# Predict Before You Explore: Predictive Planning with Specialized Memory for Embodied Question Answering

## Supplementary Material

In the supplementary material, we provide 1) extended results in action consistency and generalization of Pred-EQA (Sec. 1), 2) details of variants in ablation studies in our main paper (Sec. 2), 3) Implementation details in VLM deploying, frontier-based exploration, and agent prompts (Sec. 3), 4) extended visualizations of trajectories (Sec. 4), 5) dataset details (Sec. 5) and 6) metric details (Sec. 6).

| Pred. Plan | Spec. Mem. | mDCA↓ | FRR↓ | PRR(<0.5m)↓ | PRR(<1.0m)↓ |
|---|---|---|---|---|---|
| Qwen3-VL 8B | | 44.9° | 12.8±10.2 | 7.6±15.5 | 11.9±20.4 |
| ✓ | | 32.7° | 7.1±8.4 | 4.3±9.8 | 5.9±14.4 |
| | ✓ | 37.9° | 9.3±9.0 | 6.4±16.9 | 8.5±21.2 |
| ✓ | ✓ | **28.1° -16.8°** | **4.6±5.1 -8.2** | **1.4±3.7 -6.2** | **3.0±8.4 -8.0** |

Table 1. **Action Consistency Results.** Lower values indicate more consistent exploration behavior.

| Method | VLM Usage | LLM-Match ↑ | LLM-SPL ↑ |
|---|---|---|---|
| Baseline | Qwen2.5-VL 3B | 32.1 | 21.7 |
| Pred-EQA | Qwen2.5-VL 3B | 38.6+6.5 | 32.2+10.5 |
| Baseline | Qwen2.5-VL 7B | 40.8 | 24.6 |
| Pred-EQA | Qwen2.5-VL 7B | 46.2+5.4 | 37.8+13.2 |
| Baseline | Qwen3-VL 4B | 43.9 | 37.7 |
| Pred-EQA | Qwen3-VL 4B | 50.8+6.9 | 46.2+8.5 |
| Baseline | Qwen3-VL 8B | 45.7 | 39.5 |
| Pred-EQA | Qwen3-VL 8B | 53.3+7.6 | 48.5+9.0 |

Table 2. **Generalization abilities on different architectures and sizes on the A-EQA dataset.**

## 1. Extended Ablation Experiments

### 1.1. Action Consistency Analysis

To quantitatively evaluate the action consistency of Pred-EQA, we design three complementary metrics that capture different aspects of coherent exploration behavior:

**Mean Direction Change Angle (mDCA).** This metric measures the average angle between consecutive movement vectors, reflecting the smoothness of the agent's trajectory. A lower mDCA indicates more consistent directional choices across steps:

$$\text{mDCA} = \frac{1}{N-1}\sum_{t=1}^{N-1}\arccos\left(\frac{\vec{v}_t \cdot \vec{v}_{t+1}}{|\vec{v}_t||\vec{v}_{t+1}|}\right), \quad (1)$$

where $\vec{v}_t$ denotes the movement vector at step $t$.

**Frontier Reselection Rate (FRR).** This metric quantifies the rate at which the agent re-selects a previously chosen frontier, indicating indecisiveness or lack of coherent planning:

$$\text{FRR} = \frac{1}{N}\sum_{t=1}^{N}\mathbb{I}[f_t \in \{f_1, \ldots, f_{t-1}\}], \quad (2)$$

where $f_t$ is the frontier selected at step $t$ and $\mathbb{I}[\cdot]$ is the indicator function.

**Position Revisit Rate (PRR).** This metric measures the proportion of steps where the agent moves within a threshold distance of any previously visited location:

$$\text{PRR}(d) = \frac{1}{N}\sum_{t=1}^{N}\mathbb{I}\left[\min_{i<t}\|p_t - p_i\| < d\right], \quad (3)$$

where $p_t$ is the agent's position at step $t$ and $d$ is the distance threshold. We report both 0.5m and 1.0m thresholds.

As shown in Table 1, Pred-EQA demonstrates significantly smoother and more consistent exploration compared to the baseline. The full Pred-EQA achieves a 16.8° reduction in mDCA, indicating more stable directional choices. The FRR drops from 12.8% to 4.6%, showing that our predictive planning effectively prevents the agent from oscillating between frontiers. Similarly, the PRR metrics show substantial reductions, confirming that Pred-EQA maintains coherent trajectories with minimal unnecessary revisiting of explored regions.

### 1.2. Generalization across Architectures

Table 2 evaluates Pred-EQA on a range of VLM backbones, spanning two families (Qwen2.5-VL and Qwen3-VL) and 4 different model sizes. Across all tested VLM backbones (Qwen2.5-VL and Qwen3-VL), Pred-EQA consistently improves over the baseline, demonstrating that our predictive planning and specialized memory are architecture-agnostic and have a great generalization ability.

### 1.3. Integration with Prior Works

To verify the transferability of our proposed components, we integrate our predictive planning and specialized memory into existing EQA methods, specifically 3D-Mem [6] and Fine-EQA [1], using Qwen3-VL 8B as the backbone.

**Integration with 3D-Mem.** We integrate our components into 3D-Mem by either replacing or adapting-then-adding them. As shown in Table 3, replacing the object-centric memory with our specialized memory yields +3.6% LLM-Match and +5.5% LLM-SPL improvements. This confirms that our memory design directly reasons semantic

| Method Variants (Qwen3-VL 8B) | LLM-Match ↑ | LLM-SPL ↑ |
|---|---|---|
| 3D-Mem (Obj. Mem.(GPT-4o) + E&A) | 46.4 | 32.9 |
| 3D-Mem (Obj. Mem. + E&A) | 44.2 | 28.2 |
| 3D-Mem (Spec. Mem.* + E&A) | 47.8 +3.6 | 33.7 +5.5 |
| 3D-Mem (Obj. Mem. + H.L. Planner† + E&A) | 50.4 +6.2 | 37.8 +9.6 |

Table 3. **Integration with 3D-Mem.** Obj. Mem.: object-centric memory, E&A: single agent for both exploration and answering. * We use Recorder + Manager* to adapt in 3D-Mem. † We use image input for high-level planner and add prompts in E&A to support prediction-correction loop.

| Method Variants (Qwen3-VL 8B) | C ↑ | C* ↑ | $E_{path}$ ↑ | $d_T$ ↓ |
|---|---|---|---|---|
| Fine-EQA (FBE + GOE) | 42.0 | 63.3 | 25.2 | 6.04 |
| Fine-EQA (H.L. Planner† + GOE) | 50.5 +7.5 | 67.7 +4.4 | 32.5 +7.3 | 5.58 -0.46 |

Table 4. **Integration with Fine-EQA.** FBE: Frontier-based Exploration, GOE: Goal-oriented Exploration. † We use image input and add Executor to support prediction-correction loop.

relevance without fragile object parsing, yielding more robust and compact memory. Adding our high-level planner with prediction-correction loop further boosts performance to +6.2% LLM-Match and +9.6% LLM-SPL, demonstrating that predictive planning provides transferable benefits to reactive baselines.

**Integration with Fine-EQA.** We adapt our predictive planning into Fine-EQA by incorporating the prediction-correction loop. As shown in Table 4, this integration yields substantial improvements: +7.5% in C, +4.4% in C*, and +7.3% in $E_{path}$. The navigation metric $d_T$ also improves by 0.46m, indicating better spatial awareness. These results confirm that our predictive planning architecture can be effectively transferred to existing reactive EQA agents, enhancing both correctness and path efficiency.

## 2. Details of Variations in Ablation Studies

In this section, we provide detailed descriptions of all model variants used in the ablation studies in Table 4 and Table 5 of the main paper.

### 2.1. Variants Used in Table 4: Planning Variants

- **Executor***. We modify the prompting of the reactive Executor to explicitly encourage long-term reasoning at every step. However, despite such prompting, the controller remains fundamentally reactive and lacks explicit multi-step prediction capability.
- **TODO List Plan**†. This variant uses a high-level planner to directly output a TODO list of subgoals, paired with the default Executor in Pred-EQA for execution. The subgoal list is less informative and less changed once generated, lacking mechanisms for revising outdated decisions in partially observed environments.

### 2.2. Variants Used in Table 5: Memory Variants

For these variants, when memory management is enabled, snapshots are selectively stored or pruned according to the described mechanism. When memory management is disabled (*Baseline*), we simply maintain the most recent 15 snapshots.

- **3D-Mem** [6]. We adopt the snapshot management mechanism from 3D-Mem, which uses an object detector to extract all objects in each snapshot. Snapshot selection is performed based on 1) object–question relevance and 2) maximizing non-overlapping spatial coverage.
- **Manager**. This variant uses only the Manager module from Pred-EQA for snapshot selection, retaining question-relevant snapshots without structural textual memories.
- **Manager***. Here we prompt the Manager to simultaneously focus on two objectives: 1) storing question-relevant evidence, and 2) maintaining a broader record of the environment for exploration history. As a result, it keeps both semantic cues and general environmental context using heavy visual snapshots.

## 3. Implementation Details

### 3.1. VLM Deploying Details

To enable efficient multi-agent embodied reasoning, we deploy open-source vision-language models (VLMs) using `vLLM` [2] with a custom multi-modal extension. A key feature of our deployment is the ability to share visual encoding results across concurrent agents, eliminating redundant computation and enabling high-throughput parallel exploration.

**Shared Multi-modal Cache (MM Cache).** All agents operating within the same environment instance share a single vLLM engine and a global MM Cache. This cache stores precomputed visual embeddings indexed by a canonical Base64 key. The workflow is:

- **Cache insertion**: when the first agent receives a request including a new visual observation, *i.e.*, the snapshots not seen in the previous steps, the system 1) processes it with the vision encoder, 2) stores the resulting embedding in the MM Cache, and 3) forwards the embedding to the required agents for decoding.
- **Cache hit**: If any subsequent agent or the same agent at a later step observes an identical snapshot, the lookup yields a cache hit. The system bypasses the vision encoder entirely and directly reuses the cached embedding.

To ensure consistent cache indexing, all new images undergo deterministic preprocessing before inference. This guarantees that perceptually identical observations map to the same Base64 key.

**Prefix Cache Sharing.** Besides multi-modal caching, we also benefit from vLLM's Prefix Cache, as many agents

| Method | VLM/LLM Calls/Step | Extra | Latency/Step |
|---|---|---|---|
| ***Original Settings*** | | | |
| 3D-Mem [6] | 1/1 | Obj. Det. | 18.2s |
| ToolEQA [7] | 2/0 | Tool Call | 31.4s |
| ***Our customized vLLM Settings (Qwen3-VL 8B)*** | | | |
| Baseline | 2/0 | N/A | **6.9s** |
| 3D-Mem [6] | 1/1 | Obj. Det. | 8.4s |
| **Pred-EQA** | 3/2 | N/A | 9.8s |

Table 5. **End-to-End Latency Comparison**. In the original setting, we presented the reporting results of ToolEQA. The results of 3D-Mem are reproduced by us using GPT-4o API.

share identical instruction prefixes and environmental descriptions. These shared prefixes frequently trigger prefix-level cache hits and further reduce decoding overhead.

In practice, our multi-agent system achieves **87.3% MM Cache hit rate and 42.8% Prefix Cache hit rate** on average. These high hit rates significantly suppress redundant computation from both the vision encoder and the language model, keeping step-wise latency low even when multiple VLM calls are required.

We evaluate the end-to-end latency of our Pred-EQA against our simple multi-agent baseline and 3D-Mem [6]. All performance metrics are measured on Qwen3-VL-8B using our customized `vLLM` extension. Specifically, the Multi-agent Baseline comprises a Planner and an Answerer (2 VLM calls), 3D-Mem [6] includes an object detection module, a text-only object filtering module, a Plan&Answer module (1 VLM call, 1 LLM call, 1 object detection step), Our Pred-EQA consists of 2 text-only agents (Recorder, High-level Planner) and 3 multimodal agents (Executer, Manager, Answerer). We also present the latency of two existing methods, 3D-Mem and ToolEQA, in their original settings for comparison.

Results in Table 5 show that under the original settings, both existing systems exhibit high latency. Pred-EQA achieves 9.8s, only 42% slower than the baseline despite requiring 2.5x more model calls, better than those existing methods. This modest overhead directly benefits from our high cache hit rates, which drastically reduce redundant visual encoding and repeated prefix decoding.

### 3.2. Frontier-Based Exploration Details

Our frontier-based exploration follows Explore-EQA [5] and 3D-Mem [6]. We maintain a 3D voxel occupancy map $M$ with $0.1m$ resolution covering the full room volume. Each depth frame, combined with the agent's camera pose, updates $M$ by carving free space from an initially fully occupied grid. The navigable area is defined as the slice of unoccupied voxels at a height of $0.4m$. Within this slice, regions within $1.7m$ of the agent trajectory are marked as explored, while all other unoccupied voxels are unexplored,

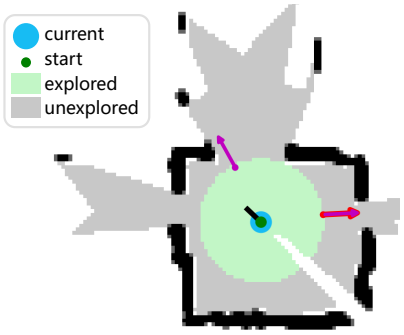


Figure 1. **Example of Frontier-Based Exploration.**

as shown in Fig. 1.

Frontiers are extracted from the boundaries between explored and unexplored regions. We apply Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to cluster connected unexplored pixels, forming frontier candidates $r$. Each frontier is represented as $F = \langle r, p, I^{\text{obs}} \rangle$, where $p$ is the closest navigable boundary point between $r$ and the explored region, and $I^{\text{obs}}$ is an image observation captured after the frontier is updated. Clusters smaller than 20 pixels are discarded. A frontier is refreshed when its region changes significantly, *i.e.*, IoU $< 0.95$. To avoid overly broad frontier regions, if $r$ spans more than $150^\circ$ in the agent's field of view, we split it into two sub-frontiers via K-Means, enabling more precise direction choices.

When interacting with the VLM agents, only image observations are included in the context. If the VLM agent selects a frontier $F$, he agent then directly navigates towards the location of the frontier $p$ from its current pose. The agent moves **a maximum distance of 1.0$m$ in each step under a collision-free assumption**.

### 3.3. Agents Prompt Details

We present the prompt used for the agent context, *i.e*, Executer (Fig. 3), High-level Planner (Fig. 4 and 5), Recorder (Fig. 6), Manager (Fig. 7), and Answerer (Fig. 8). Prompt for variations in ablation studies in our main paper can be found in the code.

## 4. Extended Visualization Analysis

Fig. 2 illustrates how Pred-EQA conducts coherent, prediction-driven exploration compared to reactive baselines. In the first example, answering "What is hanging from the oven handle?" requires discovering the oven, which is not visible in the initial frames. The baseline agent oscillates locally due to its short-sighted action selection, whereas Pred-EQA predicts the doorway to living room as a likely evidence branch and follows a stable exploratory route. As the agent approaches the oven, the visual evidence memory captures the emerging towel, leading to a

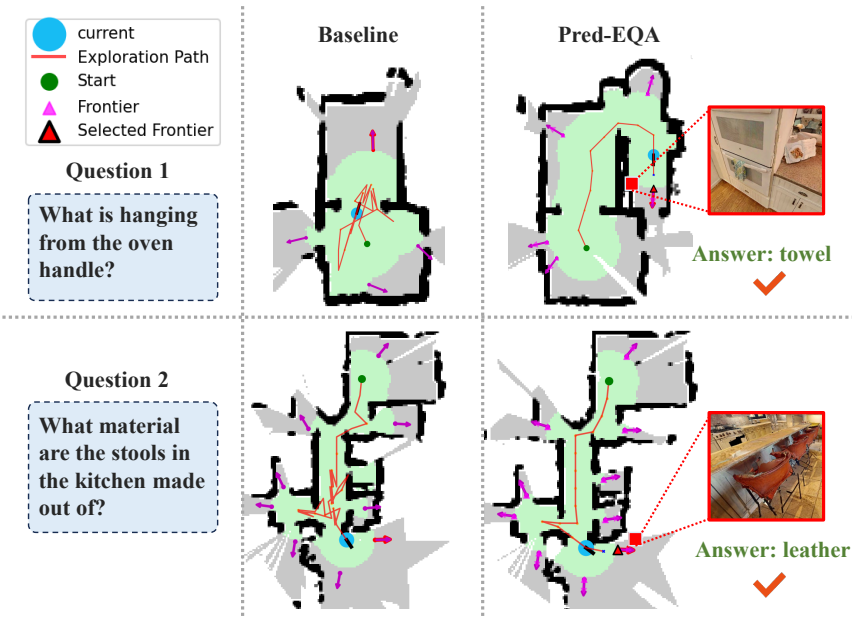


Figure 2. **Extended Visualization of Baseline and Pred-EQA.**

correct answer. In the second example, identifying the stool material requires traversing **a long, visually uninformative corridor** before reaching the dining area. This segment provides almost no immediate reward signal for reactive planners, causing baselines to divert into nearby but irrelevant rooms. In contrast, Pred-EQA maintains forward motion by relying on its high-level prediction that the target evidence lies beyond the corridor, and its semantic prior anchors this long-horizon intent. Once the agent reaches the dining region, concise visual evidence suffices to recognize the leather stools. Overall, these cases demonstrate how the prediction-correction loop sustains coherent long-range exploration and mitigates the short-sightedness of greedy policies under partial observability.

## 5. Benchmark Details

Our experiments are based on two embodied question answering (EQA) benchmarks: A-EQA and EXPRESS-Bench.

- A-EQA contains 557 questions collected from 63 scenes in the Habitat-Matterport 3D (HM3D) dataset [4]. However, only a subset, *i.e.*, 184 questions, is publicly released as part of the OpenEQA benchmark [3]. Due to this limitation, our evaluation is conducted on this open subset. The questions are open-vocabulary and open-ended, requiring agents to perform diverse daily tasks including *Object Recognition*, *Attribute Recognition*, *Object State Recognition*, *Object Localization*, *Spatial Reasoning*, *Functional Reasoning*, and *World Knowledge*.
- EXPRESS-Bench [1] includes 777 exploration trajectories and 2,044 question–trajectory pairs in HM3D dataset. Its questions fall into seven categories: *State*, *Knowledge*, *Location*, *Attribute*, *Counting*, *Existence*, and *Object*. It ensures faithful assessment by measuring the alignment between answer grounding and exploration reliability.

For each question, an agent is initialized at a specific location and must actively explore the environment to gather the necessary information before answering.

## 6. Metric Details

For A-EQA, performance is quantified using two metrics derived via GPT-4: the LLM-Match score, which measures answer accuracy, and LLM-SPL, which further incorporates exploration efficiency. Their formulations are as follows:

$$\text{LLM-Match} = \frac{1}{N}\sum_{i=1}^{N}\frac{\sigma_i - 1}{4} \times 100\%, \tag{4}$$

$$\text{LLM-SPL} = \frac{1}{N}\sum_{i=1}^{N}\frac{\sigma_i - 1}{4} \times \frac{l_i}{\max(p_i, l_i)} \times 100\%, \tag{5}$$

where $\sigma_i \in \{1, 2, 3, 4, 5\}$ is a discrete accuracy score assigned by GPT-4 based on the OpenEQA prompt [3], $l_i$ denotes the minimal path length along the ground-truth trajectory sufficient to answer the question, and $p_i$ is the actual path length traversed by the agent.

For Express-Bench, we adopt the evaluation protocol using GPT-4o-mini, featuring four complementary metrics: $C$, $C^*$, $E_{\text{path}}$, and $d_T$. Here, correctness and grounding are decoupled via two human-inspired scoring components: $\sigma_i \in \{1, \ldots, 5\}$: a VLM-assigned answer correctness score, where higher values indicate responses closer to the ground truth, and $\delta_i \in \{0, 0.5, 1\}$: a grounding score reflecting visual-semantic consistency. Specifically, $\delta_i = 1$

indicates that the agent's observations are relevant and its environmental description is accurate, $\delta_i = 0.5$ denotes relevant observations but inaccurate descriptions, and $\delta_i = 0$ signifies irrelevant or hallucinated outputs, regardless of answer correctness. Based on these, we define:

$$C = \frac{1}{N} \sum_{i=1}^{N} \frac{\sigma_i \cdot \delta_i}{5} \times 100\%, \tag{6}$$

$$E_{\text{path}} = \frac{1}{N} \sum_{i=1}^{N} \frac{\sigma_i \cdot \delta_i}{5} \cdot \frac{l_i}{\max(p_i, l_i)} \times 100\%, \tag{7}$$

where $C$ captures grounded answer correctness, and $E_{\text{path}}$ further weights correctness by navigation efficiency.

To isolate the effect of answer grounding, it reports $C^*$, which assumes perfect grounding, *i.e.*, $\delta_i = 1$, for all samples:

$$C^* = \frac{1}{N} \sum_{i=1}^{N} \frac{\sigma_i}{5} \times 100\%. \tag{8}$$

Finally, it includes a navigation-oriented metric to analyze spatial reasoning capability:

$$d_T = \frac{1}{N} \sum_{i=1}^{N} \text{dis}_g(P_E^i, P_T^i), \tag{9}$$

where $\text{dis}_g(\cdot, \cdot)$ denotes geodesic distance, $P_E^i$ is the agent's final position, and $P_T^i$ is the target location. A lower $d_T$ suggests better spatial awareness, even if not strictly necessary for task success.

## References


[1] Kaixuan Jiang, Yang Liu, Weixing Chen, Jingzhou Luo, Ziliang Chen, Ling Pan, Guanbin Li, and Liang Lin. Beyond the destination: A novel benchmark for exploration-aware embodied question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9091–9101, 2025. 1, 4

[2] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023. 2

[3] Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, et al. Openeqa: Embodied question answering in the era of foundation models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16488–16498, 2024. 4

[4] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*. 4

[5] Allen Z Ren, Jaden Clark, Anushri Dixit, Masha Itkina, Anirudha Majumdar, and Dorsa Sadigh. Explore until confident: Efficient exploration for embodied question answering. *arXiv preprint arXiv:2403.15941*, 2024. 3

[6] 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 Computer Vision and Pattern Recognition Conference*, pages 17294–17303, 2025. 1, 2, 3

[7] Mingliang Zhai, Hansheng Liang, Xiaomeng Fan, Zhi Gao, Chuanhao Li, Che Sun, Xu Bin, Yuwei Wu, and Yunde Jia. Multi-step reasoning for embodied question answering via tool augmentation. *arXiv preprint arXiv:2510.20310*, 2025. 3

Task: You are an indoor agent that needs to PHYSICALLY NAVIGATE through sequential frontier selections to finally find information needed for answering the question.

**Instructions**:
1. Analyze the question to infer what kind of visual clue would help answer it (e.g., a specific object, a room type).
2. Review past textual memories and observations to identify: Which areas have been explored, What partial clues have been seen, Where uncertainty remains.
3. Given question needs and current exploration progress, choose a frontier based on the following Core Principles and constraints:
**Principle 1** (Maximize Information per Step): Choose the frontier that most reduces uncertainty in High-Level Plan - e.g., one that reveals a new object, clarifies partially observed or ambiguous information, or confirms a room's function.
**Principle 2** (Common Sense as Guide, Not Rule): Use typical room-object associations (e.g., "refrigerator" in kitchen, "bed" in bedroom) to rank frontiers only when no visual clues exist, but never ignore observed clues in favor of assumptions.
**Principle 3** (Plan-Driven Movement): Prioritize following the ongoing plan. If you think the plan needs to be modified, please explain the reasons.
**Constraint 1**: If you find that you are still in an irrelevant area, you can only choose a frontier and continue walking in order to reach the relevant area.
**Constraint 2**: You can only access unvisited areas by selecting a frontier step-by-step.
**Constraint 3**: Keep selecting a frontier for moving until you find conclusive evidence enough to answer the question. Note that the objects mentioned in all questions are definitely available, unless there are any questions about their existence.

**Target Question**: [QUESTION]

**High-Level Plan**:
[ ] Pending task description...
[-] In progress task description... <!– status; rationale –>
[x] Completed task description... <!– status; rationale –>

**Previous Environmental/Historical Trajectory and Agent Analysis Information**:
- Step 1: ...

**Previously Observed Clues**:
Snapshot 0: [IMAGE]
...

**Available Exploration Directions**:
Frontier 0: [IMAGE]
...
Available Frontier indices: 0,...

**Output Format**:
1. Describe the current environment and the incremental historical trajectory.
2. Think step by step and explain your reasoning clearly.
3. Provide your final answer in the exact format: "Next Step: Frontier i" or "Stop Exploration", where i is the index of the frontier you choose.

Figure 3. **Prompt for Executer**.

Task: You are a HIGH-LEVEL EXPLORATION PLANNER AGENT responsible for devising a long-term navigation and search plan to answer the user's question. Based on the question, you must break down the goal into a sequence of high-level tasks (e.g., go to a room, find an object, observe an attribute) and output them as an ordered to-do list. This plan will guide the low-level agents in subsequent steps.

**Instructions**:
1. Analyze the user's question and identify its type (object recognition, attribute recognition, spatial relationship, object state, functional reasoning, world knowledge, or object localization).
2. Decompose the question into subgoals. For example:
- Object recognition: Determine which object to find and where it is likely located.
- Attribute recognition: Identify the object and which attribute to check.
- Spatial understanding: Decide which locations or objects need exploration to understand their spatial arrangement.
- Object state recognition: Determine which object's state to verify and how to observe it.
- Functional reasoning: Identify relevant objects that demonstrate the function in question.
- World knowledge: Use typical associations (e.g., kitchen contains a fridge) to infer where to search.
- Object localization: Plan a search sequence for locating the object in different rooms.
3. For each subgoal, create a clear task (e.g., "Go to the kitchen", "Find the refrigerator", "Check the microwave's door status").
4. Create Parallel Prediction-Based Branches for the immediate next step. For the most immediate unresolved navigation or search task (e.g., figuring out how to get to the kitchen), do not output a single generic task. Instead, generate multiple parallel prediction-based exploration branches. Formulate these as testable hypotheses based on current observations and world knowledge (e.g., instead of [ ] Find the kitchen, create [ ] Explore the frontier leading to the hallway since it may lead to the kitchen and [ ] Explore the frontier leading to the living area since it may lead to the kitchen).
5. Combine these immediate predictive branches and the remaining downstream high-level tasks into a single, cohesive, ordered to-do list. Place the parallel predictive branches at the very top as the active starting point, followed by the subsequent tasks.
6. Use the updateable checklist format for output. Mark tasks as [ ] pending, [-] in progress, or [x] completed based on what has been done so far. When agents investigate and eliminate predictive branches, mark the incorrect or dead-end branches as completed [x] with a brief inline explanation. Add new tasks immediately when they become apparent. Do not remove unfinished tasks unless they are truly irrelevant to the goal.

**Core Principles**:
- Before updating, always confirm which todos have been completed or invalidated since the last update.
- You may update multiple statuses in a single update (e.g., mark the previous as completed, invalidate obsolete tasks, and mark the new one as in progress).
- Dynamic Replanning: Because the environment is partially observable, new observations may completely invalidate your previous assumptions or downstream plans. If this happens, you MUST actively overhaul the plan. You are allowed to restructure, replace, or pivot away from previously planned unfinished tasks to align with the newly discovered ground truth.
- When a prediction-based branch proves incorrect (a dead-end), OR when a downstream task becomes obsolete due to a plan overhaul, mark it as [x] AND append a brief inline comment explaining why it failed or was discarded (e.g., [x] Search the kitchen ).
- Once ONE predictive branch successfully locates the target, immediately mark all other parallel predictive branches for that same goal as [x] with an explanation.
- When a completely new actionable path is discovered that pivots the entire strategy, add the new tasks immediately into the sequence and mark the old, now-irrelevant tasks as [x] with a brief explanation of the pivot.
- For regular tasks that remain relevant to the current valid strategy, only mark them as completed [x] when fully accomplished successfully (no partials, no unresolved dependencies).

**Content Constraints**:
STRICTLY NEVER mention ANY snapshot or frontier identifiers (e.g., "Snapshot 2", "Frontier 0")-these labels are step-specific and will cause confusion in later steps when the current image is no longer available.
AVOID relative directional references tied to transient views (e.g., "left of the snapshot", "right of the frontier"). Instead, describe spatial relationships using observable objects (e.g., "the chair is next to the table").

Figure 4. **Prompt for High-level Planner** (Part 1, Total 2).

**Target Question**: [QUESTION]

**High-Level Plan from Last Step**:
[ ] Pending task description...
[-] In progress task description... <!– status; rationale –>
[x] Completed task description... <!– status; rationale –>

**Previous Environmental/Historical Trajectory and Agent Analysis Information**:
- Step 1: ...

**Output Format**:
1. First, think step by step and explain your reasoning clearly.
2. Output updated branch-wise checklist in the following XML checklist format:
<update-checklist>
<branches>
[ ] pending task description
[-] in progress task description
[x] Completed task description
</branches>
</update-checklist>

Figure 5. **Prompt for High-level Planner** (Part 2, Total 2).

Task: You are a concise assistant that summarizes a step in an exploration process for long-term textual memory.

**Instructions**:
Summarize what is NEW and DIFFERENT in this step compared to previous steps. Focus on:
1. NEW areas entered, NEW objects discovered, or NEW spatial relationships observed for the FIRST TIME in this step.
2. The specific navigation decision made (which direction/area the agent chose to explore and why).
3. Updated exploration status: what was confirmed, ruled out, or remains uncertain.

**Constraints:**
1. Keep under 150 words.
2. CRITICAL: Emphasize what CHANGED or was NEWLY OBSERVED this step. Do NOT re-describe observations already captured in previous steps.
3. Be specific—use object names, room types, and spatial relationships (e.g., ”entered a bedroom with a wooden desk near the window”).
4. STRICTLY NEVER mention ANY snapshot or frontier identifiers (e.g., ”Snapshot 2”, ”Frontier 0”) - use descriptive scene content instead.
5. AVOID relative directional references tied to transient views. Use object-anchored descriptions (e.g., ”the chair is next to the table”).
6. If no meaningful new activity occurred, return ”No significant new activity in this step.”

**Current Step $t$ Agent Outputs:**
- **Manager:** ...
- **Answerer:** ...
- **Executer:** ...

**Relevant Memory Information:**
- **Step 1:** ...
...
- **Step $t$-1:** ...
Generate summarization directly.

Figure 6. **Prompt for Recorder**.

Task: You are an indoor MEMORY MANAGEMENT AGENT responsible for CURATING and PRESERVING visual snapshots and spatial information collected by the embodied agent during its navigation, working in tandem with your existing TEXTUAL MEMORY and high-level plan.

**Instructions**:
1. CAREFULLY analyze the information needed to answer the question, paying special attention to location details, objectives, object relationships, and any mentioned or implied attributes.
2. Review all available snapshots thoroughly and cross-reference them with your TEXTUAL MEMORY. When deciding whether to retain a snapshot, adopt a conservative approach - if there is ANY potential visual relevance to the current question or its context, it should be preserved. Specifically, retain snapshots that include:
   - Any room types or spaces that may be related to the question's context, even indirectly.
   - Adjacent or connected areas that could provide spatial clues or lead to relevant locations.
   - Partial views or incomplete perspectives of objects, appliances, or features that might be useful in reasoning.
   - Environmental or contextual cues (e.g., lighting, layout, orientation) that help establish spatial understanding or support inference.
   - Objects or categories explicitly mentioned in the question, as well as those that are semantically or functionally associated.
   - Any image that provides visual background or situational information not fully captured by text, which could aid in answering the question or reconstructing the environment.
3. MEMORY COMPACTION (Textual Redundancy Filter): To prevent critical visual clues from being overwhelmed by redundant trajectory images, you may DISCARD a snapshot ONLY IF it meets BOTH of the following conditions:
   - It is completely irrelevant to the primary question or objective (contains no target objects or contextual clues).
   - Its environmental content, spatial relationships, or navigational cues are already adequately and comprehensively described in your existing textual memory.
4. When in doubt—especially if you are unsure whether the textual memory fully captures the visual nuances of the scene—err on the side of retention. Even seemingly minor or indirect visual clues can become valuable during later stages of reasoning or path reconstruction.

**Question**: [QUESTION]

**High-Level Plan**:
[ ] Pending task description...
[-] In progress task description... <!– status; rationale –>
[x] Completed task description... <!– status; rationale –>

**Previous Environmental/Historical Trajectory and Agent Analysis Information**:
- Step 1: ...

**Previously Observed Clues**:
Snapshot 0: [IMAGE]
...

**Output Format**:
1. Describe the current environment and the incremental historical trajectory.
2. First, think step by step and explain your reasoning clearly.
3. Then, provide your final answer in the exact format: Retain Snapshots: {i, ...}.

Figure 7. **Prompt for Manager**.

Task: You are an indoor embodied agent tasked with determining whether the currently available information—comprising both visual snapshots and textual memory from prior reasoning steps—is sufficient to answer the given question.

**Instructions**:
1. CAREFULLY analyze what specific information is required to answer the question (e.g., object identity, location, label, content, spatial relationship, or attribute).
2. EXAMINE ALL available evidence:
- Review every visual snapshot for relevant visual clues (e.g., labels, contents, colors, shapes, positions).
- Consult the textual memory (prior reasoning, observations, or actions from earlier steps) for contextual or inferential support.
3. You may use COMMONSENSE knowledge ONLY when:
- Direct visual evidence is ambiguous or partially missing, and
- The commonsense inference is strongly supported by typical real-world conventions and consistent with the observed context.
4. Output Answer ONLY if the combined evidence (visual + memory + justified commonsense) allows a confident and unambiguous answer.
5. If the evidence remains insufficient or ambiguous—even after considering memory and reasonable commonsense—output Continue Exploration.

**Question**: [QUESTION]

**High-Level Plan**:
[ ] Pending task description...
[-] In progress task description... <!– status; rationale –>
[x] Completed task description... <!– status; rationale –>

**Previous Environmental/Historical Trajectory and Agent Analysis Information**:
- Step 1: ...

**Previously Observed Clues**:
Snapshot 0: [IMAGE]
...

**Output Format**:
1. First, think step by step and explain your reasoning clearly.
2. If answerable, provide your final answer in the exact format: ”Answer: [your concise answer] (Evidence: Snapshot [index])”. If not, use format: ”Continue Exploration”.

Figure 8. **Prompt for Answerer**.