	44.1	54.0	56.6	77.9	36.3	27.1	57.6
SceneScaffold Full	47.0	47.9	95.8	16.8	19.0	45.0	55.5	58.3	79.1	37.1	27.3	57.8

B.2 State Ablation Budget Allocation

All state-component ablations in Table 3 keep the total visual-token budget fixed at $K = 100$. The full SceneScaffold interface uses

$$75D + 8E + 8F + 8R + 1G = 100,$$

where D denotes residual detail tokens, and E, F, R, and G denote entity states, frame states, relation states, and the global scene state, respectively. When a state component is removed, its budget is reallocated to residual detail tokens. The detailed allocation is reported in Table B.1 (right).

B.3 Control Baselines for State Construction

We further evaluate several same-budget control baselines to analyze which factors are responsible for the role-factored scene-state interface. Different from the state-component ablation in Table 3, these variants keep the state-interface scale unchanged and modify how the states are constructed.

Generic query states replace the role-factored entity, frame, and relation states with homogeneous learned query states under the same state-slot budget. This baseline tests whether a generic learned token set is sufficient. **Shared-evidence role states** keep separate state readers and serialized role blocks, but use the same evidence source for different states, testing whether role-specific evidence grounding is necessary. **Object-centric frame evidence** replaces geometry-derived frame evidence with object-centric evidence, examining whether scene-frame states require boundary and region anchors. **Relation from primitive evidence** induces relation states directly from primitive evidence instead of the interaction between entity and frame states, testing the role of the entity-frame induction pathway.

Table B.2 shows that all construction-control variants underperform the full SceneScaffold model. Generic query states use the same state-slot budget but obtain lower grounding and QA results, indicating that a homogeneous learned state set provides weaker scene evidence than the role-factored interface. Shared-evidence role states also lag behind the full model, showing that separate role blocks benefit from evidence sources matched to their intended functions. Replacing geometry-derived frame evidence with object-centric evidence reduces grounding performance, which supports the use of boundary and region anchors for spatial reference preservation. Inducing relation states directly from primitive evidence weakens both grounding and language-oriented metrics, suggesting that relation states are more effective when formed through the interaction between entity and frame states. These results complement the state-component ablation in Table 3 by showing that SceneScaffold benefits from the coupled design of role-specific evidence, state-induction pathways, and fixed-budget serialization.

Table B.3: **Functional state intervention analysis on Rel.+Amb. subsets.** We replace one type of state with the corresponding state from another scene during inference. All results are evaluated on the diagnostic Rel.+Amb. subsets defined in Appendix B.1.

Intervention	ScanRefer mIoU↑	Multi3DRefer mIoU↑	ScanQA C↑	Scan2Cap C@.5↑
Full	40.6	43.5	88.6	77.9
Shuffle S_{ent}	39.1	42.0	87.4	76.7
Shuffle S_{frame}	38.4	41.3	87.2	76.5
Shuffle S_{rel}	38.6	41.6	86.9	76.2
Shuffle S_{scene}	39.8	42.8	86.6	76.4
Shuffle all states	37.8	40.8	85.0	76.0

B.4 Functional State Intervention Analysis

To complement the retrained state-component ablation, we conduct a functional state intervention analysis on a trained full SceneScaffold model. During inference, we keep the residual detail tokens, language input, model parameters, and all other states unchanged, and replace one type of state with the corresponding state sampled from another scene:

$$S_\tau \leftarrow S_\tau^{\mathrm{other}}, \quad \tau \in \{\mathrm{ent}, \mathrm{frame}, \mathrm{rel}, \mathrm{scene}\}.$$

Compared with zero masking, cross-scene replacement uses valid states produced by the same model, and therefore reduces the influence of obvious out-of-distribution perturbations.

We evaluate this intervention on the Rel.+Amb. diagnostic subsets defined in Appendix B.1. These subsets contain samples that require both explicit spatial-reference understanding and multi-instance disambiguation. Therefore, they provide a more focused stress test for examining whether the constructed states are functionally used in difficult 3D grounding and reasoning cases. We do not introduce additional entity-centric, frame-reference, or object-relation subsets; all results in this analysis are evaluated on the same Rel.+Amb. subsets.

Table B.3 reports functional state intervention results on the Rel.+Amb. subsets. The Full row is identical to the Rel.+Amb. performance of SceneScaffold in Table 2. Replacing individual state types with states from another scene consistently degrades performance, indicating that the constructed states are functionally used by the trained model. Shuffling S_{frame} causes clear drops on ScanRefer and Multi3DRefer, suggesting that frame states provide spatial-reference cues for relation-heavy and multi-instance disambiguation. Shuffling S_{rel} also weakens grounding and captioning performance, indicating that relation states contribute to object–object and object–environment relation modeling. Shuffling S_{scene} has a relatively smaller effect on grounding but a larger effect on ScanQA and Scan2Cap, which is consistent with its role as a compact global scene summary for language reasoning. Shuffling all states leads to the largest degradation and approaches the baseline performance on the Rel.+Amb. subsets, while still retaining residual detail tokens.

This analysis is not intended as a strict causal explanation of the LLM's internal reasoning process. Instead, it provides functional evidence that the constructed states are not interchangeable latent tokens: replacing different state types affects the difficult Rel.+Amb. subset in different ways.

B.5 Qualitative Results

We provide additional qualitative comparisons in Figure B.1, complementing the visual grounding cases in Figure 3 of the main text. Figure B.1 shows examples on question answering and dense captioning. Compared with the baseline, SceneScaffold produces answers and descriptions that better preserve spatial references, object identities, and scene context. These examples suggest that the constructed scene states are useful not only for localization-heavy tasks, but also for language generation tasks requiring spatial reasoning.

B.6 Additional Loss Ablation

We additionally ablate the role-preserving objectives used to stabilize different scene states. Different from the scene-state component ablation in the main text, this analysis keeps the model architecture

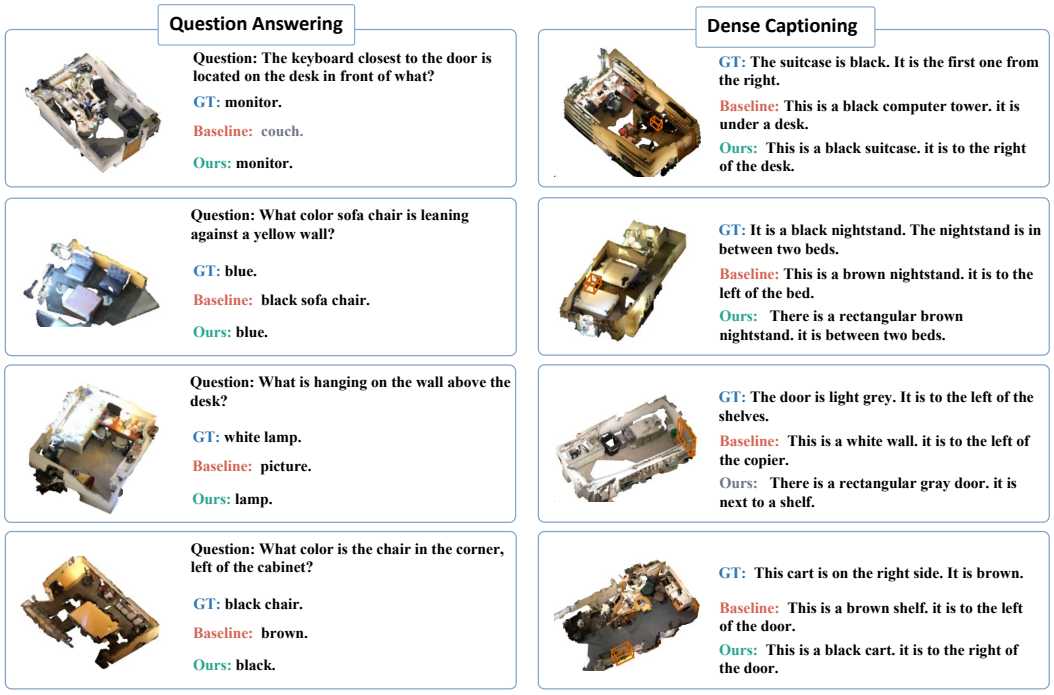


Figure B.1: **Qualitative results on question answering and dense captioning.** SceneScaffold produces responses that better match the target object identity and spatial context, while the baseline often confuses nearby objects or misses relation cues.

Table B.4: **Ablation on role-preserving objectives.** All variants keep the same model architecture and visual-token budget. Only the corresponding role-preserving objective is removed.

Variant	Grounding		ScanQA				SQA3D		Scan2Cap			
	ScanRefer mIoU↑	Multi3DRefer mIoU↑	C↑	B-4↑	M↑	R↑	EM↑	EM-R↑	C@.5↑	B-4@.5↑	M@.5↑	R@.5↑
Full	47.0	47.9	95.8	16.8	19.0	45.0	55.5	58.3	79.1	37.1	27.3	57.8
w/o $\mathcal{R}_{\text{ent}}$	46.4	47.1	95.1	16.4	18.8	44.7	55.1	58.0	78.4	36.8	27.1	57.6
w/o $\mathcal{R}_{\text{frame}}$	46.2	47.0	95.3	16.5	18.8	44.7	55.0	58.0	78.6	36.8	27.1	57.6
w/o $\mathcal{R}_{\text{rel}}$	46.0	46.8	94.7	16.2	18.6	44.4	54.9	57.7	77.9	36.5	27.0	57.4

and the visual-token budget unchanged, and only removes the corresponding training objective. Therefore, this experiment does not test whether a state component is necessary; instead, it examines whether the intended role of each existing state can be better maintained by its role-preserving objective.

Specifically, we consider entity semantic preservation $\mathcal{R}_{\text{ent}}$, frame coverage preservation $\mathcal{R}_{\text{frame}}$, and relation geometry regularization $\mathcal{R}_{\text{rel}}$. The entity objective encourages entity states to remain aligned with object-core evidence. The frame objective encourages frame states to cover diverse boundary and region anchors. The relation objective provides coarse geometry-aware supervision for relation states.

Table B.4 shows that removing any role-preserving objective weakens the model, while the degradation is generally smaller than directly removing the corresponding state component. This is expected because the state structure remains available, but its intended role is less explicitly stabilized during training. Removing $\mathcal{R}_{\text{ent}}$ leads to mild drops on object-sensitive tasks, suggesting that entity semantic preservation helps maintain object-core evidence. Removing $\mathcal{R}_{\text{frame}}$ affects grounding performance, indicating that frame coverage preservation helps maintain boundary and region references. Removing $\mathcal{R}_{\text{rel}}$ causes relatively larger drops on Scan2Cap and grounding metrics, suggesting that geometry-aware relation regularization helps stabilize relation states for spatially grounded description and referring tasks.

Overall, this ablation complements the main state-component ablation. The main ablation studies whether each state type is needed in the representation, while this analysis studies whether the roles of existing states are better maintained by the corresponding training objectives.

C Limitation and Broader Impact

Limitations. SceneScaffold is currently evaluated mainly on indoor, static, ScanNet-style environments. Its scene-frame states rely on coarse boundary and occupied-region anchors, and its relation states capture lightweight geometry-aware interaction cues rather than complete scene graphs or full physical topology. Extending this framework to outdoor scenes, dynamic settings, long-horizon embodied interaction, and noisy real-time perception remains future work. In addition, although our ablations and functional interventions provide evidence that the states contribute complementary information, they do not fully explain how the LLM internally uses each state. More adaptive state allocation and stronger interpretability tools may further improve the framework.

Broader impact. SceneScaffold does not introduce new datasets with personal or biometric information. It may benefit embodied AI, assistive robotics, augmented reality, indoor navigation, and spatially aware human-computer interaction by improving 3D spatial grounding and reasoning. However, inaccurate 3D understanding in safety-critical or privacy-sensitive environments may lead to unsafe actions or expose sensitive indoor layouts. Practical deployment should therefore include human oversight, uncertainty estimation, fail-safe mechanisms, privacy-preserving data handling, and compliance with dataset licenses and consent requirements.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction state the main claim that SceneScaffold reorganizes the fixed 3D visual bottleneck into role-factored scene states. The experimental section evaluates this claim on grounding, question answering, and dense captioning benchmarks, including diagnostic and ablation studies.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss limitations in Appendix.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: The paper does not present formal theoretical results or proofs.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: The paper specifies datasets, metrics, model backbone, token allocation, training objective, optimizer, learning rate, batch size, epochs, hardware, and evaluation protocols in the main text and appendix.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: We will provide these.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: Section 4.1 reports the evaluated datasets, metrics, backbone, visual-token budget, state allocation, trainable modules, optimizer, learning rate, batch size, number of epochs, and hardware.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [No]

Justification: We do not report error bars or confidence intervals because full multi-seed training across all 3D-LMM benchmarks is computationally expensive. We instead provide consistent results across multiple tasks, diagnostic subsets, component ablations, and functional interventions.

Guidelines:

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

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The paper reports that experiments are trained on 8 NVIDIA A800 GPUs.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The work uses standard public 3D scene understanding benchmarks and does not introduce personal, biometric, or sensitive identity data.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Appendix discusses potential positive applications in embodied AI, assistive robotics, as well as risks related to unsafe actions and privacy-sensitive indoor layouts.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The paper does not release a high-risk pretrained generative model or a newly collected dataset.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: The paper cites the existing datasets, models, and benchmarks used in the experiments.

    Guidelines:

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

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

13. **New assets**

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

    Answer: [N/A]

    Justification: The paper does not introduce a new dataset or benchmark.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: Our paper does not involve this.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: Our paper does not involve this.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

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

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: Any use of language-model tools, if applicable, was limited to writing or formatting assistance and did not affect the core methodology, experiments, or scientific claims.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.