INTERACTIVEVLN: PROACTIVE INTERACTION VIA UNCERTAINTY-AWARE DECISION FOR ZERO-SHOT VISION LANGUAGE NAVIGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Most existing Vision-Language Navigation (VLN) methods are evaluated in static environments with single-turn instructions, limiting their ability to generalize to tasks in dynamic, interactive, realistic settings. Moreover, pre-trained navigation policies often struggle to understand complex natural language instructions, tending to overfit to narrow, task-specific directives. To address these challenges, we propose a zero-shot, interaction-enabled VLN agent framework, complementary to task-specific fine-tuning, in which the agent can actively query for task-relevant information during navigation. Unlike traditional model-based approaches, our framework centers on a multimodal long-short memory and an uncertainty-triggered Ask action. This integration enables the agent to anticipate and navigate unseen scenarios when its uncertainty is high. We evaluate our method on the FreeAskWorld and R2R-CE public benchmarks. Without any task-specific fine-tuning, our method achieves the first non-zero success among the zero-shot VLN methods we compare on FreeAskWorld. Notably, the agent autonomously determines when to request guidance: when memory is removed, it seeks assistance more often because uncertainty rises, rather than navigating blindly. These results highlight the potential of our approach for embodied AI and suggest a promising direction for deploying agents in dynamic, collaborative, realistic environments.

1 INTRODUCTION

Achieving zero-shot, highly generalizable instruction-following navigation remains a central challenge in embodied AI research. This requires robots to navigate autonomously, understand natural language instructions, plan high-level subgoals, and execute complex navigation tasks. A persistent challenge in embodied AI is overfitting: whether in vision–language–action (VLA) or vision-language navigation (VLN), most pretrained methods perform well in specific environments but fail to generalize to dynamic, interactive unseen environments. In many cases, traditional Simultaneous Localization and Mapping (SLAM) methods suffice for constrained settings, yet they do not fully leverage the complex cognitive capabilities of AI. Current pretrained models still struggle with complex instructions, and existing zero-shot methods often use large language models (LLMs) in a limited way (e.g., with shallow planning or restricted interaction), as seen in CA-Nav Chen et al. (2025b). To address this limitation, we propose a general-purpose embodied AI agent framework that avoids task-specific fine-tuning. Instead, it demonstrates zero-shot generalization with long-horizon planning and proactive interaction, moving beyond purely reactive instruction execution. Furthermore, our approach enables proactive interaction in dynamic environments, exemplified by an "ask-for-directions" strategy. Compared to exhaustive search, this strategy enables more efficient information acquisition, and the agent autonomously decides when to request guidance. Experimental results further show that when memory is removed, the agent requests directions more frequently instead of navigating via unguided exploration.

In this work, we use FreeAskWorld Peng et al. (2025a), a simulator with more complex environments, longer tasks, and support for closed-loop interactions, as our evaluation platform. We benchmark representative pretrained VLN models, navigation foundation models, and zero-shot agents,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

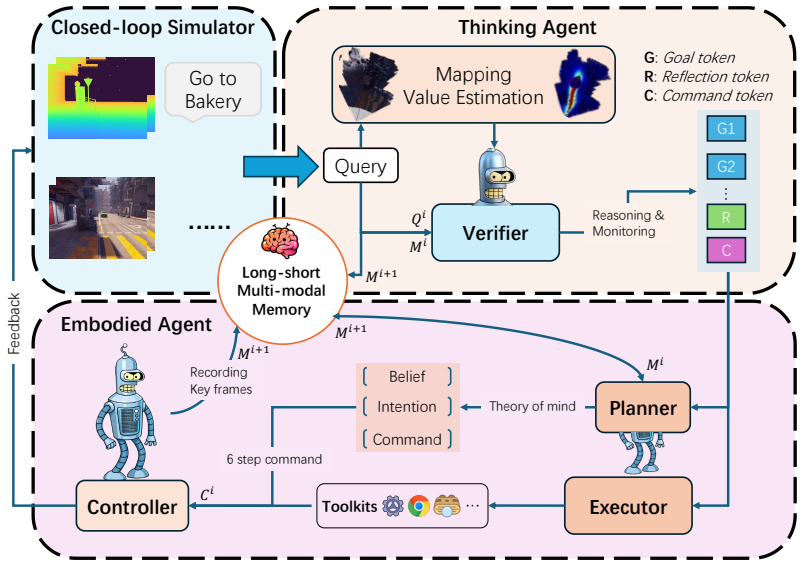


Figure 1: An overview of the InteractiveVLN framework and its data generation pipeline is presented. The system incorporates scene randomization techniques to enhance environmental diversity. The data generation module produces the FreeAskWorld dataset through this pipeline. People simulation module for modeling realistic human behaviors within virtual scenes, as well as an instruction generation module for producing navigation directives used in the Direction Inquiry Task within the simulator.

and our approach is the first zero-shot VLN method in this comparison to complete any episode on this task. In addition, we conducted experiments on the traditional R2R-CE Krantz et al. (2020b) dataset in Habitat to further validate the effectiveness of our approach.

Our key contributions of this work include:

- We formalize the task of proactive vision-language navigation as a POMDP. Navigation actions and an Ask action share one action space, and the objective trades off task success against uncertainty and query cost.
- We introduce a zero-shot embodied agent framework for long-horizon VLN without task-specific fine-tuning. A multimodal long-short memory keeps spatial experience and execution progress, and the agent asks for directions only when uncertainty is high.
- We further show that the long-short memory carries the gain. Removing it lowers oracle success and increases direction inquiries, while turning asking on or off alone leaves success unchanged. On FreeAskWorld, ours is the first zero-shot VLN method in the comparison to complete an episode.

2 Related Work

2.1 Vision Language Navigation

Vision-and-Language Navigation (VLN) is a fundamental capability for embodied agents, as it endows robots with autonomous mobility through natural language understanding and visual perception. The ability to accurately follow language instructions and reach a target location serves as a prerequisite for executing higher-level tasks, such as object manipulation and human–robot interaction. The VLN task was first formally introduced by R2R Anderson et al. (2018), which provides an indoor dataset with a discrete navigation graph. To bridge the gap between discrete navigation and real-world robot movement, R2R-CE Krantz et al. (2020b) extends VLN to continuous

environments, enabling agents to execute low-level control actions. REVERIE Qi et al. (2020) further expands the VLN paradigm to an object-oriented setting, where agents are required not only to navigate but also to identify and ground a target object specified by language. More recently, FreeAskWorld Peng et al. (2025a) transfers VLN to continuous dynamic outdoor environments under an open-vocabulary setting, where agents must navigate to named target stores. In addition to navigation, FreeAskWorld introduces interactive behaviors such as asking for directions, providing a benchmark for evaluating human-like proactive decision-making. Notably, the test dataset in FreeAskWorld is automatically generated by the simulator. FreeAD Peng et al. (2025b) further validates the effectiveness of this simulator-generated data, demonstrating consistent performance gains when training agents with the synthetic data.

Despite the progress achieved by these benchmarks, most pretrained VLN models are optimized for task-specific datasets, which often results in overfitting, limited generalization to unseen environments, and poor instruction understanding. ETPNav An et al. (2024) addresses this limitation by constructing a topological online mapping and planning obstacle-avoiding trajectories based on the learned map representation. BEVBERT An et al. (2022) further enhances this approach by introducing a hybrid perception architecture that combines a global topological map with a local metric map, enabling more effective long-term planning and fine-grained short-term navigation.

2.2 Navigation Foundation Model

Navigation foundation models typically focus on low-level planning, such as obstacle avoidance and motion control, while the target goal is specified by a high-level module. NavDP Cai et al. (2025) proposes a diffusion-policy-based planner that generates obstacle-free trajectories toward a given goal, specified either as an image or a spatial point. It demonstrates strong generalization capabilities across both simulation and real-world environments. ViNT Shah et al. (2023) is trained on large-scale visual navigation datasets and similarly supports image-conditioned goal navigation to reach target locations.

2.3 Embodied Agent

Embodied agent frameworks typically leverage the world knowledge encoded in large language models (LLMs) to perform tasks in a zero-shot manner, and employ agent-based architectures to decompose complex tasks into manageable subtasks that can be more effectively executed by the LLM. PhysiAgent Wang et al. (2025) proposes an integrated framework for vision–language–action (VLA) manipulation tasks, incorporating planning, reflection, monitoring, and memory modules. Agentic Robot Yang et al. (2025) introduces a hierarchical embodied agent framework designed for robotic manipulation tasks. AO-Planner Chen et al. (2025a) presents a hierarchical high-level and low-level agent framework for VLN with Zero-shot ability, primarily focusing on path planning rather than complex task understanding and execution. AgentFlow Li et al. (2026) introduces a comprehensive agent engineering framework for algorithm development, originally designed for internet agents with toolboxes and complex agent solver implementations. In our work, we build our embodied agent upon this foundation. CA-Nav Chen et al. (2025b) presents a VLN embodied agent framework in which the LLM is limited to subgoal decomposition, while navigation relies on value-map planning and rule-based execution, without leveraging LLMs for holistic high-level planning. AgentVLN Xin et al. (2026) uses a vision-language model to select navigation skills and can query geometric depth. It does not keep a long-short memory of visited space and execution progress, which is what our agent uses when deciding whether to ask for a language direction.

2.4 Human-inspired Intelligence

Human-inspired artificial intelligence seeks to model and leverage human cognitive processes, such as perception, memory, and reasoning, to enable more intelligent and adaptive behavior. MindPower Zhang et al. (2025) explores the incorporation of theory of mind into embodied tasks, enabling agents to exhibit and understand human-like behaviors. Ask-to-Clarify Lin et al. (2025) demonstrates the proactive capabilities of vision–language–action (VLA) models in handling ambiguous real-world tasks. It enables agents to assess task ambiguity using LLM-based reasoning and proactively interact with humans to acquire additional task-relevant information.

2.5 PROACTIVE VISION-DIALOG NAVIGATION

Talk the Walk De Vries et al. (2018) is a large-scale dialogue dataset that uniquely grounds natural language communication in both action and perception. Talk2Nav Vasudevan et al. (2021) further advances VLN with long-range instruction following using dual attention mechanisms and spatial memory modules. The Cooperative Vision-and-Dialog Navigation Thomason et al. (2020) framework established the Navigation from Dialog History (NDH) task, employing a Navigator-Oracle dynamic that mirrors the interactive logic of FreeAskWorld Peng et al. (2025a). This paradigm demonstrates the significant potential for embodied agents to navigate unknown environments by leveraging linguistic cues to resolve spatial ambiguities. However, current benchmarks highlight a substantial performance gap between AI models and human-level reasoning in utilizing historical context for effective pathfinding. Several methods learn when to ask, but the help is not a language direction. Ask4Help Singh et al. (2022) chooses between the agent's own action and an expert action. Just Ask Chi et al. (2020) uses model confusion to decide when to query a user. Good Time to Ask Zhang et al. (2023) and HARI Nguyen et al. (2022) request a goal location or a structured description of the state, goal, or subgoal. KnowNo Ren et al. (2023) asks when conformal prediction leaves more than one candidate step, in short-horizon manipulation rather than outdoor navigation. ELBAShen et al. (2025) and SCoA Zhu et al. (2021) can receive language help, but the asking policy is still trained. We do not train this policy. A fixed uncertainty score triggers a templated query, and the simulator returns a language direction for the current plan.

3 METHODOLOGY

3.1 PROBLEM FORMULATION

We formulate the zero-shot, interaction-enabled Vision-Language Navigation (VLN) task as a Partially Observable Markov Decision Process (POMDP), defined by the tuple (S, A, O, E, T, M). Crucially, we redefine the action space A to explicitly incorporate information-seeking behavior:

$$A = \mathcal{A}_{nav} \cup \{\text{Ask}\}, \tag{1}$$

where $\mathcal{A}_{nav}$ contains standard navigation primitives and `Ask` is an explicit interactive action for information seeking.

Unlike prior VLN works, we explicitly model navigation as a strategic trade-off between execution and information acquisition. At each timestep t, the agent selects an action $a_t \in A$ based on the instruction I, current observation o_t, and memory M_t. The agent's objective is to maximize task success while minimizing both cumulative uncertainty and interaction costs, formulated as:

$$\max_{\pi} \mathbb{E}\left[F(\tau) - \lambda \sum_{t=1}^{T} \mathcal{U}(o_t, M_t) - \beta \sum_{t=1}^{T} \mathbb{I}(a_t = \text{Ask})\right], \tag{2}$$

where $F(\tau)$ is the task success indicator, $\mathcal{U}(\cdot)$ quantifies the navigation uncertainty stemming from partial observability or instruction ambiguity, and β is a penalty term that regulates the frequency of querying to ensure efficient interaction.

3.2 DECISION RULE

To solve this objective, we formalize the *ask* behavior as a strategic decision governed by the agent's internal state. Instead of reactive querying, the agent executes the `Ask` action only when its uncertainty exceeds a confidence threshold δ. The decision rule is formulated as:

$$a_t = \begin{cases} \text{Ask}, & \text{if } \mathcal{U}(o_t, M_t) > \delta \\ \arg\max_{a \in \mathcal{A}_{nav}} \pi(a | I, o_t, M_t), & \text{otherwise} \end{cases} \tag{3}$$

where δ is a threshold that balances navigation efficiency and communication cost. This formulation ensures that the agent only seeks external assistance when its memory or perception is insufficient, or when the sampled actions disagree.

Once the `Ask` action is activated, the agent generates a concise natural language query q_t to seek assistance. We employ a template-based approach to ensure the query is goal-oriented and context-aware. Specifically, the agent formulates queries based on two primary taxonomies:

Table 1: Performance Comparison on FreeAskWorld Benchmark

Model	TL	SR(%)	OSR(%)	SPL(%)	NE	ONE	NDI	Time
Human								
Human (no ask)	47.5	40.2	41.3	38.2	18.3	11.3	0.00	100
Human (ask)	59.9	**82.6**	**82.6**	**71.2**	**3.49**	**1.63**	**0.78**	100
Pretrained VLN Models								
ETPNav An et al. (2024)	31.2	0.0	0.0	0.0	32.9	28.7	0.0	100
BEVBert An et al. (2022)	14.6	0.0	0.0	0.0	31.0	29.0	0.0	100
ETPNav-FT	33.6	0.0	1.09	0.0	31.6	**27.1**	0.0	100
BEVBert-FT	18.7	0.0	0.0	0.0	**30.0**	28.5	0.0	100
Navigation Foundation Models								
NavDP Cai et al. (2025)	38.44	1.09	3.26	0.95	35.09	25.07	0.0	100
ViNT Shah et al. (2023)	11.88	**2.17**	2.17	**2.13**	32.22	27.22	0.0	100
Zero-shot Methods								
VLN-Zero (GPT-4o) Bhatt et al. (2025)	2.76	0.0	0.0	0.0	31.27	30.70	0.0	150
InstructNav (GPT-4o) Long et al. (2024)	12.68	0.0	0.0	0.0	29.72	27.64	0.0	150
InteractiveVLN (Qwen2.5-VL-72B)	1.70	0.0	0.0	0.0	29.13	28.89	0.0	150
InteractiveVLN (Gemini-2.5-Flash)	5.22	0.0	0.0	0.0	28.20	27.23	0.0	150
InteractiveVLN (GPT-4o-mini)	7.14	1.09	1.09	0.69	28.31	27.24	0.09	150
InteractiveVLN (GPT-4o)	13.58	**1.09**	**4.35**	**1.16**	**27.91**	**24.70**	0.05	150

Table 2: Standard Deviation of Evaluation Metrics on the FreeAskWorld Benchmark over 5 Runs

Model	SR	OSR	NE	ONE	NDI
Navigation Foundation Models					
NavDP	0.6	0.9	2.7	1.8	0.0
ViNT	0.7	0.7	2.3	1.6	0.0
Zero-shot Methods					
VLN-Zero	0.0	0.0	2.3	2.1	0.0
InstructNav	0.0	0.0	2.0	1.8	0.0
InteractiveVLN(Qwen2.5-VL)	0.0	0.0	1.9	1.7	0.0
InteractiveVLN(Gemini-2.5)	0.0	0.0	1.8	1.6	0.0
InteractiveVLN(GPT-4o-mini)	0.5	0.5	1.9	1.7	0.03
InteractiveVLN(GPT-4o)	0.6	1.0	1.8	1.5	0.02

- Type A (Directional): Focuses on navigation guidance, e.g., *"I am at the [Location], should I turn left or right to find the [Target]?"*
- Type B (Localization): Focuses on object-target relationships, e.g. *"I can see a [Object], is the [Target] nearby?"*

The response is generated by a controlled oracle policy of FreeAskWorld Peng et al. (2025a) simulator conditioned on the state of the ground-truth environment, ensuring consistency and reproducibility.

3.3 Agent Framework

As illustrated in Fig. 1, our embodied agent framework adopts a multi-step reasoning paradigm with uncertainty-aware decision making. Following the principles of Deep Self-Evolving Reasoning Liu et al. (2025), agent frameworks operating as Markov chains with explicit reasoning steps should maintain a consistent bias toward correct outcomes rather than drifting into erroneous states. To achieve this, our framework integrates uncertainty estimation, reflective verification, structured plan-

Table 3: Performance comparison on R2R-CE benchmark.

Method	Zero-Shot	Observation				R2R Val-Unseen			
		S.RGB	Pano.	Depth	Odo.	NE↓	OS↑	SR↑	SPL↑
LAW Raychaudhuri et al. (2021)	×	✓	✓	✓	✓	7.90	26.0	24.0	20.0
Seq2Seq Krantz et al. (2020a)	×	✓		✓		7.77	30.0	25.0	20.0
Navid Zhang et al. (2024)	×	✓				5.47	49.0	37.0	35.0
NaVILA Cheng et al. (2024)	×	✓				5.22	62.5	54.0	49.0
AO-Planner Chen et al. (2025a)	✓		✓	✓		**6.95**	38.3	25.5	**16.6**
InstructNav Long et al. (2024)	✓	✓	✓	✓		9.20	47.0	17.0	11.0
CA-Nav Chen et al. (2025b)	✓	✓	✓	✓	✓	8.01	**48.0**	25.3	10.8
InteractiveVLN (Ours)	✓	✓	✓	✓	✓	7.58	47.0	**26.8**	8.26

Table 4: Ablation study of InteractiveVLN on the FreeAskWorld benchmark. "PA" denotes Proactive Action, specifically the agent's ability to execute ask for directions action.

Planner	**Memory**	**Verifier**	**PA**	**TL**	**SR(%)**	**OSR(%)**	**SPL(%)**	**NE**	**ONE**	**NDI**	**Time**
✓	✓	✓	✓	13.58	**1.09**	**4.35**	**1.16**	**27.91**	**24.70**	0.05	150
✓	✓	×	✓	13.15	0.0	0.0	0.0	29.39	26.00	**0.36**	150
✓	×	✓	✓	13.85	0.0	1.09	0.0	29.24	25.34	0.28	150
✓	×	×	✓	14.20	0.0	1.09	0.0	28.65	25.85	0.27	150
✓	×	×	×	15.11	0.0	1.09	0.0	28.88	25.98	0.0	150

ning, and proactive interaction into a unified agentic pipeline. The pipeline uses a general-purpose vision-language model and does not train a new navigation policy.

At each timestep, the agent first evaluates its internal uncertainty based on visual perception, policy ambiguity, and memory consistency as we formulate in Uncertainty Estimation.

The Verifier acts as a consistency evaluator that continuously assesses the alignment among the current observation o_t, predicted subgoal state g_t, and historical memory M_t. Formally, it computes a consistency score:

$$C_t = \mathrm{sim}(f(o_t), f(g_t))$$

where g_t denotes the expected subgoal representation. If $C_t < \epsilon$, the verifier triggers replanning and reflection. This mechanism filters irrational or misleading intermediate reasoning steps, mitigates error accumulation, and produces more reliable contextual representations for subsequent decision-making.

The Planner performs structured reasoning over the task instruction I, belief state, uncertainty estimation, memory state, and verifier feedback to generate short-horizon action sequences. Specifically, the planner incorporates uncertainty-aware reasoning to determine whether the current navigation confidence is sufficient for autonomous execution or whether proactive interaction should be activated. When the estimated uncertainty exceeds a predefined threshold, the planner invokes the *Ask* action to actively acquire external information from nearby pedestrians. This process enables adaptive decision-making under ambiguous or partially observable environments.

The planning process is implemented through structured prompting that enforces: (i) subgoal decomposition, (ii) action consistency, (iii) progress monitoring, (iv) uncertainty reflection, and (v) interaction-aware reasoning.

The Executor receives commands generated by the Planner and executes them using the integrated toolbox. The toolbox includes *Grounded-SAM2* Ren et al. (2024) and Google Research modules. Grounded-SAM2 is employed to segment ground surfaces, obstacles, and landmarks, as illustrated in Fig. 4 in the appendix, thereby improving spatial perception and environmental understanding.

Overall, the proposed framework combines uncertainty estimation, reflective verification, memory-based reasoning, and proactive interaction into a unified embodied agent architecture, enabling more human-like adaptive navigation in complex dynamic environments.

3.4 Uncertainty Estimation

The agent executes `Ask` only when a scalar uncertainty score exceeds a fixed threshold. The score is a weighted sum of three terms,

$$\mathcal{U}(o_t, M_t) = 0.1\, U_{\text{vis}} + 0.7\, U_{\text{policy}} + 0.2\, U_{\text{memory}}. \tag{4}$$

We scale each term to $[0, 1]$ before weighting. The weights are nonnegative and sum to one, so $\mathcal{U}(o_t, M_t) \in [0, 1]$. Ask is triggered when $\mathcal{U}(o_t, M_t) > 0.65$. The weights and the threshold were chosen once and then held fixed for every episode, run, and backbone. They were not calibrated, and other thresholds were not tested.

Visual uncertainty is the Shannon entropy of the 8-bit grayscale histogram of the current image I. Since this entropy is at most $\log_2 256 = 8$,

$$U_{\text{vis}}(I) = -\frac{1}{8} \sum_{b=0}^{255} p_b \log_2 p_b \in [0, 1], \tag{5}$$

where p_b is the fraction of pixels with grayscale value b, and $0 \log 0$ is taken as 0. This is a proxy for low-level visual complexity, not a color measurement.

Policy uncertainty is computed from repeated samples, not from a confidence stated by the model. At the same state the planner is queried $K = 8$ times. Let $n(a)$ be the number of samples equal to action a, and set $p(a) = n(a)/K$. The raw entropy is

$$\tilde{U}_{\text{policy}} = -\sum_{a:\, p(a)>0} p(a) \log p(a). \tag{6}$$

It is zero when all eight samples agree and increases when they disagree.

Memory uncertainty compares the current image with the latest image kept in memory. Let $\bar{e}(I)$ denote the mean embedding vector of image I. The raw distance is

$$\tilde{U}_{\text{memory}} = \|\bar{e}(I_t) - \bar{e}(I^{\text{mem}})\|_2. \tag{7}$$

A larger value means that the current view is farther from recent memory.

3.5 Mapping

We employ RTAB-Map Labbé & Michaud (2019) for online mapping, using RGB-D and odometry data to generate a top-down occupancy map. This map is fed into the value estimation module, which comprises two main components: a trajectory-based value and a directional fan value. The overall decision cost at each grid cell (i, j) is:

$$\begin{aligned} V(i,j) = \text{clip}\Big(&C_{\text{map}}(i,j) + \lambda_t\, V_{\text{traj}}(i,j) \\ &+ \lambda_f\, V_{\text{fan}}(i,j), 0, 1\Big) \end{aligned} \tag{8}$$

where C_{map} is the occupancy cost, V_{traj} penalizes cells near previously visited waypoints, and V_{fan} penalizes cells along previously explored directions. Weighting coefficients λ_t and λ_f balance the contributions of these terms.

In addition to value estimation, we explicitly mark the agent's ego position and heading on the map. This visual annotation facilitates the MLLM's understanding of the agent's current state and spatial context. Representative qualitative results are presented in Figures 3, 5, and 2.

3.6 MEMORY

We design a multi-modal memory module to store sequential observations and internal states, including visual inputs, textual information, trajectories, and positional data, all represented within a unified textual abstraction when appropriate. Furthermore, we adopt a long–short memory architecture to support both global reasoning and local execution monitoring. The long-term memory keeps a compressed record of visited regions, scene semantics, and past exploration. The short-term memory keeps the current subgoal, recent observations, and execution progress, and the verifier uses it to check the current step. Together, this hierarchical memory design allows the agent to jointly leverage global semantic understanding and local perceptual feedback, facilitating robust long-horizon planning and execution in complex environments.

4 EXPERIMENTS

4.1 BASELINES

In addition to the baselines introduced in the related work, we evaluate zero-shot methods, including VLN-Zero Bhatt et al. (2025) and InstructNav Long et al. (2024). We also report a human baseline, where five human evaluators complete the benchmark and the results are averaged across all participants. We adapted each baseline to operate within the closed-loop simulator. For models requiring specific inputs, such as RGB-D images, odometry, or other sensor data, the corresponding information was provided. Additionally, goal images were collected for the navigation foundation models.

4.2 CLOSED-LOOP EXPERIMENTS SETTING

All baselines were evaluated in the FreeAskWorld Peng et al. (2025a) simulator on a single NVIDIA GeForce RTX 5070 Ti and interfaced via ROS2. Each method was evaluated over five independent runs, which required approximately 10 hours in total for each five runs. The evaluation duration was set to 100 seconds for pretrained VLN and navigation foundation models, due to their high planning frequency, and 150 seconds for zero-shot methods, reflecting the longer reasoning time required by LLMs. For InteractiveVLN, writing out the reasoning takes about 5–6 seconds per decision, and outputting only the action takes less than 1.5 seconds. All baselines were evaluated under identical environment settings, including the number of cars, pedestrians, and other dynamic elements. We report the average performance over five runs, and the standard deviation is provided in Table 2.

Additionally, we evaluated our methods in Habitat using the R2R-CE dataset. We extracted RGB-D data, odometry, and camera intrinsic and extrinsic parameters to faithfully reproduce our methods in this environment.

4.3 MAIN RESULTS

The closed-loop evaluation results on FreeAskWorld are presented in fig. 1. By comparing different baselines, we observe that the human baseline—equipped with the ability to ask for directions achieves superior performance, highlighting the importance of proactive interaction. Pretrained models denoted by FT indicate fine-tuned variants by dataset generated by FreeAskWorld. While fine-tuning improves performance, these models still struggle to solve the task due to the long-horizon nature and the complexity of the instructions. Navigation foundation models exhibit limited high-level reasoning capabilities, as they predominantly rely on explicit goal specifications, such as a goal image or a target point. As a result, they are only able to solve the simplest instances in the test set. For example, the results of ViNT show that only one out of 46 episodes (2.17%) falls into this category of straightforward tasks. Nevertheless, these results also highlight the strong low-level navigation abilities of foundation models. When provided with explicit goals, they can effectively avoid obstacles and generate smooth and stable trajectories. This is further evidenced by ViNT, which achieves an SPL of 2.13%, outperforming InteractiveVLN (1.13%) in terms of low-level navigation execution. Moreover, NavDP attains an average trajectory length of 38.44 meters, demonstrating its strong motion planning capability. However, its large navigation error (35.09 meters) indicates inaccurate global planning, which can be primarily attributed to the lack of a high-level planning module. Zero-shot methods further demonstrate the difficulty of the task. Current

state-of-the-art approaches of R2R-CE, including VLN-Zero and InstructNav, fail to complete any episode, resulting in a success rate of 0. Notably, although InstructNav produces trajectories with lengths comparable to our method (InteractiveVLN), it fails to reach the target destinations. InteractiveVLN is instantiated with Qwen2.5-VL-72B, Gemini-2.5-Flash, GPT-4o-mini, and GPT-4o. Among them, GPT-4o achieves the best performance, demonstrating strong multimodal perception and reasoning capabilities. Specifically, InteractiveVLN (GPT-4o) attains a success rate of 1.09 and an oracle success rate of 4.35. We observe that failures often occur when the agent does not stop correctly, likely due to limited distance perception. Addressing this limitation is left for future work.

The R2R-CE experiment results are shown in Fig. 3. Our success rate is 26.8, but the SPL of 8.26 is substantially lower than AO-Planner's SPL of 16.6, indicating that our approach lacks fine-grained low-level planning capabilities. In contrast, AO-Planner excels in this aspect due to its more precise low-level trajectory planning.

4.4 Ablation Study

The ablation study results are presented in fig. 4. When the memory and verifier modules are disabled, we do not modify the underlying reasoning procedure, instead, we remove their corresponding inputs to the planner, thereby ensuring comparable reasoning time across different settings. Disabling the memory module, the verifier module, or both results in a noticeable increase in the Number of Direction Inquiries (NDI), indicating elevated uncertainty during the reasoning process. As a consequence, the agent more frequently executes the ask-for-directions action to acquire additional environmental information. Furthermore, when the verifier, memory, and proactive action modules are all disabled, the trajectory length increases. This is because the agent becomes less inclined to request directions and instead predominantly focuses on movement actions. With only the planner, turning asking on shortens the trajectory from 15.11 to 14.20 and lowers NE from 28.88 to 28.65. SR and OSR do not change. Removing the long-short memory lowers oracle success from 4.35 to 1.09 and raises NDI from 0.05 to 0.28, so the agent asks more often once that context is gone.

4.5 Qualitative Results

Qualitative results are illustrated in Fig. 2 and Fig. 5. The figure visualizes a complete navigation episode, including the final value map predicted by the agent, the goal location, and the task instruction that jointly define the task context. In addition, Fig. 2 in appendix presents an example reasoning step during online execution. The reasoning process explicitly illustrates subgoal decomposition and progress monitoring, incorporates action reflection based on the most recent observations, and provides structured feedback to guide subsequent planning decisions. Conditioned on the verifier's contextual feedback and other input, the planner formulates coherent beliefs, intentions, and actions.

4.6 Real-world Deployment

We deploy InteractiveVLN on a Unitree Go2 without task-specific fine-tuning. The robot follows the instruction to turn right and find the exit door, as shown in Figure 6. This deployment shows that the same zero-shot workflow can be executed on a physical robot.

5 Conclusion

In this work, we presented InteractiveVLN, an interaction-enabled framework for zero-shot vision-language navigation in ambiguous and dynamic environments. We further formalized an uncertainty-aware decision mechanism that enables the agent to proactively seek external information when its internal knowledge is insufficient. The framework uses a general-purpose vision-language model and a long-short memory, without training a new navigation policy. On FreeAskWorld, it is the first zero-shot VLN method in our comparison to complete an episode. On R2R-CE, its SPL remains lower than that of specialized low-level planners. Ablation studies show that removing the long-short memory lowers oracle success, while asking alone leaves SR and OSR unchanged.

AI USE STATEMENT

In this work, we used generative AI tools for no task that requires disclosure. We have not used generative AI tools for generating synthetic data, developing theoretical models, formulating mathematical claims, writing proofs, proposing hypotheses, designing experiments, implementing methods, translation, cleaning datasets, qualitative analysis, or interpreting results, and no other required-disclosure task is applicable to this work. Additionally, we used generative AI tools for editing the manuscript to improve readability. We have reviewed all AI-assisted work. The authors checked the edited text against the manuscript and the reported results. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We aim to make the reported results reproducible. The method, decision rule, and experimental protocol are described in the paper, and the same fixed settings are used across the reported runs. Implementation details needed to repeat the experiments will be included with the public code release. After acceptance, we will release the code publicly.

REFERENCES

Dong An, Yuankai Qi, Yangguang Li, Yan Huang, Liang Wang, Tieniu Tan, and Jing Shao. Bevbert: Multimodal map pre-training for language-guided navigation. *arXiv preprint arXiv:2212.04385*, 2022.

Dong An, Hanqing Wang, Wenguan Wang, Zun Wang, Yan Huang, Keji He, and Liang Wang. Etpnav: Evolving topological planning for vision-language navigation in continuous environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3674–3683, 2018.

Neel P Bhatt, Yunhao Yang, Rohan Siva, Pranay Samineni, Daniel Milan, Zhangyang Wang, and Ufuk Topcu. Vln-zero: Rapid exploration and cache-enabled neurosymbolic vision-language planning for zero-shot transfer in robot navigation. *arXiv preprint arXiv:2509.18592*, 2025.

Wenzhe Cai, Jiaqi Peng, Yuqiang Yang, Yujian Zhang, Meng Wei, Hanqing Wang, Yilun Chen, Tai Wang, and Jiangmiao Pang. Navdp: Learning sim-to-real navigation diffusion policy with privileged information guidance. *arXiv preprint arXiv:2505.08712*, 2025.

Jiaqi Chen, Bingqian Lin, Xinmin Liu, Lin Ma, Xiaodan Liang, and Kwan-Yee K Wong. Affordances-oriented planning using foundation models for continuous vision-language navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 23568–23576, 2025a.

Kehan Chen, Dong An, Yan Huang, Rongtao Xu, Yifei Su, Yonggen Ling, Ian Reid, and Liang Wang. Constraint-aware zero-shot vision-language navigation in continuous environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025b.

An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Zaitian Gongye, Xueyan Zou, Jan Kautz, Erdem Bıyık, Hongxu Yin, Sifei Liu, and Xiaolong Wang. Navila: Legged robot vision-language-action model for navigation. *arXiv preprint arXiv:2412.04453*, 2024.

Ta-Chung Chi, Minmin Shen, Mihail Eric, Seokhwan Kim, and Dilek Hakkani-Tur. Just ask: An interactive learning framework for vision and language navigation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 2459–2466, 2020.

Harm De Vries, Kurt Shuster, Dhruv Batra, Devi Parikh, Jason Weston, and Douwe Kiela. Talk the walk: Navigating new york city through grounded dialogue. *arXiv preprint arXiv:1807.03367*, 2018.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *European Conference on Computer Vision*, pp. 104–120. Springer, 2020a.

Jacob Krantz, Erik Wijmans, Arjun Majundar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision and language navigation in continuous environments. In *European Conference on Computer Vision (ECCV)*, 2020b.

Mathieu Labbé and François Michaud. Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *Journal of field robotics*, 36(2):416–446, 2019.

Zhuofeng Li, Haoxiang Zhang, Seungju Han, Sheng Liu, Jianwen Xie, Yu Zhang, Yejin Choi, James Zou, and Pan Lu. In-the-flow agentic system optimization for effective planning and tool use. In *International Conference on Learning Representations (ICLR)*, 2026.

Xingyao Lin, Xinghao Zhu, Tianyi Lu, Sicheng Xie, Hui Zhang, Xipeng Qiu, Zuxuan Wu, and Yu-Gang Jiang. Ask-to-clarify: Resolving instruction ambiguity through multi-turn dialogue. *arXiv preprint arXiv:2509.15061*, 2025.

Zihan Liu, Shun Zheng, Xumeng Wen, Yang Wang, Jiang Bian, and Mao Yang. Deep self-evolving reasoning. *arXiv preprint arXiv:2510.17498*, 2025.

Yuxing Long, Wenzhe Cai, Hongcheng Wang, Guanqi Zhan, and Hao Dong. Instructnav: Zero-shot system for generic instruction navigation in unexplored environment. *arXiv preprint arXiv:2406.04882*, 2024.

Khanh X Nguyen, Yonatan Bisk, and Hal Daumé Iii. A framework for learning to request rich and contextually useful information from humans. In *International Conference on Machine Learning*, pp. 16553–16568. PMLR, 2022.

Yuhang Peng, Yizhou Pan, Xinning He, Jihaoyu Yang, Xinyu Yin, Han Wang, Xiaoji Zheng, Chao Gao, and Jiangtao Gong. Freeaskworld: An interactive and closed-loop simulator for human-centric embodied ai. *arXiv preprint arXiv:2511.13524*, 2025a.

Yuhang Peng, Sidong Wang, Jihaoyu Yang, Shilong Li, Han Wang, and Jiangtao Gong. Bench2freead: A benchmark for vision-based end-to-end navigation in unstructured robotic environments. *arXiv preprint arXiv:2503.12180*, 2025b.

Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. Reverie: Remote embodied visual referring expression in real indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9982–9991, 2020.

Sonia Raychaudhuri, Saim Wani, Shivansh Patel, Unnat Jain, and Angel Chang. Language-aligned waypoint (law) supervision for vision-and-language navigation in continuous environments. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 4018–4028, 2021.

Allen Z Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, et al. Robots that ask for help: Uncertainty alignment for large language model planners. *arXiv preprint arXiv:2307.01928*, 2023.

Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

Dhruv Shah, Ajay Sridhar, Nitish Dashora, Kyle Stachowicz, Kevin Black, Noriaki Hirose, and Sergey Levine. Vint: A foundation model for visual navigation. *arXiv preprint arXiv:2306.14846*, 2023.

Ying Shen, Daniel Bis, Cynthia Lu, and Ismini Lourentzou. Elba: Learning by asking for embodied visual navigation and task completion. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pp. 5177–5186, 2025.

Kunal Pratap Singh, Luca Weihs, Alvaro Herrasti, Jonghyun Choi, Aniruddha Kembhavi, and Roozbeh Mottaghi. Ask4help: Learning to leverage an expert for embodied tasks. *Advances in Neural Information Processing Systems*, 35:16221–16232, 2022.

Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. Vision-and-dialog navigation. In *Conference on Robot Learning*, pp. 394–406. PMLR, 2020.

Arun Balajee Vasudevan, Dengxin Dai, and Luc Van Gool. Talk2nav: Long-range vision-and-language navigation with dual attention and spatial memory. *International Journal of Computer Vision*, 129(1):246–266, 2021.

Zhihao Wang, Jianxiong Li, Jinliang Zheng, Wencong Zhang, Dongxiu Liu, Yinan Zheng, Haoyi Niu, Junzhi Yu, and Xianyuan Zhan. Physiagent: An embodied agent framework in physical world. *arXiv preprint arXiv:2509.24524*, 2025.

Zihao Xin, Wentong Li, Yixuan Jiang, Ziyuan Huang, Bin Wang, Piji Li, Jianke Zhu, Jie Qin, and Shengjun Huang. Agentvln: Towards agentic vision-and-language navigation. In *European Conference on Computer Vision*, pp. 114–132. Springer, 2026.

Zhejian Yang, Yongchao Chen, Xueyang Zhou, Jiangyue Yan, Dingjie Song, Yinuo Liu, Yuting Li, Yu Zhang, Pan Zhou, Hechang Chen, et al. Agentic robot: A brain-inspired framework for vision-language-action models in embodied agents. *arXiv preprint arXiv:2505.23450*, 2025.

Jenny Zhang, Samson Yu, Jiafei Duan, and Cheston Tan. Good time to ask: A learning framework for asking for help in embodied visual navigation. In *2023 20th International Conference on Ubiquitous Robots (UR)*, pp. 503–509. IEEE, 2023.

Jiazhao Zhang, Kunyu Wang, Rongtao Xu, Gengze Zhou, Yicong Hong, Xiaomeng Fang, Qi Wu, Zhizheng Zhang, and He Wang. Navid: Video-based vlm plans the next step for vision-and-language navigation. *arXiv preprint arXiv:2402.15852*, 2024.

Ruoxuan Zhang, Qiyun Zheng, Zhiyu Zhou, Ziqi Liao, Siyu Wu, Jian-Yu Jiang-Lin, Bin Wen, Hongxia Xie, Jianlong Fu, and Wen-Huang Cheng. Mindpower: Enabling theory-of-mind reasoning in vlm-based embodied agents. *arXiv preprint arXiv:2511.23055*, 2025.

Yi Zhu, Yue Weng, Fengda Zhu, Xiaodan Liang, Qixiang Ye, Yutong Lu, and Jianbin Jiao. Self-motivated communication agent for real-world vision-dialog navigation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1574–1583. IEEE, 2021.

A Qualitative Results

In this appendix, we provide further qualitative results to complement the main paper.

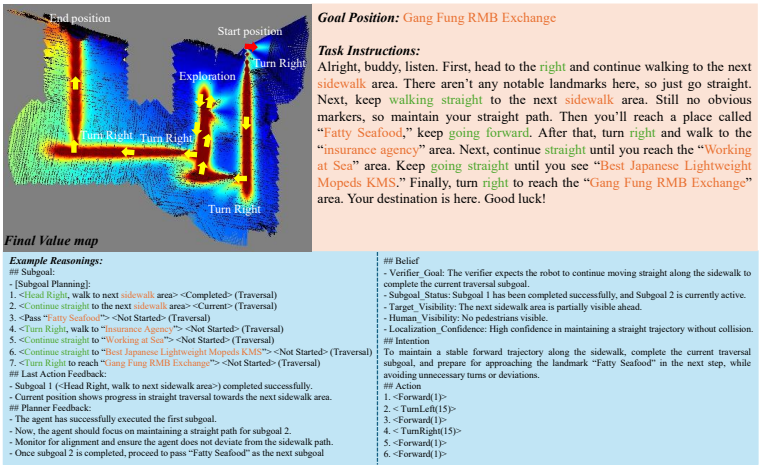


Figure 2: Qualitative results of the agent on representative navigation tasks.

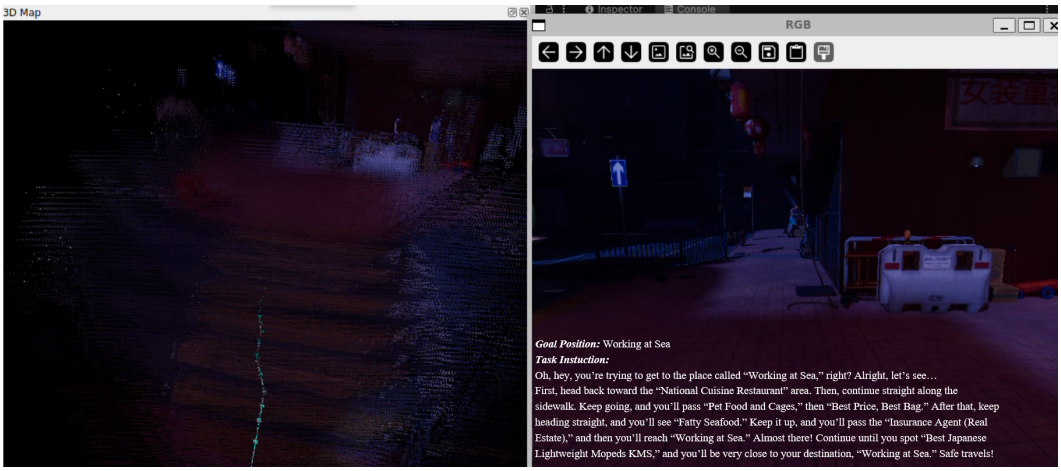


Figure 3: Qualitative results illustrating the agent's mapping and behavior in accordance with the task instructions.

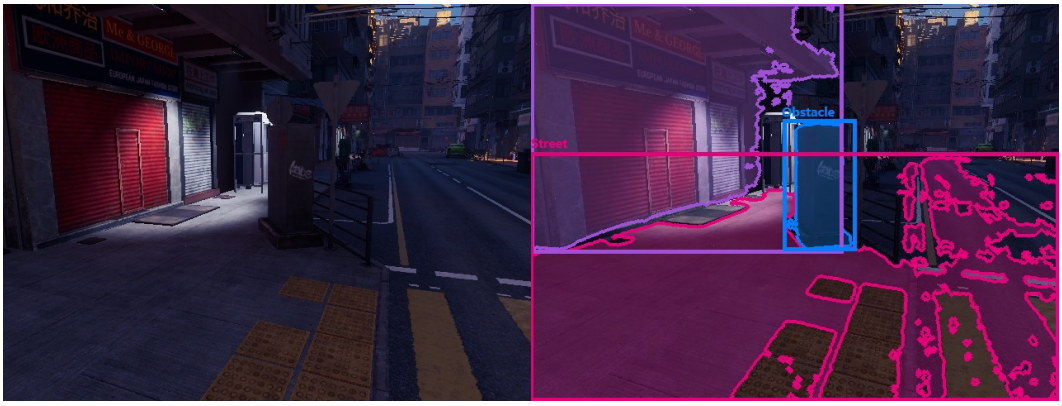
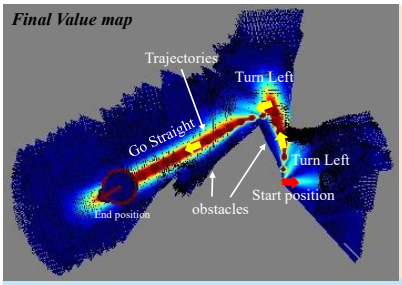


Figure 4: Qualitative results of the GroundedSAM2 toolbox used by the agent, with prompts generated by the agent specifying `[Street, Obstacle, Building]`.



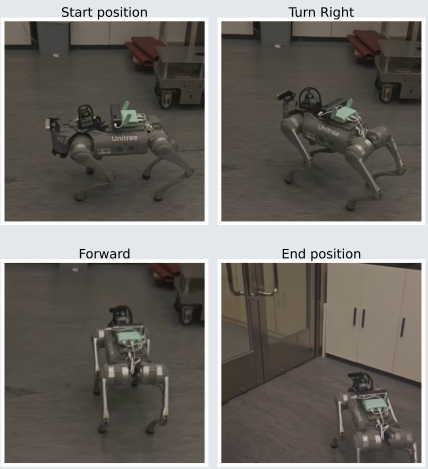
Goal Position: Jack's Restaurant

Task Instructions:
OK, You can go like this: first, walk to the left toward the area labeled "Fresh Pastries and Bread." There aren't any particularly noticeable landmarks there, but just keep going. Next, turn left until you reach the "Women's and Children's Clothing" area. There aren't any obvious markers here either, but maintain your direction. Then, walk straight ahead until you see "Jack's Restaurant." Your destination is in this area. It should take a few minutes to get there. I hope these directions help!

Example Reasonings:
Subgoal:
- [Subgoal Planning]:
1. <Turn Left> < Not Started > (Orientation)
2. <Pass "Fresh Pastries and Bread"> < Not Started > (Traversal)
3. < Turn Left, pass the "Women's and Children's Clothing"> <Not Started> (Traversal)
4. < Go straight , to the "Jack's Restaurant"> <Not Started> (Traversal)
Last Action Feedback:
First step, no previous action to verify.
Planner Feedback:
- Current position and orientation appear to show no significant movement or orientation change has been made, likely indicating the initial state or error.
- Recommend starting with turning to left, aligning with the first step of task instruction.
- Ensure the planner initiates the orientation subgoal correctly in the upcoming steps to achieve progression towards "Jack's Restaurant".

Belief
- Verifier_Goal: The verifier wants the robot to turn left.
- Target_Visibility: The immediate turning target is not visible.
- Human_Visibility: A person is visible near the storefront but is not blocking the path.
- Localization_Confidence: Moderate confidence in executing turn left without collisions.
Intention
To align with verifier instructions, perform turn left cautiously, ensuring obstacle avoidance.
Action
1. < TurnLeft(45)>
2. <Forward(1)>
3. <TurnLeft(45)>
4. <Forward(2)>
5. <Forward(1)>
6. <Forward(1)>

Figure 5: More Qualitative results.



Goal Position: Exit door

Task Instructions:

Turn right, then find the exit door.

Example Reasonings:
Subgoal:
- [Subgoal Planning]:
1. <Turn right> <Completed> (Orientation)
2. <Find the exit door> <Current> (Traversal)
Last Action Feedback:
- Subgoal 1 (<Turn right>) completed.
- The robot is moving forward after the turn.
Planner Feedback:
- The first subgoal has been executed.
- Go forward, then turn left to search for the exit door.
- Do not add another right turn.

Belief
- Verifier_Goal: The verifier expects the robot to find the exit door.
- Subgoal_Status: Subgoal 1 is completed, and Subgoal 2 is currently active.
- Target_Visibility: The exit door is not confirmed in the forward view yet.
- Human_Visibility: No person is blocking the path.
- Localization_Confidence: The right turn is complete and the robot is moving.
Intention
To go forward after the right turn, then turn left toward the exit door.
Action
1. <Forward>
2. <TurnLeft>
3. <Forward>
4. <Stop at the exit door>

Figure 6: Qualitative rollout on a Unitree Go2.