Trajectory-aware Success and Failure Experience Retrieval for VLM Agents

Somin Lee Jinsik Bang Taehwan Kim
UNIST
{komin, bang, taehwankim}@unist.ac.kr

1. Introduction

Recently, the development of embodied agents that interact with environments and handle tasks in open-world settings using large language models (LLMs) [8, 11, 15] and vision-language models (VLMs) [1, 3, 5] has rapidly advanced. VLMs have emerged as a significant research direction by unifying perception and action, enabling complex embodied tasks in diverse environments under zero-shot settings [16]. However, in embodied tasks, VLM agents still struggle to reuse past interaction experiences for more accurate planning in unseen environments. Existing studies have mainly focused on using memory from successful tasks [6, 7, 19] or analyzing failure cases to identify failure reasons and generate corrective lessons [4, 17, 18]. However, identifying the exact failure point and underlying cause of failed trajectories remains challenging, leading these approaches to depend on ground-truth pairs or synthesizing failures intentionally. Moreover, memory based approaches commonly retrieve experiences based on task instructions [12, 13]. This instruction-level retrieval may limit their ability to select experiences aligned with the current execution trajectory, as identical instructions can correspond to different scene-specific situations observed from the surrounding environment. To address these issues, we propose **Trajectory-aware Experience Retrieval for VLM Agents (TERA)**, a framework that dynamically retrieves success and failure experiences according to the current execution trajectory. TERA collects success and failure execution experiences, summarizes them based on execution flows, and stores them in memory as experience rules for retrieval. During replanning, TERA retrieves relevant experience rules using trajectory-aware similarity and incorporates them into the VLM prompt. This allows success and failure experiences to act complementarily, providing structural guidance while suppressing repetitive mistakes from similar execution contexts. In summary, our contributions are as follows:

- We propose TERA, which dynamically retrieves memory at each replanning step based on the current trajectory.
- Our framework generates experience rules from text and image grounded trajectories, enabling VLMs to infer success/failure and failure reasons for contrastive replanning.
- Our results show the complementary synergy of diverse, unpaired success and failure experiences, leading to overall zero-shot performance gains on EB-Manipulation and EB-Habitat [16].

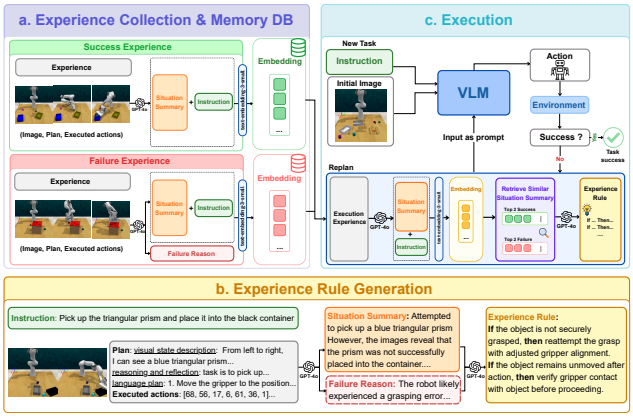

Figure 1. Overview of the proposed TERA framework.

2. Method

2.1. Execution Experience Collection

The VLM interacts with the environment in EmbodiedBench [16]. Each execution experience is represented as $m = (L, I_{0:t}, v_{0:t}, r_{0:t}, p_{0:t}, a_{0:t})$, where L denotes the language instruction, I_t the visual observation image, v_t the visual state description, r_t the reasoning and reflection, p_t the language plan, and a_t the executed action at step t, using information provided by EmbodiedBench [16]. The collected success and failure experiences are transformed into situation summaries that capture execution flows and key success or failure actions rather than precise coordinates, as Fig. 1 (a). The summarized memory is represented as $\widetilde{m}_i = (L_i, T_i, o_i, s_i, f_i, e_i)$, where L_i denotes the instruction, T_i the task type, o_i success/failure outcome, s_i the situation summary, f_i the failure reason for failed executions, and e_i the embedding generated using OpenAI text-embedding-3-small [9] for the i-th execution experience.

2.2. Experience Rule Generation

Based on the generated situation summaries s_i, the framework further constructs generalized experience rules using a

Table 1. The quantitative results on EB-Manipulation and EB-Habitat [16]. We note that Qwen2.5-VL-7B-Ins [2] is not evaluated in EB-Manipulation due to template error for executable plan generation in EmbodiedBench [16].

Model	EB-Manipulation				EB-Habitat			
	Base	Success	Failure	All	Base	Success	Failure	All
GPT-4.1 [11]	31.2	37.5	37.5	41.6	90.0	88.0	90.0	92.0
InternVL3-78B-4bit [21]	14.6	10.4	16.6	25.0	78.0	90.0	92.0	92.0
InternVL3-8B [21]	14.6	10.4	14.6	25.0	64.0	58.0	58.0	60.0
Qwen2.5-VL-7B-Ins [2]	-	-	-	-	36.0	44.0	42.0	44.0

VLM. The generated rules follow an *if-then* format to provide reusable replanning guidance across different execution contexts. For successful experiences, the rules are generated using only the corresponding situation summaries. In contrast, failure experiences additionally utilize failure reasons f_i together with the situation summaries to generate rules that suppress repetitive mistakes and invalid actions.

2.3. Trajectory-aware Experience Retrieval

During embodied execution, the agent replans whenever the observed execution outcome differs from the generated plan. At each replanning stage, the framework summarizes the partial execution trajectory using the information collected up to step t—current execution as the execution experience m—producing an execution situation summary s_j^{exec}. The generated summary is then embedded using OpenAI text-embedding-3-small [9]. We compute cosine similarity between the current execution embedding and each stored memory embedding, and retrieve the experiences with the highest similarity:

$$\text{Sim}(e_j^{\text{exec}}, e_i) = \frac{e_j^{\text{exec}} \cdot e_i}{\|e_j^{\text{exec}}\| \|e_i\|} \quad (1)$$

where e_j^{exec} denotes the execution-summary embedding at replanning stage j, and i indexes stored experiences. Using the retrieved experiences, the corresponding success and failure experience rules generated in Sec. 2.2 are incorporated into the VLM prompt for replanning. Since the execution trajectory continuously changes during interaction, the generated execution summary and retrieved experiences dynamically change at each replanning stage, enabling adaptive planning based on the current execution context.

3. Experiments

Experimental Setup. To evaluate our approach, we utilize EmbodiedBench [16], focusing on the Manipulation [20] and Habitat [14] environments with the Base subset to assess precise interaction and spatial reasoning capabilities. For memory construction, we collect embodied trajectories using multiple VLMs, including Qwen2.5-VL-7B-Ins [2], InternVL3 variants (8B and 78B-4bit) [21], and GPT-4.1 [11]. Through environment interactions, we obtain total 766 manipulation and 509 habitat execution experiences. GPT-4o [10] is then used to convert these execution experiences into *situation summaries* and generalized *experience rules*. During experience retrieval, memories originating from the identical episode are excluded to avoid information leakage. For embodied task execution, we evaluate three settings: (1) *Success only* (up to 2 success memories), (2) *Failure only* (up to 2 failure memories), and (3) *Success & Failure (All)* (up to 2 of each).

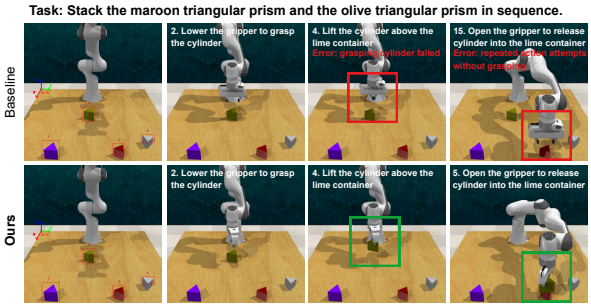

Figure 2. The qualitative results on the EB-Manipulation [16]. Both the baseline and our method utilize GPT-4.1.

Results. Table 1 presents the results of baselines and the results of adding various memory input prompts to VLMs. In EB-Manipulation, our framework demonstrates consistent performance improvements when utilizing both success and failure memories (**All**). GPT-4.1, InternVL3-78B-4bit, and InternVL3-8B each achieve a 10.4% improvement. In contrast, using only success or failure experiences occasionally leads to limited improvements or unstable behaviors, suggesting that one-sided experiences alone are insufficient for robust replanning. These results indicate that success memory provides execution structure, while failure experiences help avoid repetitive errors observed in similar execution situations. Additionally, as shown in Figure 2, our framework reduces planning steps and successfully completes tasks following the given instructions.

In EB-Habitat, our framework also shows performance improvements, with InternVL3-78B-4bit and Qwen2.5-VL-7B-Ins improving by 14.0% and 8.0%, respectively. In contrast, InternVL3-8B performs worse than the baseline. This result suggests that, together with the low success rate of Qwen2.5-VL-7B-Ins, VLMs with fewer parameters may have limited ability to infer useful summaries when navigation is involved, as in Habitat. In Qwen2.5-VL-7B-Ins, however, instruction tuning appears to partially compensate for this limitation. Overall, unlike EB-Manipulation, both the success, failure-only settings improve performance in most cases, while contrastive experience understanding further helps generate accurate plans and actions.

4. Conclusion and Future Work

In this study, we propose TERA, which converts text-image grounded trajectories into success, failure experience rules and retrieves them for trajectory-aware replanning. Our experiments show overall improvements, where success memories provide structural guidance while failure memories help reduce repetitive errors. Future work will extend TERA to vision-language action model for continuous environments and navigation problem.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, 2022. 1

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025. 2

[3] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. PaLM-E: An embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, 2023. 1

[4] Jiafei Duan, Wilbert Pumacay, Nishanth Kumar, Yi Ru Wang, Shulin Tian, Wentao Yuan, Ranjay Krishna, Dieter Fox, Ajay Mandlekar, and Yijie Guo. AHA: A vision-language-model for detecting and reasoning over failures in robotic manipulation. In *International Conference on Learning Representations*, 2025. 1

[5] Linxi Fan, Guanzhi Wang, Yunfan Jiang, Ajay Mandlekar, Yuncong Yang, Haoyi Zhu, Andrew Tang, De-An Huang, Yuke Zhu, and Anima Anandkumar. Minedojo: Building open-ended embodied agents with internet-scale knowledge. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 1

[6] Runnan Fang, Yuan Liang, Xiaobin Wang, Jialong Wu, Shuofei Qiao, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. MemP: Exploring agent procedural memory. *arXiv preprint arXiv:2508.06433*, 2025. 1

[7] Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. VIMA: General robot manipulation with multimodal prompts. In *Proceedings of the 40th International Conference on Machine Learning*, 2023. 1

[8] Bang Liu, Xinfeng Li, Jiayi Zhang, Jinlin Wang, Tanjin He, Sirui Hong, Hongzhang Liu, Shaokun Zhang, Kaitao Song, Kunlun Zhu, et al. Advances and challenges in foundation agents: From brain-inspired intelligence to evolutionary, collaborative, and safe systems. *arXiv preprint arXiv:2504.01990*, 2025. 1

[9] OpenAI. New embedding models and api updates. https://openai.com/index/new-embedding-models-and-api-updates/, 2024. 1, 2

[10] OpenAI. Hello GPT-4o. https://openai.com/index/hello-gpt-4o/, 2024. Accessed: 2026-05-12. 2

[11] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1, 2

[12] Siru Ouyang, Jun Yan, I-Hung Hsu, Yanfei Chen, Ke Jiang, Zifeng Wang, Rujun Han, Long T. Le, Samira Daruki, Xiangru Tang, Vishy Tirumalashetty, George Lee, Mahsan Rofouei, Hangfei Lin, Jiawei Han, Chen-Yu Lee, and Tomas Pfister. ReasoningBank: Scaling agent self-evolving with reasoning memory. In *International Conference on Learning Representations*, 2026. 1

[13] Wangtao Sun, Xuanqing Yu, Shizhu He, Jun Zhao, and Kang Liu. Expnote: Black-box large language models are better task solvers with experience notebook. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023. 1

[14] Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. In *Advances in Neural Information Processing Systems*, 2021. 2

[15] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Ji-Rong Wen. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 2024. 1

[16] Rui Yang, Hanyang Chen, Junyu Zhang, Mark Zhao, Cheng Qian, Kangrui Wang, Qineng Wang, Teja Venkat Koripella, Marziyeh Movahedi, Manling Li, Heng Ji, Huan Zhang, and Tong Zhang. EmbodiedBench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025. 1, 2

[17] Zewei Ye et al. Robofac: A comprehensive framework for robotic failure analysis and correction. *arXiv preprint arXiv:2505.12224*, 2025. 1

[18] Tianjun Zhang, Aman Madaan, Luyu Gao, Steven Zheng, Swaroop Mishra, Yiming Yang, Niket Tandon, and Uri Alon. In-context principle learning from mistakes. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. 1

[19] Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. ExpeL: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024. 1

[20] Kaizhi Zheng, Xiaotong Chen, Odest Chadwicke Jenkins, and Xin Eric Wang. Vlmbench: A compositional benchmark for vision-and-language manipulation. In *Advances in Neural Information Processing Systems*, 2022. 2

[21] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 2