

# HISTORY TO FUTURE : Evolving Agent with Experience and Thought for Zero-shot Vision-and-Language Navigation

## Supplementary Material

*This document provides more details of datasets, visualization examples, and discussion, which are organized as:*

- *Datasets Details* (cf. § A);
- *Visualization Example* (cf. § B).
- *Discussion* (cf. § C).

### A. Experimental Details

*This supplement is for Section 5.1 of the main paper.*

#### A.1. R2R-CE Benchmark

We evaluate our approach using the R2R-CE dataset within simulated environments. R2R-CE is based on the Matterport3D [1] scenes, and it converts the discrete paths of the original R2R dataset into continuous environments using the Habitat Simulator. For a fair comparison, we strictly follow to the protocol of previous zero-shot approaches [9, 13, 14, 19] across 100 episodes to ensure direct comparability. In the simulated evaluation, the model was run on a single RTX 4090 GPU.

#### A.2. NavRAG-CE Benchmark

Following the above protocol, we randomly sample 100 trajectory-instruction pairs from more complex NavRAG-CE [17] for a better evaluation, which requires the agent to understand difficult human demands. Inspired by previous methods [1, 8, 10, 12, 15], NavRAG-CE set up several different user roles (with varying ages, genders, occupations, lifestyles, and demands to the navigation agent) to simulate and record the instructions sent to the navigation agent during one day of this role. For each 3D scene, NavRAG constructs a scene description tree [3] in a bottom-up manner for hierarchical scene representations.

### B. Visualization Example

#### B.1. Navigation Demos of Simulator

*This supplement is for Figure 3 of the main paper.* We visualize a navigation example in the simulated environments, as shown in Figure 1. Here, we also report navigation trajectories of the baseline model (the bottom) for a better comparison with our approach (the top). As shown in Figure 1, the baseline model achieves navigation error as it obtains poor decision-making by **Naive Reasoning** [4, 9, 18], while our EvoNav achieves correct decision-marking of navigation by promoting the agent with **Feedback Reasoning** of history experience and future thought via History Chain-of-Experience and future Chain-of-Thought.

#### B.2. Navigation Demos of the Real World

*This supplement is for Figure 4 of the main paper.* We visualize several navigation examples in the real-world environments, as shown in Figure 2. Give a human instruction, EvoNav can effectively adapt the general robot and navigate to the target location. It proves its capability for navigation in the real world.

#### B.3. Prompt of Action Prediction

*This supplement is for Section 4.4 of the main paper.* Figure 3 illustrates the complete prompts required for action prediction, which leverage historical experience and the future landmark to aid decision-making.

### C. Discussion

*This supplement is for Section 6 of the main paper.*

#### C.1. Lifelong Learning: Memory vs our Experience

Recent works [5, 6, 16, 20] propose maintaining a global memory to store the 3D structure of the environment for embodied tasks, offering a promising solution for lifelong navigation. While sharing a similar spirit of evolving decision-making through historical experience, our approach diverges in three key aspects: (1) Dependency: Updating such memories typically necessitates computationally heavy **localization** tools (e.g., **SLAM** [6, 16]) to construct explicit representations, whereas our framework is **SLAM-free**, eliminating the need for complex geometric mapping; (2) Content: Existing methods primarily store raw geometric structures that lack **task-specific semantics**, while our method enhances policies via **task-oriented experience and procedural reasoning**; (3) Generalization: Such memory systems are inherently **limited to revisiting the same environments**, whereas EvoNav **generalizes effectively to unseen environments**. Overall, by shifting from scene-specific geometry to generalizable semantic and procedural knowledge, our approach offers stronger generalization, richer decision reuse, and higher scalability.

#### C.2. Experience: Traditional Offline vs our Online

For the obtained experience, we use chroma (a vector database enabling scalable storage and retrieval of embeddings) to store it in the experience bank. Note that instead of traditional RAG [2, 7, 11] workflows that primarily build an **offline database**, where *it first executes all navigation episodes in advance to collect the experience bank, and then*

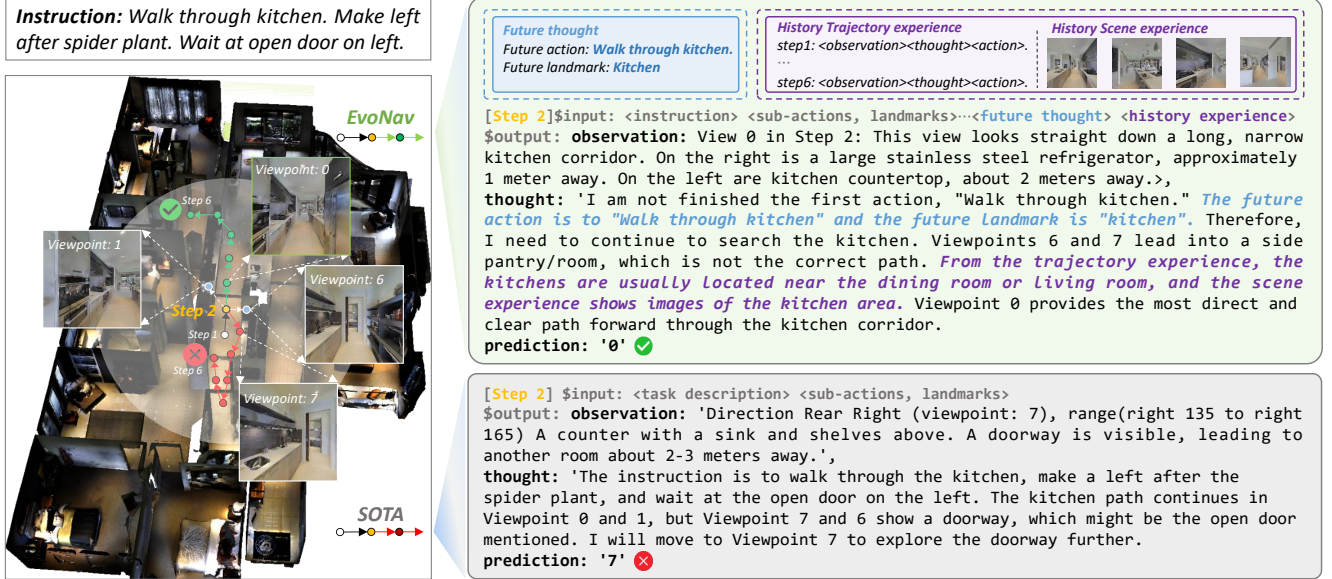


Figure 1. Navigation demo in simulated environment on R2R-CE. The left denotes navigation trajectories between the SOTA method (Open-Nav [9]) and our EvoNav. For a better view, we also provide the navigation decision-making details of the key step, which highlight the effect of the proposed future thoughts and historical experience.

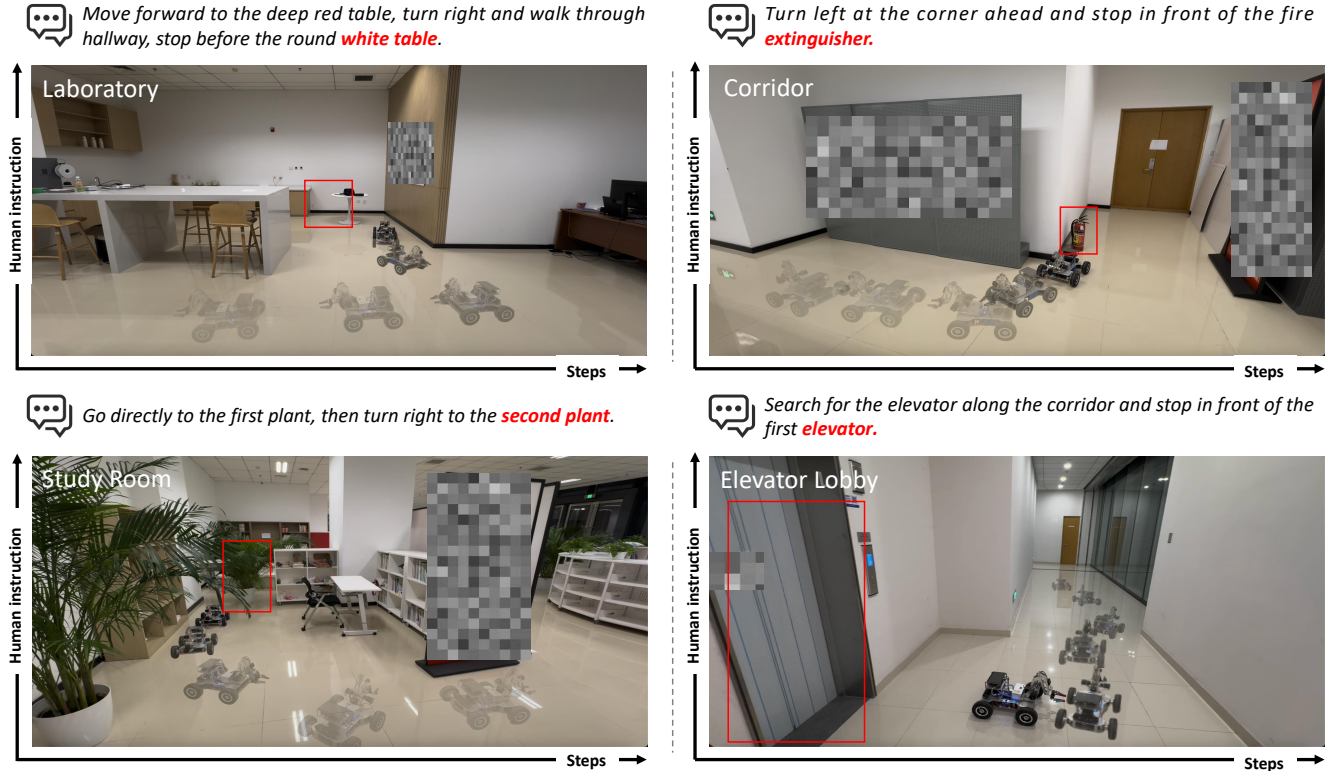


Figure 2. Navigation demo in the real world. Give the instruction, EvoNav can effectively navigate to the target place (marked in red).

uses the experience bank to assist navigation evaluation. This paradigm, i.e., first utilizes all episodes to construct the experience bank offline and then uses it for navigation, not only expanding navigation time but also increasing the

risk of data leaks. In contrast, we simultaneously collect **online experience** and perform navigation, updating the experience bank in real-time as navigation executes. This enhances data reliability and experience flexibility.

'system': "You are a navigation robot that navigates in the real world. You need to understand visual environment and follow instruction to move in an indoor environment with the least action steps, and ultimately find the destination to stop. At each navigation step, I will give you one global instruction, a series of images of different directions of the environment, and information about landmarks. \

In addition, I will also give you navigation history, estimation of executed actions, future landmark, and a few-shot navigation text and image experience containing similar landmarks for reference. \

The "\future landmark\" indicates a landmark prediction of the next step. \

The "\few-shot navigation text experience\" refers to memories of past navigation, containing the decision-making processes of similar tasks in the past. You can refer to it for decision-making. \

The "\few-shot navigation image experience\" refers to refer to a few images similar to the current landmark. These images match your current environments and help you accurately understand the landmark objects within your environment. \

few-shot text and image experience. You can refer to it for decision-making. \

You can observe the visual environment through provided visual images from multiple directional viewpoints around you. \

Each direction images contains direction viewpoint ids you can move to. Your task is to predict moving to which direction viewpoint. \

Your answer includes three parts: "\Observation\", "\Thought\", and "\Prediction\". \

First, in the "\Observation\", your output must be a Python-style dictionary. Each key corresponds to a Viewpoint ID and maps to a description paragraph for that viewpoint image. \

The description must indicate both the Viewpoint ID and the current Step ID in the following forma

Follow strictly this format for output: \

Observation: {\\"0\": \"Direction Viewpoint ID: 0 in Step ID 0: ...\", \\"1\": \"Direction Viewpoint ID: 1 in Step ID 0: ...\", \\"2\": \"Direction Viewpoint ID: 2 in Step ID 0: ...\"} \

For each value in the dictionary: \

- (1) List all recognizable objects in the image. \
- (2) According to the results of (1), identify the object located approximately at the center of the image. Then, describe what objects appear to its front, back, left, and right (if any). \
- (3) Estimate how far these objects are from the camera, and provide the results in approximate meters. \

\n \

Then, in the "\Thought\", you should think as detailed as possible following procedures: \

- (1) The viewpoint ID you predicted must be one of the Direction Viewpoint ID in Candidate Viewpoint IDs List. The Candidate Viewpoint IDs List show the Direction Viewpoint ID that you should go. This means that there should be only a number after "\Prediction\" without any other words or characters. \
- (2) Check whether the latest executed action has been completed by comparing current environment and landmark in the latest executed action. \
- (3) Determine the action you should execute and landmark you should reach now. If the latest executed action have not been completed, you should continue to execute it. Otherwise, you should execute the next action in the given instruction. \

Analyze which direction in the current environment is most suitable to execute the action you decide and explain your reason. \

- (4) Your thoughts need to refer to the information in "\future landmark\". \
- (5) Your thoughts also need to refer to the information in "\few-shot navigation text and image experience\". \
- (6) Predict moving to which direction viewpoint based on your thought process. \
- (7) The "\Thought\" you predicted should be a single paragraph. \
- (8) If you believe you have completed the instruction, you must still strictly follow the requirements to predict the next viewpoint in the "\Prediction\". \
- (9) If you want to make a left turn, you usually need to select a viewpoint ID between 1 and 5. If you want to make a right turn, you usually need to select a viewpoint ID between 7 and 11. However, the viewpoint ID you predict must be within the Current Environment. \
- (10) Your output after "\Prediction\" must be one of the number in Candidate Viewpoint IDs List without any other words. \

\n \

Last, in the "\Prediction\", please make decision on the next viewpoint. \

Your decision is very important, must make it very carefully. \

You need to double check the output in "\Prediction:". The output must be in the Candidate Viewpoint IDs without any other words. \

You also need to double check the output in "\Thought\". The output must be a single paragraph.",

'user': "Candidate Viewpoint IDs List: [{]} Step ID {} Instruction: {} Actions: {} Landmarks: {} Navigation History: {} \

Estimation of Executed Actions: {} Future landmark: {} Few-shot navigation text Experience: {} -> Observation: ... Thought: ... Prediction: ... \

You should simplify observation description as short and clear as possible. Your output after "\Observation\" MUST be a single paragraph. \

You should simplify navigation thought process as short and clear as possible. You ONLY need to summarize the what actions you did and what landmarks you passed in "\Thought\" using ONLY a single paragraph. Do NOT include Direction information. \n \

Your output after "\Prediction\" must be one of the number in Candidate Viewpoint IDs List without any other words."

Figure 3. Action prediction prompt for LLMs. We here provide the complete prompt, including user and system prompt, to demonstrate the navigation details about the proposed future thought and history experience, for a better view.

### C.3. Evaluation protocol and GT trajectory

To clarify, **EvoNav strictly adheres to the training-free zero-shot protocol**, utilizing only the agent’s own historical path ( $\mathcal{H}^*$ ) and post-hoc metrics (SR/SPL) rather than simulator ground truth (GT). The “corrected process” is generated via **LLM Self-Reflection**, where the model critiques its **own trajectory** against the original instructions to “re-imagine” a more logical navigational sequence. While these

imagined paths may not perfectly align with the GT, they function as critical experiences to diagnose reasoning flaws and refine future decision reliability. This experience-driven mechanism simulates human-like reflection on failure **without external supervision**, ensuring a fair comparison with Open-Nav by strictly avoiding **training, GT storage, or GT trajectory**.

## References

- [1] Angel X. Chang, Angela Dai, Thomas A. Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from RGB-D data in indoor environments. In *3DV*, pages 667–676, 2017. [1](#)
- [2] Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 17754–17762, 2024. [1](#)
- [3] Jiaqi Chen et al. Mapgpt: Map-guided prompting for unified vision-and-language navigation. In *ACL*, pages 9796–9810, 2024. [1](#)
- [4] Kehan Chen, Dong An, Yan Huang, Xu, and et al. Constraint-aware zero-shot vision-language navigation in continuous environments. *IEEE TPAMI*, 47(11):10441–10456, 2025. [1](#)
- [5] Muhammad Fadhil Ginting, Dong-Ki Kim, Xiangyun Meng, Andrzej Reinke, Bandi Jai Krishna, Navid Kayhani, Oriana Peltzer, David D Fan, Amirreza Shaban, Sung-Kyun Kim, et al. Enter the mind palace: Reasoning and planning for long-term active embodied question answering. *arXiv preprint arXiv:2507.12846*, 2025. [1](#)
- [6] Mukul Khanna, Ram Ramrakhya, Gunjan Chhablani, Sriram Yenamandra, Theophile Gervet, Matthew Chang, Zsolt Kira, Devendra Singh Chaplot, Dhruv Batra, and Roozbeh Motlaghi. Goat-bench: A benchmark for multi-modal lifelong navigation. In *CVPR*, pages 16373–16383, 2024. [1](#)
- [7] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020. [1](#)
- [8] Rui Liu, Wenguan Wang, and Yi Yang. Vision-language navigation with energy-based policy. *Advances in Neural Information Processing Systems*, 37:108208–108230, 2024. [1](#)
- [9] Yanyuan Qiao, Wenqi Lyu, Hui Wang, Zixu Wang, Zerui Li, Yuan Zhang, Mingui Tan, and Qi Wu. Open-nav: Exploring zero-shot vision-and-language navigation in continuous environment with open-source llms. In *ICRA*, pages 6710–6717, 2025. [1](#), [2](#)
- [10] Santhosh K Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alex Clegg, John 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. *arXiv preprint arXiv:2109.08238*, 2021. [1](#)
- [11] Alireza Salemi and Hamed Zamani. Evaluating retrieval quality in retrieval-augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2395–2400, 2024. [1](#)
- [12] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019. [1](#)
- [13] Xiangyu Shi, Zerui Li, Wenqi Lyu, Jiatong Xia, Feras Dayoub, Yanyuan Qiao, and Qi Wu. Smartway: Enhanced way-point prediction and backtracking for zero-shot vision-and-language navigation. In *IROS*, pages 16923–16930, 2025. [1](#)
- [14] Xiangyu Shi, Zerui Li, Yanyuan Qiao, and Qi Wu. Fast-smartway: Panoramic-free end-to-end zero-shot vision-and-language navigation. *arXiv preprint arXiv:2511.00933*, 2025. [1](#)
- [15] 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 *NeurIPS*, pages 251–266, 2021. [1](#)
- [16] Zihan Wang, Seungjun Lee, and Gim Hee Lee. Dynam3d: Dynamic layered 3d tokens empower vlm for vision-and-language navigation. *arXiv preprint arXiv:2505.11383*, 2025. [1](#)
- [17] Zihan Wang, Yaohui Zhu, Gim Hee Lee, and Yachun Fan. Navrag: Generating user demand instructions for embodied navigation through retrieval-augmented llm. *arXiv preprint arXiv:2502.11142*, 2025. [1](#)
- [18] Hang Yin, Haoyu Wei, Xiuwei Xu, Wenxuan Guo, Jie Zhou, and Jiwen Lu. Gc-vln: Instruction as graph constraints for training-free vision-and-language navigation. *arXiv preprint arXiv:2509.10454*, 2025. [1](#)
- [19] 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. [1](#)
- [20] Qi Zheng, Daqing Liu, Chaoyue Wang, Jing Zhang, Dadong Wang, and Dacheng Tao. Esceme: Vision-and-language navigation with episodic scene memory. *IJCV*, 133(1):254–274, 2025. [1](#)