Do VLMs Know What to Do After Failed Actions? A Diagnostic Probe for Embodied Error Recovery

Tianxin Huang[1] Jinrui Mai[2] Jiayu Chen[2]
[1]McGill University [2]The University of Hong Kong
tianxin.huang@mail.mcgill.ca mmm1441@connect.hku.hk jiayuc@hku.hk

Abstract

Embodied agents must not only execute instructions, but also recover after failed actions. We study whether vision-language models (VLMs) can diagnose what went wrong and choose the next corrective action under ambiguous before–after visual evidence. We introduce a compact AI2-THOR-based diagnostic probe of 240 manually constructed and audited post-failure examples across four failure mechanisms: wrong view, wrong object or unrelated action, unsatisfied precondition/order, and task-appropriate but ineffective execution. Each example requires two outputs: a failure category and a one-step recovery action. We evaluate three hosted VLM endpoints under four input conditions: full before–after context, after-only context, text-only context, and a temporal swap-test. The results reveal three diagnostic effects. First, models exhibit a diagnosis–action gap: failure-category prediction is often easier than recovery-action selection. Second, text-only performance exposes language-prior shortcuts in some wrong-object cases. Third, temporal swapping sharply degrades wrong-object recovery, showing that temporal direction matters for embodied error recovery.

1. Introduction

Embodied agents often fail locally before completing a task: they may face the wrong view, interact with a distractor, violate an ordering constraint, or perform a locally appropriate action that still produces no effective state change. Existing embodied benchmarks commonly emphasize final task success, trajectory quality, or goal completion [1, 2, 4, 6, 8]. These metrics are essential, but they can obscure the local post-failure decision: *what failed, and what should the agent do next?*

We propose a compact diagnostic probe for one-step embodied error recovery. Given an instruction, an attempted action, and condition-dependent visual evidence, a model predicts both a failure category and a one-step recovery action. Figure 1 shows a representative F2 example: the instruction is to heat bread in a pan on the stove, but the attempted action is `Pick(pot)`, so the correct local recovery is to put the distractor pot down before proceeding.

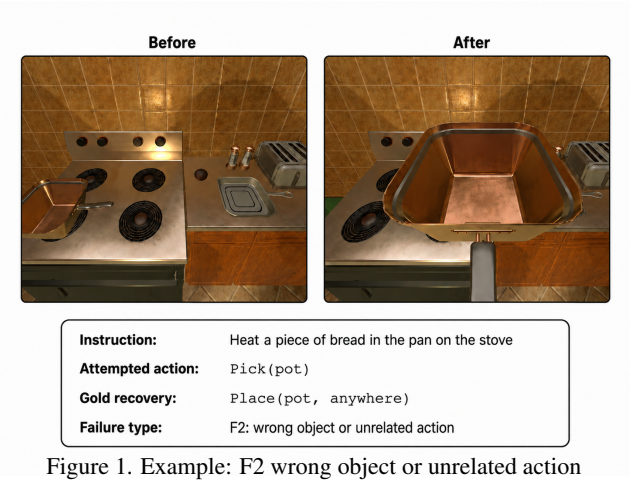

Figure 1. Example: F2 wrong object or unrelated action

Formally, the task maps an instruction I, attempted action a, and visual context V to a failure category c and recovery action r, i.e., $(I, a, V) \rightarrow (c, r)$. This decouples two abilities that are often conflated in end-to-end evaluation: broad failure diagnosis and concrete corrective-action selection. The distinction matters because a model may correctly identify the failure type while still choosing an unhelpful recovery action.

Our benchmark is built from AI2-THOR scenes and observations. Rather than using raw simulator logs as-is, we manually construct or correct post-failure scenarios by pairing an instruction, an attempted action, before/after observations, a failure label, and an audited one-step recovery action. Every retained example is manually audited for visual plausibility, instruction–action consistency, failure-category correctness, recovery-action correctness, and candidate-vocabulary validity. The final split contains 240 examples, balanced across four failure categories with 60 examples each. Prior benchmarks mainly focus on long-horizon task completion, instruction following, navigation,

or interactive reasoning [1, 3, 5, 7, 9–11]. In contrast, our compact probe isolates local post-failure recovery, decouples diagnosis from action selection, and emphasizes controlled perturbations rather than leaderboard-style ranking.

2. Benchmark and Protocol

Failure taxonomy. We use four recovery-relevant failure categories: **F1** target not visible or wrong view; **F2** wrong object or unrelated action; **F3** target visible but precondition/order unsatisfied; and **F4** task-appropriate but ineffective execution. The F2/F4 distinction is based on local alignment with the intended subgoal: F2 involves a semantically inconsistent action or target, whereas F4 involves a locally appropriate action whose observed effect is incomplete or insufficient.

Input conditions. All conditions use the same instruction, attempted action, taxonomy, action-space description, candidate vocabulary, and output schema. They differ only in visual evidence: **full** provides before/after images; **after-only** provides only the post-attempt image; **text-only** removes images; and **swap-test** uses the same prompt template but swaps the order of the two visual inputs. Text-only probes language-and-vocabulary priors, while swap-test probes temporal-direction sensitivity. Text-only success indicates possible reliance on instruction–action mismatch or candidate-vocabulary priors rather than visual grounding. Full-over-after gains indicate use of visual transition evidence, while swap-test degradation indicates sensitivity to temporal direction.

Models and metrics. We evaluate GPT-4o, GPT-5.4, and GPT-4o-mini with temperature 0 and JSON-style outputs. We report Category Acc. and Action Acc. as exact match after light normalization and candidate-vocabulary validation. This conservative action metric improves reproducibility but may under-credit semantically valid alternatives. The split is balanced, so a uniform random category predictor obtains 0.25 accuracy.

3. Results

Table 1 reports overall results. The ablations reveal that better diagnosis does not necessarily imply better recovery-action selection. For example, GPT-5.4 obtains the highest full-condition category accuracy, but GPT-4o obtains the highest full-condition action accuracy. Across models and conditions, category accuracy is generally higher than action accuracy, indicating a diagnosis–action gap.

Temporal direction matters. The strongest diagnostic effect appears in F2 wrong-object recovery. For GPT-4o, F2 action accuracy drops from 0.633 in the full condition to 0.017 under swap-test; GPT-5.4 drops to 0.000. This suggests that wrong-object recovery depends strongly on before/after temporal direction rather than static scene content alone.

Table 1. Overall results. Each cell reports Category Acc. / Action Acc.. GPT-4o-mini is computed over valid rows only.

Model	Full	After	Text	Swap
GPT-4o	.758/.708	.713/.579	.592/.500	.738/.529
GPT-5.4	**.792**/.663	**.746**/.513	**.729**/.363	**.788**/.517
GPT-4o-mini	.491/.307	.363/.238	.364/.123	.459/.217

Text-only inputs expose language shortcuts. Text-only performance is highly class-dependent. For GPT-4o, text-only F2 action accuracy is 0.767, while text-only F3 action accuracy is only 0.167. Thus, some wrong-object cases can be inferred from instruction–action mismatch and candidate-vocabulary priors, whereas precondition/order failures require stronger visual grounding.

Retry failures are visually subtle. F4 examples are locally task-appropriate but ineffective, so the expected recovery is usually to retry the attempted action type. GPT-4o's F4 action accuracy drops to 0.183 in the after-only condition, suggesting that the post-attempt image alone often fails to reveal whether the local action was ineffective.

Model comparison reveals a diagnosis–action gap. GPT-5.4 obtains stronger category accuracy than GPT-4o across all four input conditions, but this advantage does not consistently transfer to recovery-action selection. In the full condition, GPT-5.4 reaches 0.792 category accuracy, whereas GPT-4o obtains the highest action accuracy at 0.708. This pattern suggests that stronger failure-type recognition or linguistic consistency may still leave open the harder embodied question of which local corrective action should be taken next. In this sense, GPT-4o's stronger action accuracy on several recovery cases is not an anomaly, but evidence that diagnosis and action selection should be evaluated separately.

4. Discussion

Within this compact diagnostic setting, our results show that local recovery should be evaluated separately from broad failure diagnosis. Models may know what failed but still choose the wrong corrective action; text-only success can expose language-prior shortcuts; and swap-test degradation shows that temporal direction matters. These findings position post-failure recovery as a distinct diagnostic target for embodied VLMs. Code, prompts, and evaluation outputs will be released.

References

[1] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2

[2] Angel X. Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *International Conference on 3D Vision*, 2017. 1

[3] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, Jose M.F. Moura, Devi Parikh Lee, and Dhruv Batra. Embodied question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–10, 2018. 2

[4] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017. 1

[5] Jaewon Lee, Jaeseok Heo, Gunmin Lee, Howoong Jun, Jeongwoo Oh, and Songhwai Oh. Rvn-bench: A benchmark for reactive visual navigation. *arXiv preprint arXiv:2603.03953*, 2026. 2

[6] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019. 1

[7] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Devi Parikh, and Dhruv Batra. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019. 2

[8] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 1

[9] Sanjana Srivastava, Chengshu Li, Michael Lingelbach, Roberto Martín-Martín, Fei Xia, Kent Vainio, Silvio Savarese, et al. Behavior: Benchmark for everyday household activities in virtual, interactive, and ecological environments. *arXiv preprint arXiv:2108.03332*, 2021. 2

[10] Fei Xia, William B. Shen, Chengshu Li, Priya Kasimbeg, Micael Edmond Tchapmi, Alexander Toshev, Roberto Martín-Martín, and Silvio Savarese. Interactive gibson benchmark: A benchmark for interactive navigation in cluttered environments. *IEEE Robotics and Automation Letters*, 5(2):713–720, 2020.

[11] Ninghan Zhong, Steven Caro, Avraiem Iskandar, Megnath Ramesh, and Stephen L. Smith. Bench-npin: Benchmarking non-prehensile interactive navigation. *arXiv preprint arXiv:2505.12084*, 2025. 2

A. Appendix

A.1. Dataset Construction and Audit Protocol

All examples are built from AI2-THOR scenes and visual observations. Failure cases are manually constructed or corrected rather than directly sampled from raw simulator logs. Each retained example is checked for the following criteria:

- **Instruction–action consistency:** the attempted action is interpretable with respect to the instruction.
- **Visual plausibility:** the before/after observations are visually consistent with the intended failure scenario.
- **Failure-category validity:** the assigned F1–F4 category matches the local failure mechanism.
- **Recovery-action validity:** the gold one-step recovery action is locally appropriate for the audited state.
- **Candidate-vocabulary validity:** all gold action arguments appear in the candidate vocabulary, while gold labels and diagnostic notes are excluded from the prompt.

A.2. Failure Taxonomy Details

F1: Target not visible / wrong view. The agent cannot currently observe the relevant target or is facing an unhelpful view. The expected recovery often involves `LookAround()` or navigation to a relevant target.

F2: Wrong object or unrelated action. The attempted action targets an object or action that is semantically inconsistent with the instruction. In many F2 cases, the recovery first clears an incorrectly held distractor by placing it on an audited support surface.

F3: Precondition/order unsatisfied. The target may be visible, but a required precondition or ordering constraint is missing. For example, the model may need to pick an object before placing it, open a container before placing an object inside, or complete a prerequisite interaction first.

F4: Task-appropriate but ineffective execution. The attempted action type or target is locally appropriate, but the observed effect is incomplete or insufficient. The expected local recovery is usually `Retry(action_type)` rather than switching to a different subgoal.

A.3. Prompt Sketch

The released prompt contains the instruction, attempted action, failure taxonomy, action-space description, candidate vocabulary, and condition-dependent visual evidence. It asks the model to output a JSON object with:

```
{
  "failure_category": <1|2|3|4>,
  "recovery_action": "<one-step action>"
}
```

Gold labels, symbolic audit states, and diagnostic notes are not included in the model input.

A.4. Representative Examples

Table 2. Representative audited examples.

Class	Instruction	Attempt	Gold recovery
F1	Cut apple slice from fridge and place in sauce pan	`Navigate(fridge)`	`LookAround()`
F2	Put sliced bread in fridge	`Pick(bowl)`	`Place (bowl,table)`
F3	Place plate with cut apple slice in fridge	`Place(apple, plate)`	`Pick(apple)`
F4	Place plate with cut fruit slice in fridge	`Pick(plate)`	`Retry(Pick)`

A.5. Full Per-Class GPT-4o Results

Table 3. GPT-4o per-class action accuracy across input conditions.

Condition	F1	F2	F3	F4
Full	.950	.633	.633	.617
After-only	.967	.567	.600	.183
Text-only	.500	.767	.167	.567
Swap-test	.850	.017	.600	.650

A.6. Full-Condition Per-Class Action Accuracy

Table 4. Per-class action accuracy in the full input condition.

Model	F1	F2	F3	F4
GPT-4o	.950	.633	.633	.617
GPT-5.4	.883	.383	.750	.633
GPT-4o-mini	.772	.259	.085	.121

A.7. Limitations

The benchmark is intentionally compact and diagnostic rather than a broad leaderboard. It evaluates one-step recovery rather than long-horizon replanning. The closed recovery-action space makes exact-match scoring reproducible, but it may under-credit semantically reasonable alternative recoveries. The candidate vocabulary reduces parsing ambiguity but can introduce language-and-vocabulary shortcuts. Different failure mechanisms may require different construction procedures; therefore, we emphasize within-condition and within-class perturbation effects rather than absolute claims about intrinsic class difficulty. Future work should expand the benchmark, add open-source VLMs and human agreement analysis, replace single-label exact match with set-valued or simulator-validated recovery criteria, and test whether predicted recoveries improve closed-loop success in interactive simulation.