3DLLM-MEM: Long-Term Spatial-Temporal Memory for Embodied 3D Large Language Model

Wenbo Hu[1] Yining Hong[1] Yanjun Wang[1] Leison Gao[1] Zibu Wei[1]
Xingcheng Yao[1] Nanyun Peng[1] Yonatan Bitton[2] Idan Szpektor[2] Kai-Wei Chang[1]

[1]University of California, Los Angeles, [2]Google Research

https://3dllm-mem.github.io

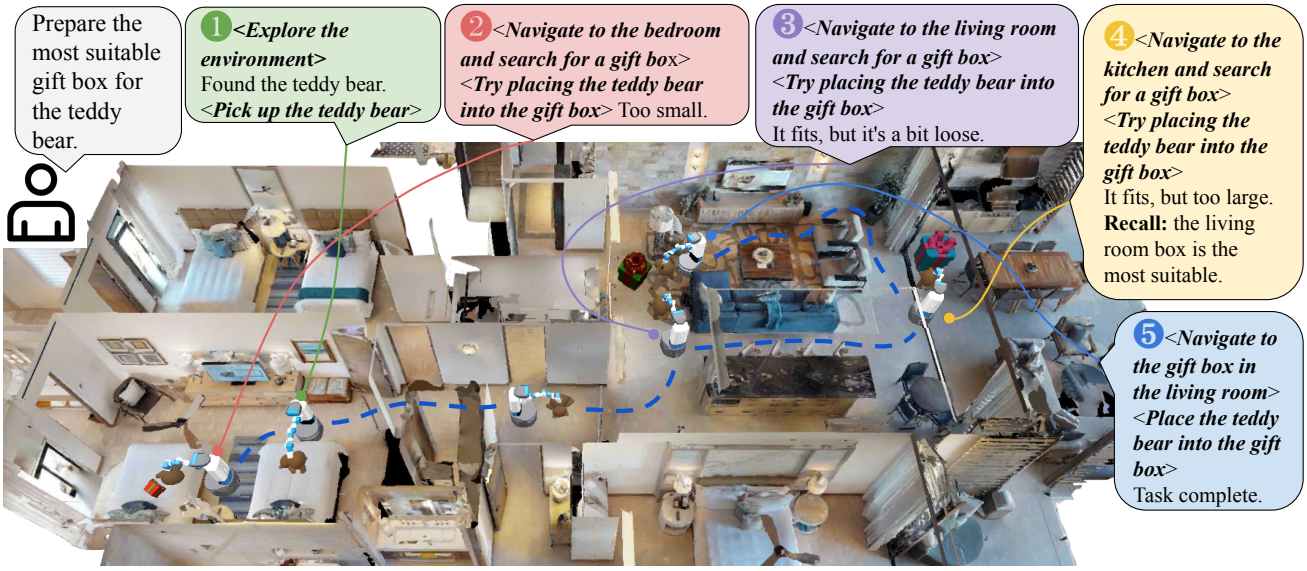


Figure 1. We propose 3DLLM-MEM, a memory-enhanced 3D embodied agent that explores and incorporates feedback from the environment, interacts with objects, and incrementally builds and maintains a task-relevant long-term memory throughout its trajectory. For illustration purposes, agents from multiple time steps are shown simultaneously.

Abstract

Humans excel at performing complex tasks by leveraging long-term memory across temporal and spatial experiences. In contrast, current Large Language Models (LLMs) struggle to effectively plan and act in dynamic, multi-room 3D environments. We posit that part of this limitation is due to the lack of proper 3D spatial-temporal memory modeling in LLMs. To address this, we first introduce 3DMEM-BENCH*, a comprehensive benchmark comprising over 26,000 trajectories and 2,892 embodied tasks, question-answering and captioning, designed to evaluate an agent's ability to reason over long-term memory in 3D environments. Second, we propose* 3DLLM-MEM*, a novel dynamic memory management and fusion model for embodied spatial-temporal reasoning and actions in LLMs. Our model uses working memory tokens, which represents current observations, as queries to selectively attend to and fuse the most useful spatial and temporal features from episodic memory, which stores past observations and interactions. Our approach allows the agent to focus on task-relevant information while maintaining memory efficiency in complex, long-horizon environments. Experimental results demonstrate that* 3DLLM-MEM *achieves state-of-the-art performance across various tasks, outperforming the strongest baselines by 16.5% in success rate on* 3DMEM-BENCH*'s most challenging in-the-wild embodied tasks.*

1. Introduction

Picture yourself traversing an unfamiliar home, as illustrated in Figure 1, on a mission to explore multiple rooms and evaluate various gift boxes to find the most suitable one for wrapping a teddy bear. As you navigate from room to room, your brain instinctively creates a 3D cognitive map of the environment, maintains a working memory of objects you've encountered, forms episodic memories that link observations across space and time, and plans efficient actions. This seamless integration of 3D spatial understanding, long-term memory encoding and retrieval, fluid switching between working and episodic memory, and purposeful action planning — cognitive processes that humans take for granted — remain formidable challenges for embodied AI systems today.

Recent extensions of Large Language Models (LLMs) to 3D environments have birthed 3D-LLMs [14, 15, 17, 20, 45] that can perceive and reason about 3D spaces, while 3D Vision-Language-Action models [21, 54, 55] further incorporate the ability to plan and act within these environments. Despite these advances, several critical limitations persist that prevent models from performing the kinds of tasks described above. First, current models struggle to maintain long-term memory chains when performing complex tasks that unfold across multiple visual scenarios, such as several rooms in a house, and extended time frames. Real-world 3D physical scenes are remarkably vast and information-dense, where every detail can matter for long-horizon embodied tasks — for instance, in Figure 1, finding the most suitable gift box requires remembering all the gift boxes encountered along the way and their characteristics and interaction with teddy bear. Dense 3D representations are particularly valuable as they capture comprehensive spatial information, preserving intricate geometric relationships and environmental details that sparse or object-centric approaches might miss. However, how to accurately and efficiently store dense 3D memory remains a fundamental challenge - retrieving the entire history would overwhelm the model's context limits, while selective retrieval [42, 44, 49] risks omitting critical information needed for accurate reasoning and decision-making. The second challenge resides in the entanglement of spatial and temporal memory — agents must track both where objects are and how they change over time through exploration and interaction. As environments evolve, maintaining coherent representations of previously seen spaces while incorporating new information continues to exceed the capabilities of current embodied AI models.

Our efforts at solving this challenge are two-fold. First, we introduce a novel benchmark for reasoning, planning and acting with long-term spatial-temporal memory in embodied environments. Our benchmark, 3DMEM-BENCH, encompasses multi-room 3D scenes from the Habitat environment, augmented with interactive objects to enable manipulation tasks across extended spatial-temporal horizons. Notably, we define fine-grained embodied tasks across varying levels of difficulty—from simple to hard—enabling deeper insight into model performance, which we believe is not addressed in prior benchmarks as shown in Table 1. Our task set spans a wide range of complexities, from straightforward object collection to challenging comparative reasoning tasks that require integrating observations across multiple rooms and time steps. Additionally, we include in-the-wild challenge tasks to evaluate the model's generalization capabilities beyond seen environments. The benchmark includes three evaluation categories: (1) embodied tasks requiring extended action sequences across multiple rooms, (2) spatial-temporal embodied question answering (EQA) that evaluates understanding of spatial relationships over time, and (3) long-term scene captioning that tests memorization of previously observed environments. Our dataset includes 26,000+ trajectory examples spanning 182+ unique scenes with an average of 18 rooms per scene.

Second, we introduce 3DLLM-MEM, a 3D embodied LLM with dynamic memory management capabilities designed specifically for embodied spatial-temporal reasoning, planning and acting. To our knowledge, we are among the first to explore dense 3D representations as memory for embodied 3D LLMs — addressing a significant gap in current research as noted in [49]. Unlike standard approaches that rely solely on context windows [17, 20, 57], 3DLLM-MEM implements a dual-memory system: a limited-capacity working memory for current observations and an expandable episodic memory that stores past spatial-temporal information as dense 3D representations. The key innovation is our memory fusion module that actively integrates information from both memory systems based on task relevance and spatial-temporal relationships. This allows the model to leverage the benefits of dense 3D representations while mitigating their computational demands, maintaining coherent spatial-temporal understanding across extended task horizons. The fusion process preserves critical spatial relationships while accounting for their evolvement through agent interactions over time.

We evaluate popular 3D-LLMs and memory mechanisms on 3DMEM-BENCH. Experimental results demonstrate 3DLLM-MEM significantly outperforms all existing approaches in both in-domain and in-the-wild embodied tasks. Notably, while the performance of other methods drops sharply in the challenging in-the-wild setting, our method remains robust, achieving an average success rate of 32.1%—demonstrating strong generalization capabilities. As task complexity increases from simple to hard, all existing approaches degrade significantly, achieving only $\sim$5% success rate in hard in-the-wild tasks. In contrast, 3DLLM-MEM maintains a strong performance of 27.8%, demonstrating its scalability and effectiveness in managing

Benchmark	#Test Tasks	#Train Traj.	Memory	Fine-grain	EQA	Cap.
ALFWorld [35]	274	3,553	×	×	NA	NA
Behavior-1K [25]	1,000	NA	×	×	NA	NA
VisualAgentBench [27]	746	4,482	×	×	NA	NA
EmbodiedBench [48]	1,128	NA	×	×	NA	NA
3DMEM-BENCH (ours)	1,860	26,276	✓	✓	865	167

Table 1. Comparison with related benchmarks. 3DMEM-BENCH focus on spatial-temporal memory through fine-grained embodied tasks and EQA that span multiple "pieces" of long-term memory, distinguishing it from prior benchmarks that typically target single-step or short-horizon reasoning. Train Traj. indicates train trajectories. Fine-grain is short for fine-grained complexity which indicates our embodied task spans from simple to medium to hard. Cap. stands for Captioning.

longer-term memory representations.

Our contributions can be summarized as below:

- We propose a novel task that requires agents to execute action chains while maintaining and utilizing long-term spatial-temporal memory.
- We construct 3DMEM-BENCH, a comprehensive benchmark comprising over 26,000 trajectories and 1,860 fine-grained long-term memory embodied tasks—ranging from simple to hard—along with question-answering tasks that target memory changes across time and space, and captioning tasks in complex 3D environments.
- We propose 3DLLM-MEM, an embodied 3D LLM with a novel memory fusion module for spatial-temporal reasoning, planning, and acting-which utilizes working memory tokens as queries to selectively fuse relevant features from episodic memory for efficient, task-aware decision-making.
- Experimental results on embodied tasks, question-answering, and captioning demonstrate that 3DLLM-MEM outperforms baselines by a large margin.

2. The Embodied 3D Long-Term Spatial-Temporal Memory Benchmark

2.1. Overview of 3DMEM-BENCH

Design principles Long-term memory [3, 12, 58] can be categorized into *explicit memory* and *implicit memory*. Explicit memory includes *semantic memory*, which stores general knowledge and facts about the world, and *episodic memory*, which consists of personal experiences that are time-stamped and context-specific. In contrast, implicit memory primarily involves *procedural memory*, such as learned skills and habits.

To comprehensively evaluate 3D long-term memory for real-world applications, we design 3DMEM-BENCH following three core task categories: embodied tasks, long-term memory EQA, and captioning. As illustrated in Figure 2, *embodied tasks* require an embodied agent to solve realistic indoor environment challenges by leveraging both implicit and explicit long-term memory. *Long-term memory EQA* tests the agent's ability to answer complex embodied questions using spatial-temporal memory. This task includes five subcategories: spatial reasoning questions, long-term object navigation, comparative reasoning, multi-room layout understanding, and semantic object counting. *Captioning* tasks involve summarizing the agent's episodic memory to highlight shared and distinctive features across experiences, enabling more informed decision-making under the current task context.

2.2. Data Collection

Base environment construction We build our scenes on top of the Habitat-Matterport 3D (HM3D) semantics dataset [31], which has 1000 3D spaces and 10,600 rooms within those spaces. Pre-processing for the axis-aligned bounding box and using valid semantic label annotation, we filter to 182 3D spaces and 2,602 rooms. However, existing objects in HM3D scene are not interactive in Habitat-sim [36]. To expand our task diversity and enable embodied tasks, we add interactive objects from Objaverse [10] which consists of 800K 3D objects spanning rich categories. More environment construction details are illustrated in Appendix B.

Generating task trajectories Following Hong et al. [17, 18], we adopt box-demonstration-instruction-based prompting, which utilizes the axis-aligned bounding boxes (AABB) of both rooms and objects within the 3D scenes to prompt Gemini [38] to generate diverse tasks. We further prompt Gemini to incorporate interactive objects based on task requirements and their appropriateness within indoor environments. Detailed prompt instructions and few-shot demonstration examples are provided in Appendix E. To ensure the validity of the generated trajectories, we develop a trajectory simulation pipeline that verifies each trajectory step-by-step. At every step, the simulator checks: (1) the correctness of the agent's location, (2) the existence and validity of referenced objects, and (3) the correctness of pick-up and put-down actions. Finally, we ensure that high-level actions can be executed in the simulator, following [37, 48]. Details of this implementation are in Appendix F.1. On average, our filtering process yields a validation rate of approximately 24%, ensuring the correctness and feasibility of the generated trajectories.

Embodied data collection In our task settings, an embodied agent first performs random exploration within the environment to collect RGB-D observations and corresponding camera poses. Then the agent follows the task trajectory, incrementally exploring new environments, executing interaction actions, and receiving feedback with new RGB-D observation data. All interaction results are

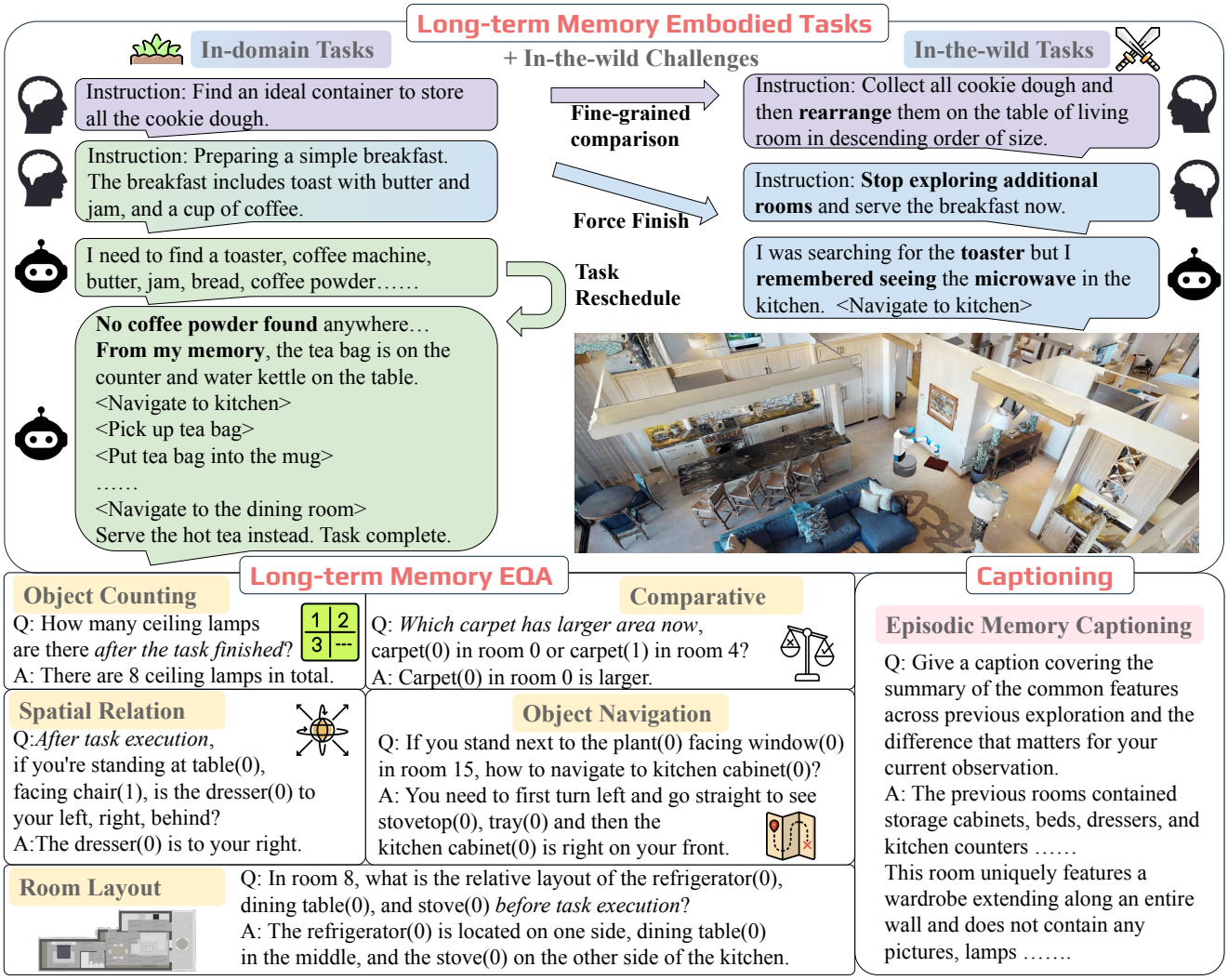


Figure 2. Overview of 3DMEM-BENCH. For long-term memory embodied tasks, we further incorporate in-the-wild challenges to test 3D agent's generalization abilities. For complete embodied task trajectories, please refer to Appendix C.

recorded and the reconstructed point cloud data is precomputed and stored locally to enable faster loading during both training and inference.

2.3. Data Curation

As mentioned previously, we collect embodied data by prompting Gemini. To enable a fine-grained analysis of long-term memory capacity, we divide the tasks into three subcategories: *simple*, *medium*, and *hard*, comprising of 3, 5 and 10 multi-room scene settings respectively. In total, we collect 51K trajectories, with 31K in the simple setting, 10K in the medium, and 10K in the hard.

To construct in-domain evaluation sets, we first remove training tasks and filter for instances that never shown in the agent's working memory. For the in-the-wild evaluation set, we apply additional filtering to assess the agent's generalization capabilities. Specifically, we select instances involving unseen objects and entirely unseen memory context, and we introduce novel in-the-wild challenges that differ from those encountered during training, as illustrated in Figure 2.

For EQA data curation, we extract complete trajectories explored by agents and then prompt Gemini to generate question-answer pairs. The questions are categorized into spatial reasoning, long-term object navigation, comparative reasoning, multi-room layout understanding, and semantic object counting. As shown in Figure 2, these questions evaluate models on spatial-temporal changes in memory during embodied task execution. For long-term memory captioning, which primarily targets semantic episodic memory, we collect data across multiple rooms before and after the execution of each trajectory, enabling comparison and summa-

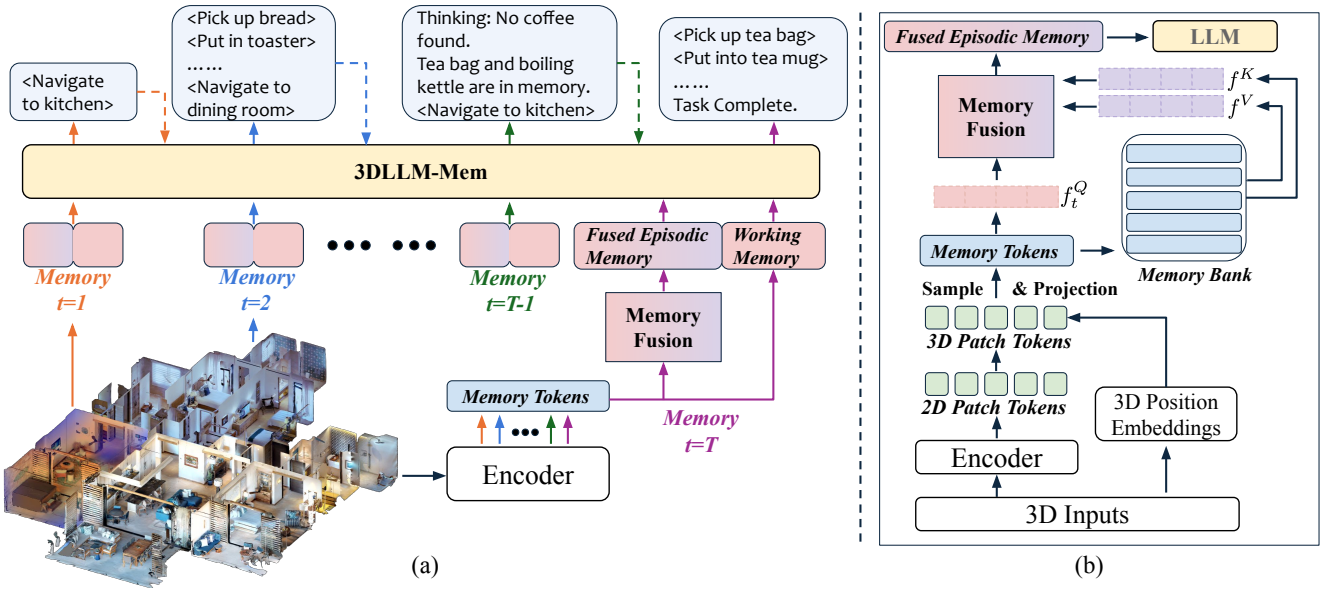


Figure 3. (a) We propose 3DLLM-MEM, a memory-enhanced 3D embodied agent that gradually form its long-term memory while executing tasks. Multiple timesteps are shown together but in different colors, with each timestep's memory including the prior one. The task is "prepare a simple breakfast" as shown in Figure 2. (b) Overview of our memory fusion mechanism.

rization of memory-relevant experiences.

Quality control After constructing the entire benchmark, we implement two quality control procedures: automatic validation using trajectory simulation rules and a manual review of each benchmark instance. The automatic check involves re-running the trajectory simulation validation pipeline, as described in §2.2, particularly for the in-the-wild tasks. For human validation, four student experts in the field manually inspect each benchmark example. We render multi-view images of the entire scene using the simulator and verify whether the benchmark annotations accurately correspond to the simulated environment. More details are in Appendix F.2.

3. 3D Long-Term Spatial-Temporal Memory Model (3DLLM-MEM)

3.1. Preliminary

Recent work on 3D Large Language Models (3D-LLMs) has showcased robust capabilities. We choose LLaVA-3D [57] as the base model to build our long-term memory 3D-LLM. LLaVA-3D directly builds on 2D-LLM with multi-view images as input and utilizing the 3D position embeddings to bring the 2D patches within a 3D spatial context to construct 3D patches. For each frame image, a CLIP encoder splits the image $X \in \mathbb{R}^{3\times W\times H}$ into patches at the patch size P. For each 3D scene, V multi-view image patch features are encoded and then projected into LLM space as $X_p \in \mathbb{R}^{V\times d\times w\times h}$, where $h = \lfloor \frac{H}{P} \rfloor, w = \lfloor \frac{W}{P} \rfloor$, and d represents LLM's hidden dimension. The 3D positions in the 3D world are obtained with known depth image, camera intrinsic and extrinsic parameters and are further encoded into 3D position embeddings $P \in \mathbb{R}^{V\times d\times w\times h}$. These are directly added to the 2D patch visual tokens X_p, resulting in pixel-aligned 3D patches $X_{3D} \in \mathbb{R}^{V\times d\times w\times h}$. To reduce redundancy in 3D patches, we adopt the Farthest Point Sampling (FPS) strategy to downsample the 3D features to a fixed number of tokens, resulting in $X_{\text{3D Feat}} \in \mathbb{R}^{N\times d}$.

3.2. 3DLLM-MEM Memory Module

A 3D embodied agent gradually explores the environment by collecting observations and interacting with surrounding environments. For humans, current observations are held in *working memory*, while longer-term observations and experiences are stored in *episodic memory*. Inspired by human cognitive structure, 3DLLM-MEM is designed with a similar paradigm as illustrated in Figure 3. The current observation at time step $t = i$, denoted as $X^{[t=i]} \in \mathbb{R}^{N\times d}$, remains within the context window and serves as the agent's *working memory*. As the agent accumulates more experiences, past observations from time steps 1 to T, represented as $X^{[t=1:T]} \in \mathbb{R}^{T\times N\times d}$, are stored as part of its *episodic memory*, where T denotes the total number of timesteps.

Episodic memory To manage episodic memory, we propose the use of a memory feature bank. For each observation at time step j, where $1 \le j \le T$, we first apply a multi-layer perceptron (MLP) layer to project the observation into a memory-specific feature space, which is then stored in the memory bank for future retrieval. To further enhance the temporal understanding of the agent's explo-

ration, we incorporate sinusoidal positional embeddings to encode each time step $t = j$, and then directly added to the corresponding memory feature representations.

Memory fusion Our motivation is that an agent should leverage its current observations to recall the most relevant information from its episodic memory in order to complete the current task. To achieve this, we propose a mechanism called *3D memory fusion*. Specifically, we encode the 3D features from the working memory into a shared memory space and use this representation as the query feature, denoted as $f_t^Q \in \mathbb{R}^{N\times M}$, where M is the dimensionality of the memory feature space.

The episodic memory bank stores the corresponding key and value features from past observations: $f^K \in \mathbb{R}^{T\times N\times M}$ and $f^V \in \mathbb{R}^{T\times N\times M}$, respectively. Here, T is the number of past timesteps and N is the number of memory tokens per timestep. This structure allows the agent to retrieve task-relevant information through memory-query attention. The fused memory feature is then concatenated with the working memory feature to produce the final memory-enhanced representation f^M for the agent:

$$f^Q_{\text{fuse}} = \text{Softmax}\left(\frac{f^Q_t (f^K)^\top}{\sqrt{C}}\right) f^V \tag{1}$$

$$f^M = \text{Concat}\left[f^Q_{\text{fuse}}; f^Q_t\right] \tag{2}$$

Memory update The working memory is dynamic and updated online. As the agent interacts with the environment, changes in the environment are immediately reflected in the working memory through updated 3D representations. When the agent moves to a new environment, the previous working memory is transferred to the episodic memory bank. If the corresponding environment already exists in the memory bank and has been modified by the agent, the memory entry is updated accordingly. Thus, the memory bank remains dynamic and reflects the latest state of the explored environments. As described in §2.2, environment changes and corresponding observations are pre-collected and stored locally to facilitate efficient data loading during both training and inference.

4. Experiments

In this section, we first introduce the experimental setup and existing memory management baselines in §4.1. Then, we benchmark existing approaches on 3DMEM-BENCH, and present comprehensive results on embodied tasks, EQA, and captioning tasks to demonstrate the effectiveness of our 3DLLM-MEM in §4.2, along with qualitative results. Finally, in §4.3, we conduct an ablation study of key design choices in 3DLLM-MEM, demonstrating the effectiveness of our proposed memory fusion mechanism.

4.1. Experimental Setup

Implementation details We implement our model based on LLaVA-3D [57], modifying it to be compatible with Google TPUs with PyTorch/XLA frameworks [30, 39] . We first expand the model's context window to 8192 tokens to accommodate long-term memory inputs. We then fine-tune our proposed memory module along with the LLM decoder using our training split. Training is conducted on 8 Google Cloud TPU v5p cores with a batch size of 256. Our model is trained using supervised fine-tuning (SFT) with a standard language modeling loss. More details are provided in Appendix D.

Baselines We compare 3DLLM-MEM against a broad range of memory management approaches:

- **Everything in Context.** For a small subset of scenes, it is feasible to fit all observations directly into the model's context window.
- **Most Recent Memory.** Since retaining all observations in context is infeasible, we keep the most recent observations, assuming they are most relevant to the current task.
- **Retrieval-Augmented Memory.** Inspired by retrieval-based techniques, we adopt a memory bank that stores past observations. During inference, the most relevant memory entries are retrieved and appended before the working memory to augment reasoning.
- **3D-LLM** [17]. A popular 3D LLM recognized by the community. We finetune it on our training data and report its performance using the "everything in context" strategy with the longest context window supported. Further details are provided in Appendix G.
- **3D-Mem** [49]. A framework designed for 3D scene memory in embodied exploration and reasoning. However, this method does not support embodied interaction or action execution.

4.2. Experimental Results

Results on embodied tasks As shown in Table 2a, 3DLLM-MEM significantly outperforms all existing approaches on both in-domain and in-the-wild embodied tasks. Notably, while the performance of other methods drops sharply in the in-the-wild setting, our method demonstrates strong generalization capabilities with a average success rate of 32.1%. 3D-LLM showcases the lowest performance even under simple task settings, highlighting the necessity of incorporating an explicit memory module. Both the *Most Recent Memory* and *Retrieval-Augmented Memory* (RAG) baselines perform poorly in this setting, with RAG showing only a slight improvement, highlighting the challenges of retrieving relevant episodic memory. Interestingly, the *Everything in Context* baseline performs better than both recent memory and RAG approaches, suggesting that when all information can fit within the context window,

Model	Simple				Medium				Hard				Average			
	In-domain		In-the-wild		In-domain		In-the-wild		In-domain		In-the-wild		In-domain		In-the-wild	
	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR
3D-LLM (Finetuned)	10.4	20.3	9.1	18.5	-	-	-	-	-	-	-	-	-	-	-	-
Everything in Context	35.5	63.9	32.4	45.2	-	-	-	-	-	-	-	-	-	-	-	-
Most Recent Memory	32.8	62.3	23.4	38.6	20.1	34.8	12.4	25.3	10.4	20.7	5.4	12.1	21.1	39.3	13.7	25.3
Retrieval-Augmented Memory	34.2	63.0	28.3	46.2	21.8	40.2	13.7	28.0	10.8	21.6	4.8	10.6	22.3	41.6	15.6	28.3
3DLLM-MEM (Ours)	45.5	73.4	37.0	65.4	36.8	67.8	31.6	57.4	30.5	46.2	27.8	42.1	37.6	62.5	32.1	55.0

(a) Results on 3DMEM-BENCH embodied tasks. SR stands for success rate. Sub-SR stands for sub-success rate. Our model outperforms existing approaches by a large margin.

Model	Embodied Task		Embodied Question Answering (EQA)					Captioning		
	In-domain	In-the-wild	Spatial	Nav.	Comparative	Layout	Count	BLEU1	BLEU4	METEOR
3D-LLM (Finetuned)	-	-	2.9	5.8	0.0	7.7	0.0	42.3	12.0	30.6
3D-Mem (GPT4-o)	-	-	39.9	11.0	25.8	19.1	7.8	41.7	4.7	31.8
3D-Mem (Gemini-2.5-Flash)	-	-	41.6	18.2	37.6	30.2	12.7	42.8	4.8	29.6
3D-Mem (Gemini-2.5-Pro)	-	-	39.7	27.7	36.0	35.2	16.4	41.5	3.0	28.6
Most Recent Memory	21.1	13.7	27.5	30.2	24.3	20.1	10.5	32.4	10.1	25.6
Retrieval-Augmented Memory	22.3	15.6	38.0	33.4	31.8	29.7	15.6	40.8	11.5	29.3
3DLLM-MEM (Ours)	37.6	32.1	62.8	40.6	41.4	39.9	26.3	58.2	18.8	37.3

(b) Results on all tasks in 3DMEM-BENCH. Average success rate is reported for embodied tasks. *Nav.* stands for long-term object navigation. We report accuracy score for open-ended EQA evaluation and follow the standard LLM-as-judge evaluation protocol by prompting Gemini. Evaluation details are provided in Appendix E.

Table 2. Comparison with 3D memory models and standard memory management approaches. Our model, 3DLLM-MEM, achieves the best performance across embodied, EQA and captioning tasks.

the model can effectively utilize it. However, 3DLLM-MEM still outperforms *Everything in Context*, indicating the benefits of selectively fusing task-relevant memory features to better guide embodied reasoning and execution. As task complexity increases from simple to hard, all existing approaches degrade significantly, achieving only $\sim$5% success rate in hard in-the-wild tasks. In contrast, 3DLLM-MEM maintains a strong performance of 27.8%, demonstrating its scalability and effectiveness in managing longer-term memory representations.

Results on long-term EQA and captioning As shown in Table 2b, 3DLLM-MEM consistently outperforms all existing approaches across all tasks in our benchmark. Notably, 3D-LLM achieves the second-best performance on the captioning task, highlighting its strong ability to summarize object-centric semantic memory. However, due to limited context length, it performs poorly on the EQA tasks, which require long-term spatial-temporal reasoning. 3D-Mem demonstrates improved performance in EQA over other baseline approaches. However, it falls short on spatial relation, navigation and object counting tasks, indicating the limitation of relying solely on aggregated image-centric memories. 3DLLM-MEM significantly outperforms both *Most Recent Memory* and *RAG Memory*, which further demonstrates the effectiveness of our memory fusion technique.

Qualitative results We provide qualitative examples in Figure 4 and a more detailed version with explanations in Figure 6 (Appendix H), demonstrating that 3DLLM-MEM is capable of maintaining long-term memory and executing complex tasks in embodied environments. Additional examples are included in the **supplementary materials**.

4.3. Ablation Study

Our approach initializes the fused memory using working memory features, aiming to fuse the most relevant memories for the current task. We ablate several design choices for initializing the fusion query, as shown in Table 3 (Appendix A). When using either the most recent episodic memory or learnable zero parameters, performance degrades compared to our proposed method. Interestingly, using the most recent memory outperforms zero initialization in the simple setting but underperforms in the hard setting. One possible explanation is that recent memory initialization encourages fusion with nearby observations, which may be sufficient for simple tasks and leads to faster convergence. In contrast, zero initialization is guided solely by training supervision to learn which memories are most useful. In summary, the ablation results demonstrate that initializing fusion queries with working memory tokens provides the most effective and robust design choice for long-term memory fusion.

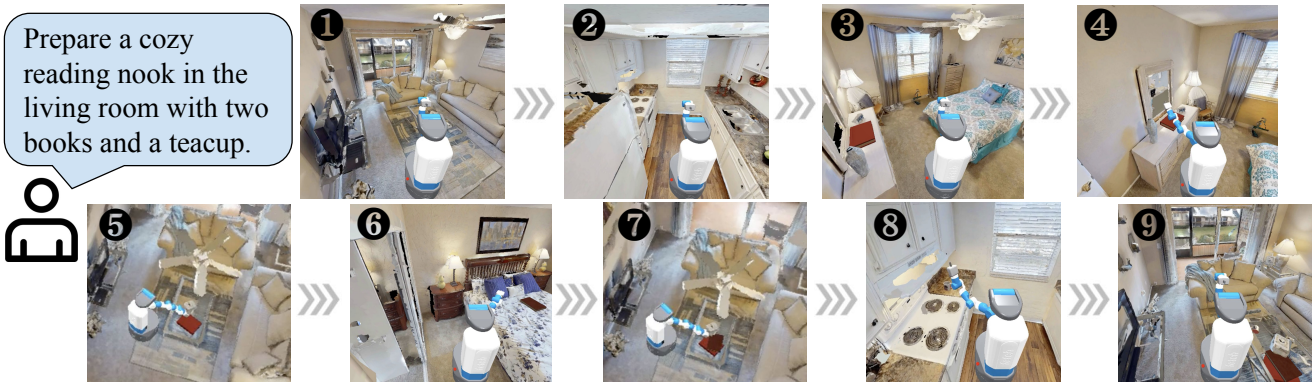


Figure 4. Qualitative example of 3DLLM-MEM, which maintains and utilizes a long-term memory to complete the task. Detailed task execution trajectory can be found in Figure 6.

5. Related Works

3D Large Language Models 3D Large Language Models (3D-LLMs) have demonstrated promising results across a wide variety of tasks, including 3D scene understanding, object detection, and segmentation [6, 17, 19, 45, 56]. In parallel, 3D embodied agents have expanded these capabilities to planning and action in interactive environments [1, 2, 5, 20]. Yet, existing models face significant challenges when performing long-horizon embodied tasks in densely populated 3D environments that require reasoning over long-term spatial-temporal memory. To address this, we propose an explicit memory module inspired by the structure of human implicit and explicit memory. Our model employs a memory fusion mechanism that efficiently retrieves and learns task-relevant information, resulting in enhanced performance on complex embodied tasks.

Long-term Embodied Trajectories Embodied AI simulators [4, 23, 32, 36] have fostered the development of embodied AI agents. Grounded in these environments, some existing benchmarks focus on high-level planning tasks, typically involving short trajectories that can often be completed within single-room settings, thereby requiring minimal spatial-temporal memory [25, 26, 34, 35, 37, 48]. Other benchmarks emphasize long-term scene exploration with extended trajectories, but are primarily centered around navigation tasks and often lack embodied interaction support [9, 22, 24, 31]. To bridge this gap, we introduce 3DMEM-BENCH, a benchmark specifically designed to evaluate long-horizon task execution that requires rich spatial-temporal memory and full embodied task support, as summarized in Table 1.

Embodied Question Answering Benchmark Embodied Question Answering (EQA) benchmarks [8, 43, 50] have been developed to advance goal-driven agents that can perceive their environment. Some EQA benchmarks also include embodied memory QA evaluation, such as OpenEQA [28], which includes an episodic memory QA split, and Yang et al. [47], which focuses on spatial memory QA. In contrast, our benchmark, 3DMEM-BENCH jointly targets both spatial and episodic memory, especially their changes over time, while also supporting embodied action tasks, EQA and captioning. For specific comparison on EQA, our long-term memory EQA tasks are designed to require reasoning over multiple "pieces" of memory and their changes across time and space. Additionally, we consider the agent's location in the scene at the moment of answering each question during evaluation.

Memory System Memory is a fundamental component of AI systems, with early work in the context of LLM agents that utilize memory for decision-making in web-based and sandbox environments [29, 33, 51, 52]. Most existing approaches construct an experience pool or memory bank and focus on improving the retrieval of useful past information [13, 46, 53]. In the computer vision domain, temporal memory has been studied extensively in video understanding and generation tasks [11, 41], while spatial memory has been applied to scene-level visual understanding and 3D reconstruction [40, 59]. Recent work such as 3D-Mem [49] has investigated 3D scene memory for exploration and reasoning by prompting vision-language models. In contrast, our work focuses on dense 3D memory representations that are critical for real-world embodied scenarios, where task execution depends heavily on maintaining and reasoning over long-term spatial-temporal memory.

6. Conclusion

In this work, we introduce 3DMEM-BENCH, a comprehensive benchmark containing fine-grained long-term memory embodied tasks—ranging from simple to hard—along with question-answering tasks that target memory changes across time and space, and captioning task in complex 3D environments. We propose 3DLLM-MEM, an embodied

3D-LLM with novel memory fusion approach for spatial-temporal reasoning, planning, and acting. One limitation of our model is that currently 3DLLM-MEM does not involve low-level navigation and control policy, but utilizes high-level pre-defined policies in simulator for carrying out the actions. We think that such aspects are orthogonal to our study, and could be explored and seamlessly integrated into our framework in the future.

References

[1] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2024. 8

[2] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alex Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control. 2023. 8

[3] Eduardo Camina and Francisco Güell. The neuroanatomical, neurophysiological and psychological basis of memory: Current models and their origins. *Frontiers in Pharmacology*, 8:438, 2017. 3

[4] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)*, 2017. 8

[5] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Dorsa Sadigh, Leonidas J. Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 14455–14465. IEEE, 2024. 8

[6] 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*, abs/2405.10370, 2024. 8

[7] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *ArXiv preprint*, abs/2210.11416, 2022. 3

[8] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1–10. IEEE Computer Society, 2018. 8

[9] Matt Deitke, Winson Han, Alvaro Herrasti, Aniruddha Kembhavi, Eric Kolve, Roozbeh Mottaghi, Jordi Salvador, Dustin Schwenk, Eli VanderBilt, Matthew Wallingford, Luca Weihs, Mark Yatskar, and Ali Farhadi. Robothor: An open simulation-to-real embodied AI platform. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 3161–3171. IEEE, 2020. 8

[10] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 13142–13153. IEEE, 2023. 3

[11] Xingjian Diao, Chunhui Zhang, Weiyi Wu, Zhongyu Ouyang, Peijun Qing, Ming Cheng, Soroush Vosoughi, and Jiang Gui. Temporal working memory: Query-guided segment refinement for enhanced multimodal understanding, 2025. 8

[12] Gary N Friedman, Luke Johnson, and Zachary M Williams. Long-term visual memory and its role in learning suppression. *Frontiers in Psychology*, 9:1896, 2018. 3

[13] Jinglong Gao, Xiao Ding, Yiming Cui, Jianbai Zhao, Hepeng Wang, Ting Liu, and Bing Qin. Self-evolving gpt: A lifelong autonomous experiential learner, 2024. 8

[14] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5021–5028. IEEE, 2024. 2

[15] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, and Pheng-Ann Heng. Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following, 2023. 2

[16] Yining Hong, Chunru Lin, Yilun Du, Zhenfang Chen, Joshua B. Tenenbaum, and Chuang Gan. 3d concept learning and reasoning from multi-view images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 9202–9212. IEEE, 2023. 1

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

2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, 2023. 2, 3, 6, 8

[18] Yining Hong, Zishuo Zheng, Peihao Chen, Yian Wang, Junyan Li, and Chuang Gan. Multiply: A multisensory object-centric embodied large language model in 3d world. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 26396–26406. IEEE, 2024. 3

[19] 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 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. 8

[20] 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 *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. 2, 8

[21] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. 2

[22] Mukul Khanna, Ram Ramrakhya, Gunjan Chhablani, Sriram Yenamandra, Théophile Gervet, Matthew Chang, Zsolt Kira, Devendra Singh Chaplot, Dhruv Batra, and Roozbeh Mottaghi. Goat-bench: A benchmark for multi-modal lifelong navigation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 16373–16383. IEEE, 2024. 8

[23] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. AI2-THOR: An Interactive 3D Environment for Visual AI. *arXiv*, 2017. 8

[24] Jacob Krantz, Stefan Lee, Jitendra Malik, Dhruv Batra, and Devendra Singh Chaplot. Instance-specific image goal navigation: Training embodied agents to find object instances, 2022. 8

[25] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Wensi Ai, Benjamin Martinez, Hang Yin, Michael Lingelbach, Minjune Hwang, Ayano Hiranaka, Sujay Garlanka, Arman Aydin, Sharon Lee, Jiankai Sun, Mona Anvari, Manasi Sharma, Dhruva Bansal, Samuel Hunter, Kyu-Young Kim, Alan Lou, Caleb R Matthews, Ivan Villa-Renteria, Jerry Huayang Tang, Claire Tang, Fei Xia, Yunzhu Li, Silvio Savarese, Hyowon Gweon, C. Karen Liu, Jiajun Wu, and Li Fei-Fei. Behavior-1k: A human-centered, embodied ai benchmark with 1,000 everyday activities and realistic simulation, 2024. 3, 8

[26] Manling Li, Shiyu Zhao, Qineng Wang, Kangrui Wang, Yu Zhou, Sanjana Srivastava, Cem Gokmen, Tony Lee, Li Erran Li, Ruohan Zhang, Weiyu Liu, Percy Liang, Li Fei-Fei, Jiayuan Mao, and Jiajun Wu. Embodied agent interface: Benchmarking llms for embodied decision making. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. 8

[27] Xiao Liu, Tianjie Zhang, Yu Gu, Iat Long Iong, Yifan Xu, Xixuan Song, Shudan Zhang, Hanyu Lai, Xinyi Liu, Hanlin Zhao, et al. Visualagentbench: Towards large multimodal models as visual foundation agents. *ArXiv preprint*, abs/2408.06327, 2024. 3

[28] Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul McVay, Oleksandr Maksymets, Sergio Arnaud, Karmesh Yadav, Qiyang Li, Ben Newman, Mohit Sharma, Vincent-Pierre Berges, Shiqi Zhang, Pulkit Agrawal, Yonatan Bisk, Dhruv Batra, Mrinal Kalakrishnan, Franziska Meier, Chris Paxton, Alexander Sax, and Aravind Rajeswaran. Openeqa: Embodied question answering in the era of foundation models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 16488–16498. IEEE, 2024. 8

[29] Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems, 2023. 8

[30] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Florian Gribonval, Rafal Jozefowicz, et al. Pytorch. https://pytorch.org/, 2019. 6, 3

[31] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. 2021. 3, 8

[32] Bokui Shen, Fei Xia, Chengshu Li, Roberto Martín-Martín, Linxi Fan, Guanzhi Wang, Claudia Pérez-D'Arpino, Shyamal Buch, Sanjana Srivastava, Lyne P. Tchapmi, Micael E. Tchapmi, Kent Vainio, Josiah Wong, Li Fei-Fei, and Silvio Savarese. igibson 1.0: a simulation environment for interactive tasks in large realistic scenes. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, page accepted. IEEE, 2021. 8

[33] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 8

[34] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. ALFRED: A benchmark for interpreting grounded instructions for everyday tasks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10737–10746. IEEE, 2020. 8

[35] Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew J. Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 3, 8

[36] Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel X. Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 251–266, 2021. 3, 8

[37] Andrew Szot, Max Schwarzer, Harsh Agrawal, Bogdan Mazoure, Rin Metcalf, Walter Talbott, Natalie Mackraz, R. Devon Hjelm, and Alexander T. Toshev. Large language models as generalizable policies for embodied tasks. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 3, 8

[38] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *ArXiv preprint*, abs/2312.11805, 2023. 3

[39] XLA team. Xla: Optimizing compiler for machine learning. https://www.tensorflow.org/xla, 2017–2025. 6, 3

[40] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *ArXiv preprint*, abs/2408.16061, 2024. 8

[41] Hao Wang, Weining Wang, and Jing Liu. Temporal memory attention for video semantic segmentation. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 2254–2258. IEEE, 2021. 8

[42] Zixuan Wang, Bo Yu, Junzhe Zhao, Wenhao Sun, Sai Hou, Shuai Liang, Xing Hu, Yinhe Han, and Yiming Gan. Karma: Augmenting embodied ai agents with long-and-short term memory systems, 2024. 2

[43] Erik Wijmans, Samyak Datta, Oleksandr Maksymets, Abhishek Das, Georgia Gkioxari, Stefan Lee, Irfan Essa, Devi Parikh, and Dhruv Batra. Embodied question answering in photorealistic environments with point cloud perception. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 6659–6668. Computer Vision Foundation / IEEE, 2019. 8

[44] Quanting Xie, So Yeon Min, Pengliang Ji, Yue Yang, Tianyi Zhang, Aarav Bajaj, Ruslan Salakhutdinov, Matthew Johnson-Roberson, and Yonatan Bisk. Embodied-rag: General non-parametric embodied memory for retrieval and generation, 2024. 2

[45] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *Computer Vision – ECCV 2024*, pages 131–147. Springer Nature Switzerland, 2025. 2, 8

[46] Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-mem: Agentic memory for llm agents. *ArXiv preprint*, abs/2502.12110, 2025. 8

[47] Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in Space: How Multimodal Large Language Models See, Remember and Recall Spaces. *ArXiv preprint*, abs/2412.14171, 2024. 8

[48] Rui Yang, Hanyang Chen, Junyu Zhang, Mark Zhao, Cheng Qian, Kangrui Wang, Qineng Wang, Teja Venkat Koripella, Marziyeh Movahedi, Manling Li, Heng Ji, Huan Zhang, and Tong Zhang. Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents, 2025. 3, 8

[49] Yuncong Yang, Han Yang, Jiachen Zhou, Peihao Chen, Hongxin Zhang, Yilun Du, and Chuang Gan. 3d-mem: 3d scene memory for embodied exploration and reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2, 6, 8, 5

[50] Licheng Yu, Xinlei Chen, Georgia Gkioxari, Mohit Bansal, Tamara L. Berg, and Dhruv Batra. Multi-target embodied question answering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 6309–6318. Computer Vision Foundation / IEEE, 2019. 8

[51] Danyang Zhang, Lu Chen, Situo Zhang, Hongshen Xu, Zihan Zhao, and Kai Yu. Large language model is semi-parametric reinforcement learning agent. *ArXiv preprint*, abs/2306.07929, 2023. 8

[52] Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. A survey on the memory mechanism of large language model based agents, 2024. 8

[53] Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: LLM agents are experiential learners. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 19632–19642. AAAI Press, 2024. 8

[54] Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, Ankur Handa, Ming-Yu Liu, Donglai Xiang, Gordon Wetzstein, and Tsung-Yi Lin. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models, 2025. 2

[55] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla:

3d vision-language-action generative world model. *ArXiv preprint*, abs/2403.09631, 2024. 2

[56] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 8

[57] Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. *ArXiv preprint*, abs/2409.18125, 2024. 2, 5, 6, 3

[58] Guillermo Zlotnik and Aaron Vansintjan. Memory: An extended definition. *Frontiers in Psychology*, 10:2523, 2019. 3

[59] Xueyan Zou, Yuchen Song, Ri-Zhao Qiu, Xuanbin Peng, Jianglong Ye, Sifei Liu, and Xiaolong Wang. M3: 3d-spatial multimodal memory. In *ICLR*, 2025. 8

3DLLM-MEM: Long-Term Spatial-Temporal Memory for Embodied 3D Large Language Model

Supplementary Material

A. Ablation Study

We conduct ablation study as illustrated in 3. Our approach initializes the fused memory using working memory features, aiming to fuse the most relevant memories for the current task. We ablate several design choices for initializing the fusion query, as shown in Table 3 (Appendix A). When using either the most recent episodic memory or learnable zero parameters, performance degrades compared to our proposed method. Interestingly, using the most recent memory outperforms zero initialization in the simple setting but underperforms in the hard setting. One possible explanation is that recent memory initialization encourages fusion with nearby observations, which may be sufficient for simple tasks and leads to faster convergence. In contrast, zero initialization is guided solely by training supervision to learn which memories are most useful. In summary, the ablation results demonstrate that initializing fusion queries with working memory tokens provides the most effective and robust design choice for long-term memory fusion.

B. Environment Construction

To support navigation-centric interaction, the agent requires precise knowledge of two things: the traversable layout of each scene and the exact locations of all movable objects. Following 3D-CLR [16], we build this spatial substrate from HM3D's richly annotated indoor scans. We rely on the semantic surface mesh that accompanies each scene to calculate the room and objects' locations. The mesh scan has a unique (24-bit) hexadecimal color for every surface triangle that provides a semantic table that links each color to a surface label (e.g., floor or ceiling) and a room identifier.

We first derive an axis-aligned bounding box for every room in each HM3D scene. We query the semantic table to retrieve, for every room identifier, the hex colors assigned to its floor and ceiling surfaces. Before processing individual rooms we aggregate the minimum heights of all floor surfaces in the entire scan, producing a global set of candidate floor elevations. For each room we then load the point clouds of its floor and ceiling. When both are available, the vertical bounds are given by the floor's lowest point and the ceiling's highest point. If the floor is missing, the lower bound is set to the highest global floor elevation that still lies below the room's ceiling; if the ceiling is missing, the upper bound defaults to the highest point in the available cloud (i.e., the surface immediately above the floor). Rooms for which both surfaces are absent are discarded because no reliable vertical evidence is present. Horizontal limits are simply the minimum and maximum coordinates of the room's floor and ceiling points, and together these six coordinates constitute the room's axis-aligned bounding box.

Based on the bounding boxes for each room, we calculate the bounding boxes of objects in that room by gathering the corresponding vertices and computing the bounding boxes of the object by taking the minimum and maximum coordinates along the three axes. We merge room-level and object-level bounding boxes by their shared room index. After the aforementioned processing, the dataset comprises 182 distinct 3-D scenes covering a total of 2,602 rooms.

C. Embodied Task Data Examples

In this section, we provide complete embodied task trajectories.

Here is an example of in-the-wild task:

Task: Rearrange desk items by size. The items to be rearranged are: vase, box, and printer.

Trajectory:

"I'm in room 10. I will start exploring the room for the items I need.", "I see a printer(0) on the desk(0). I also see a box(0) on the cabinet(0).", "I need to remember their locations and sizes for comparison later.",
"¡GO TO NEW ROOM¿",
"Now I'm in room 11. I'm exploring this room for the items.", "I don't see a vase in the room, but I see a printer(0) on a rack(1).", "I will remember these objects and their locations.",
"¡GO TO NEW ROOM¿",
"Now I'm in room 6. I'm exploring this room.", "I do not see any of the items I am looking for.",
"¡GO TO NEW ROOM¿",
"I'm in room 8. I am exploring this room.", "I see flower vase(0) on the floor. I will remember the object and its location.",
"¡GO TO NEW ROOM¿",
"I'm in room 7. I am exploring this room.", "I do not see any of the items I am looking for.",
"¡GO TO NEW ROOM¿",
"I'm in room 9. I am exploring this room.", "I do not see any of the items I am looking for.",
"¡GO TO NEW ROOM¿",
"Now I am in room 5. I am exploring this room.", "I do not see any of the items I am looking for.",
"¡GO TO NEW ROOM¿",

Model	Simple				Medium				Hard				Average			
	In-domain		In-the-wild		In-domain		In-the-wild		In-domain		In-the-wild		In-domain		In-the-wild	
	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR	SR	Sub-SR
3DLLM-MEM	45.5	73.4	37.0	65.4	36.8	67.8	31.6	57.4	30.5	46.2	27.8	42.1	37.6	62.5	32.1	55.0
Init with Most Recent Episodic Memory	42.3	69.4	28.6	50.7	32.4	58.6	23.7	45.1	22.6	37.8	15.3	31.4	32.4	55.3	22.5	42.4
Init with Learnable Zero Parameters	41.4	67.2	27.9	50.0	33.0	59.2	23.4	45.8	24.2	40.4	18.6	35.6	32.9	55.6	23.3	43.8

Table 3. Ablation study of query initialization designs in our memory fusion module.

"Now I am in room 2. I am exploring this room.", "I see a box(0) on the armchair(1). I will remember these objects and their locations.",
"¡GO TO NEW ROOM¿",
"I'm in room 1. I do not see any items that are part of the rearrangement.",
"¡GO TO NEW ROOM¿",
"Now I'm in room 12. I will start exploring this room.", "I think I've seen the vase, printer, and box in other rooms, so I'll head back to room 8 to pick up the vase first.",
"¡GO TO ROOM(8)¿",
"I need to pick up flower vase(0) in this room first. I pick up flower vase(0).",
"¡PICK UP flower vase(0) from room(8) in room(8)¿",
"I'm carrying flower vase(0) to room 10 where I saw printer and box.",
"¡GO TO ROOM(10)¿",
"Now I am in room 10. I will put flower vase(0) on the desk(0) for now so I can pick up box(0).",
"¡PUT DOWN flower vase(0) from room(8) on desk(0) in room(10)¿",
"Now I will pick up box(0) from cabinet(0).",
"¡PICK UP box(0) from room(10) in room(10)¿",
"Now I will put box(0) on the desk(0) so I can pick up printer(0).",
"¡PUT DOWN box(0) from room(10) on desk(0) in room(10)¿",
"Now I will pick up printer(0) from beside the desk(0).",
"¡PICK UP printer(0) from room(10) in room(10)¿",
"I need to compare the sizes of the vase, box and printer now to determine the rearranging order.", "I recall that printer(0) in this room had the largest bounding box dimensions based on my earlier observation. I need to put down printer(0).",
"¡PUT DOWN printer(0) from room(10) on desk(0) in room(10)¿",
"Now I will pick up box(0).",
"¡PICK UP box(0) from room(10) in room(10)¿",
"I remember that box(0) from cabinet(0) had medium bounding box dimensions, so I will put it on the desk.",
"¡PUT DOWN box(0) from room(10) on desk(0) in room(10)¿",
"Now I will pick up flower vase(0).",
"¡PICK UP flower vase(0) from room(8) in room(10)¿",
"I remember that flower vase(0) had the smallest bounding box dimensions, so I will put it on the desk.",
"¡PUT DOWN flower vase(0) from room(8) on desk(0) in room(10)¿",
"I will now arrange the objects from small to large: flower vase(0), box(0) and printer(0) on the desk(0).", "Now I pick up the medium object, box(0).",
"¡PICK UP box(0) from room(10) in room(10)¿",
"I will put the medium object on the left of the largest object, printer(0).",
"¡PUT DOWN box(0) from room(10) on desk(0) in room(10)¿",
"Now I pick up the smallest object, flower vase(0).",
"¡PICK UP flower vase(0) from room(8) in room(10)¿",
"I will put the smallest object on the left of the medium object, box(0).",
"¡PUT DOWN flower vase(0) from room(8) on desk(0) in room(10)¿",
"The objects are now arranged from small to large: flower vase(0), box(0) and printer(0) on the desk(0). Task Complete."

Here is an example of in-domain task:

Task: Preparing for Cooking Stir-fried Tomato and Scrambled Eggs. You need to prepare the ingredients, seasonings and cooking tools.

Trajectory: "I'm at room 4. It seems to be living room, I need to prepare ingredients and seasonings and cooking tools and setup them in the kitchen", "There is seasonings on the table.",
"¡PICK UP seasonings(0) from room(4) in room(4)¿",
"No other things found for my task",
"¡GO TO NEW ROOM¿",
"I'm at room 5. It seems to be dining room. I saw an apron on dining table.", "I need the apron but I have seasonings(0) on my hand. Lets find kitchen to put the seasonings first and go back for apron",
"¡GO TO NEW ROOM¿",
"I'm at room 8. It seems to be kitchen with two cabinets. I found tomatoes and eggs on the cabinet(1) that is far away from me and cooking pan in the sink.", "There are two countertops in this room. countertop(1) is closer to stove(0). I should setup on countertop(1).",
"¡PUT DOWN seasonings(0) from room(4) on countertop(1) in room(8)¿",
"¡PICK UP tomatoes(0) from room(8) in room(8)¿",

"¡PUT DOWN tomatoes(0) from room(8) on countertop(1) in room(8)¿",
"¡PICK UP eggs(0) from room(8) in room(8)¿",
"¡PUT DOWN eggs(0) from room(8) on countertop(1) in room(8)¿",
"¡PICK UP cooking pan(0) from room(8) in room(8)¿",
"¡PUT DOWN cooking pan(0) from room(8) on stove(0) in room(8)¿",
"I remember I saw apron in room 5. I need to find it back.",
"¡GO TO ROOM(5)¿",
"¡PICK UP apron(0) from room(5) in room(5)¿",
"¡GO TO ROOM(8)¿",
"¡PUT DOWN apron(0) from room(5) on countertop(1) in room(8)¿",
"The setup in kitchen has been prepared. Task Complete."

D. Implementation Details

We implement our model based on LLaVA-3D [57], modifying it to be compatible with Google TPUs with PyTorch/XLA frameworks [30, 39] . We first expand the model's context window to 8192 tokens to accommodate long-term memory inputs. We then fine-tune our proposed memory module along with the LLM decoder using our training split, initializing from LLaVA-3D's pretrained weights. Training is conducted on 8 Google Cloud TPU v5p cores with a batch size of 256 for 1000 steps, which takes about 1 day to complete. We use Adam optimizer with learning rate of 2e-5 with no weight decay. Additionally, we apply a linear warmup of the learning rate during the initial 3% steps, increasing from 10^{-8} to 10^{-5}, followed by a cosine decay scheduler.

E. Prompts for Gemini

As mentioned in § 2.2, we prompt Gemini to generate the long-term trajectories as illustrated in Table 4, generate the question-answering tasks as shown in Table 5, and generate caption tasks as shown in Table 6. For open-ended QA evaluation, we followed standard LLM-as-judge protocol by prompting Gemini as illustrated in Table 7.

F. Data Validation

F.1. Trajectory Validation

We implement a trajectory simulation pipeline driven by the commands listed in Table 4. For each command, the simulator records the agent's current room and the full set of objects it is holding, then updates the set of objects in each room to reflect pick-up and put-down actions. A pick-up removes the specified object (along with any nested items) from the room the agent occupies and adds it to the agent's hand; a put-down removes the object from the agent's hand and places it into the designated room. The pipeline validates each command based on these criteria: (1) the agent's location; (2) the referenced object and (3) the correctness of pick-up and put-down actions. For location validation, a command is marked as invalid if the agent attempts to pick up an object from a room that does not match its current room, or tries to drop an object into a room other than the one it currently occupies. Additionally, if the agent tries to visit a room that does not exist in the scene, or attempts to enter a new room when all rooms have already been explored, the trajectory is also considered invalid. For object validation, a pick-up command is invalid if the target object does not exist in the current room, and a put-down command is invalid if the agent is not currently holding the specified object. For pick-up and put-down validation, the agent is allowed to hold only one object at a time. A command is considered invalid if the agent attempts to pick up an object while already holding one, or tries to put down an object when its hand is empty. Finally, after all commands have been executed, if the trajectory ends with the agent still holding an object that was never put down, the entire trajectory is marked as invalid.

F.2. Human Validation

As mentioned in §2.3 After automatic trajectory validation, we further conduct human validation, in which four student experts in the field manually inspect each benchmark example. We render multi-view images of the entire scene using the simulator and verify whether the benchmark annotations accurately correspond to the simulated environment as illustrated in Figure 5.

G. Evaluation Setup Details

3D-LLM Similar to the 3D-LLM work [17], we use their direct reconstruction method to extract the 3D features from each scene in our training data. To process our long-term memory data, which requires multi-scene input across each task, we feed each room in the task through the 3D-LLm Q-Former head independently to get separate 32-token dense representation of each room with per-room 3d positional embeddings injected into the features. Then we concatenate the representations before feeding the input into the frozen t5-flanxl [7] backbone like the original work.

The 3D-LLM model also included learned location tokens used to describe certain locations within each room in the scene. To fit 3D-LLM to our task data, we substitute the location tokens with our specific interaction tokens (eg. ¡GO TO ROOM¿ used by all models in our experiments) and train the model to learn the new tokens to stay consistent with our higher level interaction used across our training data. Analysis of the 3D-LLM model evaluation output, indicated the primary struggle for the model was retaining long term memory of semantic observations in the scene, so we prioritized aligning 3D-LLM with the high level long-

System message

You are an AI assistant and task generator for a 3D embodied agent operating in a multi-room environment. The environment provides detailed object instance information, including bounding boxes and IDs. Your goal is to generate a complex task that requires the agent to explore multiple rooms, navigate, and crucially use long-term memory to recall details observed earlier.

Prompt

1. Environment and Object Information

Object Representation: Each object is given with a bounding box in the format: "¡object_name¿(num)": [x_min, y_min, z_min], [x_max, y_max, z_max] Here, (num) indicates the ID, with (0) being the closest to the origin [0,0,0]. IDs reset for each room (e.g., sofa(0) in room 2 and sofa(0) in room 4 if each room has one sofa).

Actions Available: ¡GO TO ROOM(id)¿: Navigate to a room that has already been visited. ¡GO TO NEW ROOM¿: Navigate to a new, unexplored room (and unlock its objects). Do not use this for rooms that have been visited before. ¡PICK UP object_name(id) from room(id) in room(id)¿: Pick up an object that originally belongs to a specific room while in that same room. ¡PUT DOWN object_name(id) from room(id) on object_name(id) in room(id)¿: Place an object (that originally belongs to a room) onto another object (such as a table or floor) in a room.

New Objects: You can add extra objects to diversify the task. Important: Use only object names from the provided new_objects_name_list. If a room already has an object with the same name, the new object should have a new ID (e.g., if lamp(0) exists, the added one should be lamp(1)). These extra objects are only for task design; the agent's trajectory should not mention adding them.

2. Task Design Requirements

Multi-Room Exploration: Design a task that spans several rooms. The room order (given in a Room Order list) should be chosen so that necessary items are distributed across rooms. The agent should explore every room in the specified order.

Long-Term Memory and Implicit Cues: Do not simply list all items as a checklist at the start. Instead: Provide a vague overall goal (e.g., "prepare a meal"). Later in the trajectory, have the agent recall these earlier observations when the need arises. Ensure the agent must remember something seen long ago rather than simply following an explicit list.

Update Memory and make new decision based on your current observations: The agent originall planned to use one object for completing its task, but couldn't find it after exploration of rooms. It has to change to a another similar object to complete its task.

Inventory and Action Constraints: The agent can only hold one item at a time. Never perform consecutive PICK UP or PUT DOWN actions. If the agent holds an item, it must put it down before picking up another. When temporarily storing an object (e.g., on a table), include a "thought" explaining why the object is being set down and later recalled.

3.Reasoning and Object Comparisons: If your task requires choosing a specific object instance (e.g., selecting table(1) because it is bigger than table(0)), compare their bounding boxes and explain your choice in the trajectory.

For clarity, consider these examples: {In-context examples}

———————-

Here is the scene information: {Input scene information}

Table 4. Prompt template for generating task trajectories. {In-context examples} are in-context examples. {Input scene information} are scene, room and object semantics along with their bounding boxes.

term memory representation in our data over low level spatial understanding of the scene.

Our longer task data input also required truncation to fit within the 512 token context length of 3D-LLM's t5-flanxl backbone. We retain the task description and move the question to the beginning of the prompt for the QA data to ensure the model still receives the information necessary to understand its tasks. The longer trajectory of past events is then the only information that gets truncated before fed into the t5 encoder.

For finetuning on our data, we use the hyperparameters provided by 3D-LLM and finetune until model loss stops decreasing. Due to compute limitations, we trained on captioning task for 15 epochs, question-answering task for 20 epochs, and allocated most of the compute time on the embodied task, which we trained on for 75 epochs.

Prompt
You are an AI assistant / task generator in the room. All object instances in this 3D scene are given, along with their bounding boxes and ids." Each object's bounding boxes are represented by a 3D coordinate '¡obj_name¿(num)': [x min, y min, z min],[x max, y max, z max]' with units of meters, and each represents left-bottom corner and the right-top corner coordinate.

You will also receive a trajectory composed of the following tokens and reasoning chains.
¡GO TO ROOM(id)¿: which navigates back to a specific room (id). This can only be done if the agent already go to this room. ¡PICK UP object_name(id) from room(id) in room(id)¿: Pick up an object that originally belongs to a specific room while in that same room. ¡PUT DOWN object_name(id) from room(id) on object_name(id) in room(id)¿: Place an object (that originally belongs to a room) onto another object (such as a table or floor) in a room. ¡GO TO NEW ROOM¿: which navigates to a new room you haven't explored and unlocks objects there.

This trajectory is what the agent have executed over the past. You need to propose several questions and answers that focused on the reasoning abilities of the long-term memory of the agent. These reasoning questions should focus on what have changed temporally or spatially in this agent's memory. It's important that this change challenged the agent's memory. For example the questions should contain object counting, spatial relation, comparison between objects across rooms, long-term multi-room room layout, long-term multi-room object navigation. Remember spatial memory is important, you should design questions that asked about the 3D object spatial relation and layout in the room that need the agent to perform a hard reasoning for the final answer.

For clarity, consider these examples: {In-context examples}
———————-
Here is the scene information: {Input scene information}
Here is the agent's trajectory: {Input agent's trajectory}

Table 5. Prompt template for generate QA data. {In-context examples} are in-context examples. {Input scene information} are scene, room and object semantics along with their bounding boxes. {Input agent's trajectory} is the 3D agent's explored trajectories and action chains.

3D-Mem We benchmark 3D-Mem [49] on the question-answering and captioning splits of 3DMEM-BENCH. 3D-Mem is a snapshot-based 3D memory architecture originally developed for embodied exploration and reasoning; it keeps two complementary stores—memory snapshots, a compact set of multi-view RGB-D frames with per-object bounding boxes summarizing the areas the agent has inspected, and frontier snapshots, boundary views that suggest where useful new information may be found next. In its native setting the agent navigates an unfamiliar scene by selecting the frontier view most likely to advance its task and then answers visual questions using the most relevant memory snapshots. Because our evaluation focuses on post-exploration reasoning rather than active exploration, we disable the frontier component and retain only the memory snapshots. For these two tasks, the system will capture memory snapshots in each room from the room center, and finish the QA and captioning base on the memory snapshots of all the explored rooms.

H. Qualitative Examples

We provide qualitative examples as shown in Figure 6. It demonstrates that 3DLLM-MEM can maintain a long-term memory and perform complex tasks in the embodied environments. More examples can be found in the **supplementary materials**.

Prompt

You are provided with a scene description containing multiple rooms. Each room includes a list of objects along with their positions in the room, represented by bounding boxes. Each object's bounding box is defined by a 3D coordinate in the format: ¡object_name¿(num): [x min, y min, z min],[x max, y max, z max] with units in meters (defining the left-bottom and right-top corners). Your task is to generate an object caption for each room in the form of a coherent, descriptive paragraph that conveys the 3D spatial arrangement and relative positions of all objects within that room. Then, you will receive the object descriptions and caption for the current 3D room you are in. You will also be provided with the previous rooms' captions as well. Your task is to generate new captions covering the summarization of the common features across all rooms based on your current room and important difference based on your current room. The reasons of generating the new caption is to help the agent to remind of what are in previous rooms memories can help the agent in this current room. The past objects and observations should be related to current room by examining the summarization of common things and differences.

For clarity, consider these examples: {In-context examples}

———————-

Here is the scene information: {Input scene information}

Here is current room you are in and previous rooms you went: {Input agent's location}

Table 6. Prompt template for generate QA data. {In-context examples} are in-context examples. {Input scene information} are scene, room and object semantics along with their bounding boxes. {Input agent's location} is the location for current room in the scene and the past explored rooms.

System message

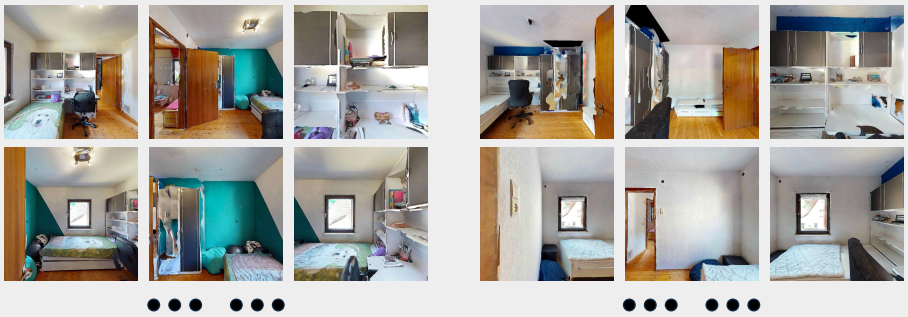
Please act as an impartial judge and evaluate the quality of the response provided by an AI assistant to the user question. Your evaluation should consider correctness and helpfulness. You will be given a reference answer and the assistant's answer. You evaluation should focus on the assistant's answer to the second question. Begin your evaluation by comparing the assistant's answer with the reference answer. Identify and correct any mistakes. Be as objective as possible. After providing your explanation, you must rate the response on a scale of 1 to 10 by strictly following this format: "[[rating]]", for example: "Rating: [[5]]".

Prompt

¡—The Start of Reference Answer—¿
User:
question_1
Reference answer:
ref_answer_1
User:
question_2
Reference answer:
ref_answer_2
¡—The End of Reference Answer—¿
¡—The Start of Assistant A's Conversation with User—¿
User:
question_1
Assistant A:
answer_1
User:
question_2
Assistant A:
answer_2
¡—The End of Assistant A's Conversation with User—¿

Table 7. Prompt template for open-ended QA evaluation following standard LLM-as-judge protocol.

Room 8 **Room 9**



Object Captioning

Room 8 contains distinct sleeping, working, and lounging zones. **Bed**(0) extends along one side, with **pillow**(0) near its head and **cabinet**(1) beside it, which supports **decoration**(0) and is flanked by **book**(0) and **book**(1). Near the foot of bed(0), **yoga mat**(0), **bag**(0), and **box**(0) form a small activity area. At the far end, **wardrobe**(0) stands near wall(4), next to **beanbag chair**(0) and **exercise ball**(0), with **clothes**(0) draped nearby, creating a relaxed corner. The office area, positioned across from bed(0), includes **desk**(0), **office chair**(0), and **cabinet**(0), with **trashcan**(0) nearby and **ceiling lamp**(0) overhead. **Window**(0) on the **parapet**(0) brings in natural light. The arrangement ensures functional use of space while maintaining clear separation between different activities.

Question Answering

Object Counting

- Q: How many chairs are there in Room 8?
- A: There are 2 chairs in total: one office chair(0) and one beanbag chair(0).

Spatial Relation

- Q: If you are sitting on the office chair(0) working at the desk(0) in Room 8, where is the window(0) relative to you?
- A: The window(0) is behind you to the left.

Comparative QA

- Q: Which room has more books, Room 8 or Room 9?
- A: Room 8 has more books. It contains two books, book(0) and book(1), both placed on top of the cabinet(0) near the desk(0).

Object Navigation

- Q: If you are sitting on the beanbag chair(0) in Room 8, how do you get to the books?
- A: You need to stand up, turn left, and walk across the room toward the cabinet(0) near the desk(0), where the books are placed.

Room Layout

- Q: In Room 8, what is the relative layout of the bed(0), desk(0), and office chair(0)? Which one is closest to the window(0)?
- A: The bed(0) is placed along the side wall near the window(0), the desk(0) is located in the far corner of the room along wall(5), and the office chair(0) is positioned directly in front of the desk(0). Among these, the bed(0) is closest to the window(0), which is embedded in the wall beside it.

Figure 5. Example of human annotators manually check the data quality on QA and captioning tasks through multiple rendered multi-view images from each room.

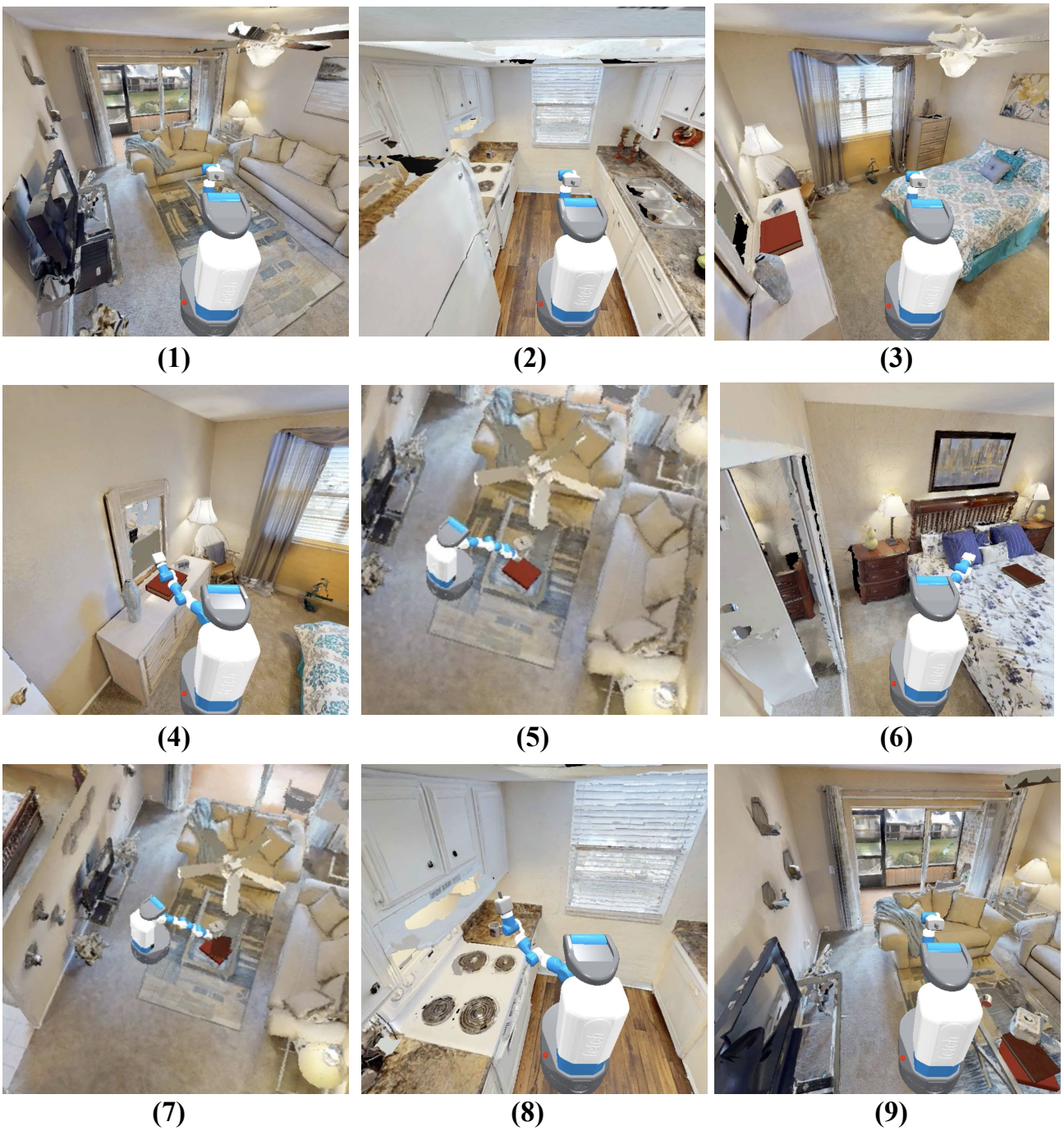


Figure 6. Qualitative example of 3DLLM-MEM. The task instruction is: *Prepare a cozy reading nook in the living room with two books and a teacup*. In images (1) and (2), the agent explores the environment randomly, forming an initial memory of the scene. After receiving the task instruction, it recalls its memory and navigates to the bedroom to pick up a book from the cabinet, as shown in images (3) and (4). The agent then returns to the living room and places the book on the table in front of the sofa (image 5). Unable to recall any additional books, the agent resumes exploration and finds a second book on the bed, which it picks up (image 6) and stacks on top of the first book (image 7). Finally, the agent recalls seeing a teacup in the kitchen, navigates to retrieve it (image 8), and places it on the table in the living room (image 9). The task is successfully completed.