Failure-Aware Query Refinement for Reliable Open-Vocabulary Home-Robot Perception

Daun Jeong [*1] **Sohyeon Kim** [*1] **Jouwon Song** [2] **Kyeongbo Kong** [1]

Abstract

Home robots in open-ended indoor environments must localize user- or task-specified objects using imperfect perception tools. Frozen open-vocabulary detectors are attractive, but class-name queries can miss the target, produce weak detections, or fire on distractors. We study failure-aware query adaptation for a frozen detector, modifying only the text query at test time. We propose *Failure-Aware Multi-Agent Query Refinement*, which diagnoses detector responses into `GOOD`, `MISS`, `WEAK`, and `NOISY` states, routes failures to state-specific query agents, and re-evaluates candidates by re-running the detector. On 9,740 ODSR-IHS image–target pairs, our method improves YOLOE from 0.6513 to 0.7667 AP_{50} and from 0.6129 to 0.7504 Top-1 accuracy, with the largest gains in `WEAK` and `NOISY` cases.

1. Introduction

Home robots operate in open-ended indoor environments where users may ask them to find objects such as "the slippers next to the bed" or "the wire on the floor." In this setting, localization is tied to task reliability and safety: an embodied agent must judge whether its perception response is trustworthy before acting on it. (Yenamandra et al., 2023) Open-vocabulary (OV) (Cheng et al., 2024; Liu et al., 2024; Li et al., 2022; Minderer et al., 2022) detectors appear suitable because targets can be specified with text queries. However, open vocabulary does not guarantee reliable open-domain localization. Class-name queries that work on large-scale generic detection benchmarks such as COCO (Lin et al., 2014), LVIS (Gupta et al., 2019), and Objects365 (Shao et al., 2019) often become unreliable in home environments, where occlusion, small objects, household-specific appearances, and visually similar distractors can cause the detector to miss the target, produce only a weak response, or respond to irrelevant objects. (Bianchi et al., 2024; Wang et al., 2023)

This creates a safety-relevant perception problem, but not one that can be solved by simply asking a VLM to write a better phrase. Per-home retraining is impractical, and lightweight OV detectors such as YOLOE (Wang et al., 2025; Sapkota & Karkee, 2026) do not necessarily parse long relational or attribute-rich expressions as intended. Moreover, unconstrained query rewriting can introduce query drift: a phrase may become more descriptive in natural language while becoming less effective for the detector or less faithful to the intended target. We therefore treat query selection as deployment-time adaptation for a frozen detector, where the goal is to find *detector-effective* queries that preserve target identity while producing more reliable detector responses.

A key challenge is that detector failures are not uniform. A missed target, a weak target-like detection, and a false-positive-heavy response require different recovery strategies. Generic VLM rewriting or iterative detector feedback may improve average performance, but they do not explicitly identify what kind of failure occurred. This matters for home-robot agents because recovering an absent response is different from strengthening a weak target response or suppressing distractors when the detector fires on the wrong object. We study query refinement as a failure-aware recovery problem rather than as unconstrained language generation.

In this paper, we propose **Failure-Aware Multi-Agent Query Refinement**, a test-time framework that makes detector query adaptation more diagnosable and failure-aware. A Plan Agent interprets the detector response, target evidence, visual context, and query history to diagnose the current failure mode. State-conditioned specialists then propose candidate queries for missed, weak, or false-positive-heavy responses. Each candidate is verified by re-running the frozen detector, and the best response is stored as history for subsequent refinement. We do not claim formal safety guarantees; instead, we focus on empirical recovery from hard perception failure modes that arise when home agents use OV detectors in cluttered indoor scenes.

[1]Pusan National University, Busan, Republic of Korea [2]LG Electronics, Seoul, Republic of Korea. Correspondence to: Kyeongbo Kong <kbkong@pusan.ac.kr>.

Published at the Second Workshop on Agents in the Wild: Safety, Security, and Beyond (AIWILD) at ICML 2026. Copyright 2026 by the author(s).

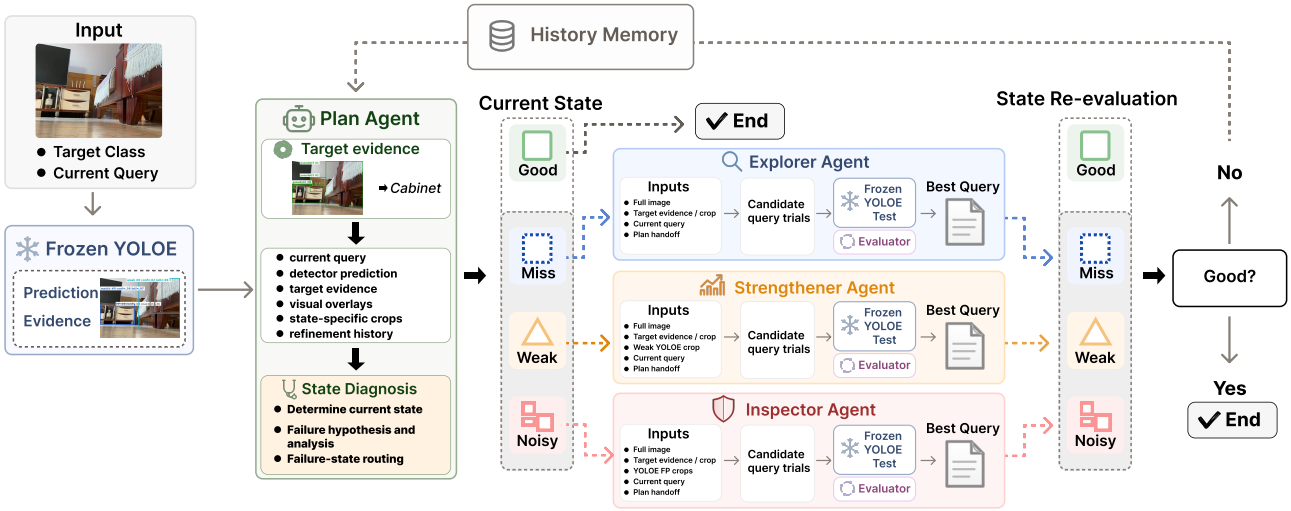

Figure 1. **Proposed failure-aware query refinement framework.** The Plan Agent diagnoses the frozen detector response into GOOD, MISS, WEAK, or NOISY states using prediction evidence and VLM-generated target evidence. Failure cases are routed to state-specific agents that propose refined queries, which are verified by re-running the frozen YOLOE detector. The loop iterates with history memory and state re-evaluation until a GOOD state is reached.

Our contributions are threefold:

- We formulate home-robot open-vocabulary grounding as failure-aware deployment-time query adaptation, where an embodied agent diagnoses unreliable detector responses before refinement.
- We introduce a failure-aware planner–specialist workflow that analyzes each failure case and uses dedicated specialists for query refinement.
- On ODSR-IHS (Lv et al., 2021) with 9,740 image–target pairs, our method improves YOLOE from 0.6513 to 0.7667 AP_{50} and from 0.6129 to 0.7504 Top-1 accuracy, with the clearest gains in hard failure modes such as `WEAK` and `NOISY`.

2. Method

We propose *Failure-Aware Multi-Agent Query Refinement*, a failure-aware test-time adaptation framework for using a frozen OV detector as a home-robot perception tool. Figure 1 illustrates the overall loop. Given an image, target class, and current query, frozen YOLOE first produces query-conditioned predictions. A Plan Agent then combines these predictions with target evidence and history memory to diagnose the detector state as `GOOD`, `MISS`, `WEAK`, or `NOISY`. `GOOD` terminates the loop; failure states are routed to specialist agents: Explorer for missed responses, Strengthener for weak responses, and Inspector for noisy false-positive-heavy responses. The selected specialist proposes candidate queries, which are verified by re-running the frozen detector and then re-evaluated by the same state-diagnosis loop until a `GOOD` state or the maximum step count is reached.

2.1. Problem Formulation

Each task instance is defined as an image–target pair. The system receives an image x, a target class c, and an initial query q_0, which is typically the class name itself. For user commands that contain additional context, such as "find the slippers next to the bed," we focus on the target-category localization component and use the extracted target class as the initial detector query. A frozen detector f_{det} takes an image–query pair (x, q) as input and returns a set of bounding boxes with confidence scores, denoted as $D(q) = f_{\text{det}}(x, q)$. The goal is to select, without updating f_{det}, a refined query q_T that better localizes the target in the current visual context. No human annotations are used during refinement. Target evidence denotes a VLM-generated noisy reference to the intended target's region and appearance, used only inside the refinement loop for state diagnosis and candidate comparison, while final AP50 and Top-1 evaluation uses held-out human annotations.

2.2. Failure-State Diagnosis

Failure states. Detector behavior for a given target class is heterogeneous: the detector may produce no usable response near the target, a weak target-like response, or a target response accompanied by many high-confidence false positives. We therefore define one success state, `GOOD`, and three failure states that trigger refinement: `MISS` (no reliable response near the target), `WEAK` (weak target-like response), and `NOISY` (target response with strong distractor responses). These states are used to diagnose the current detector response and to analyze how previous query attempts failed during refinement.

The *Plan Agent* predicts the routing state by jointly inspect-

Table 1. Failure states and routing. GOOD terminates the loop.

State	Diagnosis	Specialist	Goal
GOOD	Well-localized	—	End
MISS	No reliable target response	*Explorer*	Recover
WEAK	Weak target-like response	*Strengthener*	Boost
NOISY	Many high-confidence FPs	*Inspector*	Suppress

ing the current query, detector predictions, target evidence, visual overlays, state-specific crops, and refinement history. target evidence provides an internal reference for judging whether the detector response is near the intended target, while visual and historical evidence help explain ambiguous or repeated failures. The Plan Agent outputs a routing state and a compact handoff that summarizes the failure hypothesis, relevant history, and intended recovery direction, helping the selected specialist reason about the next query candidates. If no usable target evidence is available, the instance is not refined further and the system returns the current best query.

Specialist agents. Three specialists handle the three failure states. *Explorer* (MISS) handles cases where the detector produces no reliable response near the target. To recover detector response while preserving target identity, it inspects the weak target crop and the full image and proposes short class-preserving reformulations of the target query. *Strengthener* (WEAK) handles cases where a weak response already exists near the target. Instead of broadly rewriting the query, it compares the low-confidence detection crop with the target evidence and adds minimal visible cues that reinforce the detector's response on the same object.

The *Inspector* (NOISY) handles cases where target is detected together with high-confidence false positives. It contrasts the target crop with false-positive crops and adds target-positive cues that distinguish the intended target from nearby distractors, without using negative phrases such as "not" or "without." This is important because OV detectors may not parse negative language reliably; the query should describe what the target is, rather than what it is not.

All specialists are constrained to produce short detector-facing noun phrases rather than long natural-language descriptions. Candidate queries must preserve the target identity, either by retaining the original class token or by using a class-preserving lexical variant. The specialists only propose query hypotheses; they do not decide the final query. Final selection is performed by detector-in-the-loop state re-evaluation.

2.3. Candidate Verification and Iterative Refinement

Candidate queries proposed by a specialist are treated as hypotheses rather than final decisions. For each candidate $q' \in Q_t$, the evaluator re-runs the frozen detector and feeds the resulting response back to the same state-diagnosis pipeline. A candidate is considered successful if the re-evaluated state becomes GOOD, meaning that the detector response is sufficiently aligned with the current visual target evidence.

When multiple candidates are available, we select the one with the strongest verification signal. Visual target evidence is used only as an internal reference for comparing detector responses, not as ground truth or for final evaluation. If no candidate reaches GOOD, the system keeps the best-so-far query $q^\star$ and records failed attempts in the history.

The loop repeats with the selected or best-so-far query until a GOOD state is verified or the maximum iteration count is reached. The final output is $q^\star$ and its detector response. This does not certify correctness, but makes query adaptation failure-aware by requiring each VLM-proposed query to be checked through frozen-detector re-evaluation.

3. Experiments

3.1. Experimental Setup

Setup. We use YOLOE-26X (Sapkota & Karkee, 2026) as the frozen open-vocabulary detector and Qwen-VL-Plus (Bai et al., 2023) for target evidence generation and Plan/Specialist agent responses. Detector weights are fixed throughout refinement; only the text query is modified. All experiments are run on four NVIDIA RTX 3090 GPUs.

Benchmark and metrics. We evaluate on ODSR-IHS (Lv et al., 2021), a near-ground home-robot object detection benchmark. Following Appendix A.2, we use 1,000 images for auxiliary query analysis and report final results on 9,740 held-out image–target pairs from the remaining 4,993 annotated images. Human annotations are used only for final evaluation, not during refinement. We report AP_{50} and Top-1 localization accuracy, where Top-1 checks whether the highest-confidence prediction localizes the target.

3.2. Main Results

Table 2 compares query refinement baselines with different levels of detector-response diagnosis. The class-name YOLOE baseline achieves 0.6513 AP@0.5 and 0.6129 Top-1 accuracy, showing that static class-name queries are unreliable in cluttered home-robot scenes. Simply adding detector feedback to an Iterative VLM Agent gives only a small improvement, reaching 0.6678 AP@0.5 and 0.6304 Top-1 accuracy. This suggests that repeated generic rewriting is not sufficient for reliable query adaptation.

The Coarse-State VLM Agent improves substantially by separating GOOD and NON−GOOD cases, reaching 0.7569

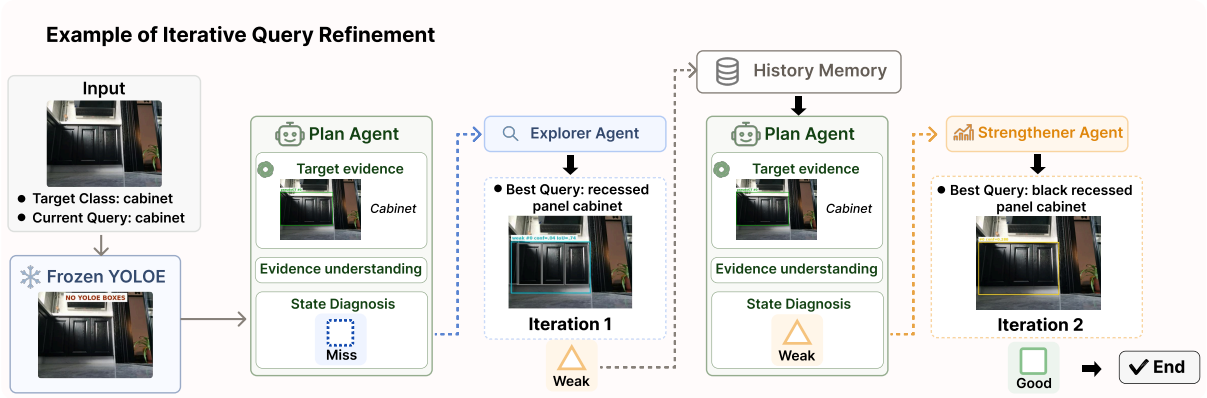

Figure 2. **Qualitative example of iterative query refinement.** Starting from the initial query "cabinet," the frozen YOLOE detector fails to produce a reliable localization, and the Plan Agent diagnoses the state as `MISS`. The Explorer proposes "recessed panel cabinet," which recovers the target but remains `WEAK`. After state re-evaluation, the Strengthener refines the query to "black recessed panel cabinet," producing a confident localization and a final `GOOD` state.

Table 2. **Overall comparison with query refinement baselines.** Diag. denotes the diagnosis exposed to the agent; SR denotes specialist routing; DF denotes detector feedback. All methods are evaluated using held-out human annotations; target evidence is used only inside the refinement loop and not for final scoring.

Method	Diag.	SR	DF	AP@0.5↑	Top-1↑
Class-name YOLOE	None	No	No	0.6513	0.6129
Iterative VLM Agent	None	No	Yes	0.6678	0.6304
Coarse-State VLM Agent	G/NG	No	Yes	0.7569	0.7319
Ours	**G/M/W/N**	**Yes**	**Yes**	**0.7667**	**0.7504**

AP@0.5 and 0.7319 Top-1 accuracy. Our full method achieves the best overall performance, with 0.7667 AP@0.5 and 0.7504 Top-1 accuracy. The gain over the coarse-state baseline is modest in AP but clearer in Top-1, indicating that fine-grained `GOOD`/`MISS`/`WEAK`/`NOISY` diagnosis with specialist routing helps select queries that better localize the intended target.

3.3. Analysis by Initial Failure State

Table 3 breaks down performance by the initial detector failure state. In `MISS` cases, both Coarse-State VLM and Ours recover many failures, suggesting that broad query reformulation is often enough when the detector initially produces no usable response. The key difference appears in `WEAK` and `NOISY` cases, where the detector already responds but remains unreliable.

In `WEAK` cases, Ours improves over Coarse-State VLM from 0.6954 / 0.5785 to 0.7435 / 0.6828. In `NOISY` cases, Ours improves from 0.6828 / 0.6906 to 0.7184 / 0.7812. These results support our main claim: detector feedback alone is not enough, and coarse diagnosis recovers many easy failures, but fine-grained failure diagnosis is most useful for hard cases that require confidence strengthening or distractor-aware disambiguation.

Table 3. **State-wise performance by initial failure state.** Initial states are determined from the class-name YOLOE response. Each cell reports AP@0.5 / Top-1 accuracy for samples belonging to that initial state.

Method	Diagnosis	Initial Failure State		
		MISS (AP / Top-1)	WEAK (AP / Top-1)	NOISY (AP / Top-1)
Class-name YOLOE	None	0.0951 / 0.0952	0.4738 / 0.2398	0.5821 / 0.4974
Iterative VLM Agent	None	0.3222 / 0.2928	0.5077 / 0.3422	0.6375 / 0.6000
Coarse-State VLM Agent	G/NG	0.5960 / **0.5593**	0.6954 / 0.5785	0.6828 / 0.6906
Ours	**G/M/W/N**	**0.5984** / 0.5545	**0.7435 / 0.6828**	**0.7184 / 0.7812**

3.4. Qualitative Refinement Trace

Figure 2 shows an iterative refinement example. Starting from the query "cabinet," the frozen YOLOE detector fails to localize the target, and the Plan Agent diagnoses `MISS`. The Explorer proposes "recessed panel cabinet," which recovers the target but remains `WEAK`. After state re-evaluation, the Strengthener refines the query to "black recessed panel cabinet," producing a confident localization and final `GOOD` state. This illustrates how the framework applies state-specific recovery rather than generic query rewriting.

4. Conclusion

We presented *Failure-Aware Multi-Agent Query Refinement*, a failure-aware test-time framework for frozen open-vocabulary detectors in home-robot scenes. Our method diagnoses detector failures into `MISS`, `WEAK`, and `NOISY` states, routes them to state-specific query agents, and verifies refinements through detector re-evaluation. Experiments on ODSR-IHS show that detector feedback alone is limited, while fine-grained diagnosis with specialist routing improves localization, especially in `WEAK` and `NOISY` cases.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. RS-2024-00456152). This work was also supported by LG Electronics. The authors gratefully acknowledge the Cluster Server for Computational Science at Pusan National University for providing computational resources.

References

Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., and Zhou, J. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.

Bianchi, L., Carrara, F., Messina, N., Gennaro, C., and Falchi, F. The devil is in the fine-grained details: Evaluating open-vocabulary object detectors for fine-grained understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22520–22529, 2024.

Cheng, T., Song, L., Ge, Y., Liu, W., Wang, X., and Shan, Y. Yolo-world: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16901–16911, 2024.

Gupta, A., Dollar, P., and Girshick, R. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5356–5364, 2019.

Li, L. H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.-N., et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10965–10975, 2022.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024.

Lv, Y., Fang, Y., Chi, W., Chen, G., and Sun, L. Object detection for sweeping robots in home scenes (odsr-ihs): A novel benchmark dataset. *IEEE Access*, 9:17820–17828, 2021.

Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., et al. Simple open-vocabulary object detection. In *European conference on computer vision*, pp. 728–755. Springer, 2022.

Sapkota, R. and Karkee, M. Yoloe-26: Integrating yolo26 with yoloe for real-time open-vocabulary instance segmentation. *arXiv preprint arXiv:2602.00168*, 2026.

Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., Li, J., and Sun, J. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8430–8439, 2019.

Wang, A., Liu, L., Chen, H., Lin, Z., Han, J., and Ding, G. Yoloe: Real-time seeing anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 24591–24602, 2025.

Wang, J., Zhang, H., Hong, H., Jin, X., He, Y., Xue, H., and Zhao, Z. Open-vocabulary object detection with an open corpus. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6759–6769, 2023.

Yenamandra, S., Ramachandran, A., Yadav, K., Wang, A., Khanna, M., Gervet, T., Yang, T.-Y., Jain, V., Clegg, A. W., Turner, J., et al. Homerobot: Open-vocabulary mobile manipulation. *arXiv preprint arXiv:2306.11565*, 2023.

A. Dataset Details

A.1. ODSR-IHS Dataset Overview

ODSR-IHS is a home-scene benchmark dataset constructed for object detection in indoor mobile robots such as sweeping robots. Unlike conventional object detection datasets that are typically collected from human-level viewpoints or web images, ODSR-IHS is captured from a near-ground robot-view camera. Therefore, it reflects practical challenges in home-robot perception, including low observation viewpoints, indoor clutter, partial occlusion, small obstacles, and irregular object layouts. The data collection was designed to cover diverse residential environments, apartment layouts, and interior decoration styles, with more than 50 indoor scenes and variations in daytime/nighttime and weather conditions. This setting is motivated by the fact that object detection models trained on generic images do not directly transfer well to indoor environments observed from a robot viewpoint. Table 4 summarizes the dataset statistics and camera specifications of ODSR-IHS.

The original dataset consists of 6,000 images and 16,409 object instances across 14 object categories: bed, sofa, cabinet, stool, table, closestool, trashcan, slippers, wire, socks, carpet, book, feces, and curtain. These categories were selected based on indoor objects that affect obstacle avoidance and navigation decisions of sweeping robots. In particular, small or deformable objects such as wires, socks, and slippers are important targets because they can interfere with robot movement. As shown in Table 4(a), all objects are annotated with bounding boxes following the PASCAL VOC format, and the images are normalized to a resolution of 640 $\times$ 480. Table 4(b) reports the camera configuration: the dataset was collected using a Sony STARVIS IMX327 sensor with a 120° wide-angle lens, where the camera center is positioned 45 mm above the ground and the horizontal angle is 70°.

Table 4. Overview of the ODSR-IHS benchmark, including dataset statistics and camera specifications.

(a) Dataset Statistics

Item	**Value**
Images	6,000
Object instances	16,409
Classes	14
Annotation type	Bounding box
Annotation format	PASCAL VOC
Image resolution	640 $\times$ 480

(b) Camera Specifications

Property	**Value**
Camera sensor	Sony STARVIS IMX327
Lens	120° wide-angle
Focal length	3.6 mm
Interface	12 mm
Mirror size	1/2.5
Camera height	45 mm
Horizontal angle	70°
Resolution	640 $\times$ 480

A.2. Data Split and Auxiliary Query Analysis

To separate final evaluation from auxiliary analysis, we exclude 7 images without annotations from the 6,000 ODSR-IHS images and use the remaining 5,993 images. Among them, 1,000 images are used as a development split, and the remaining 4,993 images are used as a held-out evaluation split. The development split is used only for analyzing query formulations and designing refinement prompts, while final performance is reported only on the held-out evaluation split. Samples from the development split are not included in the final test-set evaluation.

On the development split, we conduct an auxiliary query-axis analysis to better understand the potential upper bound of test-time query refinement. For each image-level target annotation, we ask a VLM to describe the target object along several class-preserving axes, including color, geometry, material, distinctive parts, and everyday lexical variants. We then attach each description to the target class name to construct multiple detector-facing query variants.

Each query variant is evaluated by re-running the frozen YOLOE detector without updating detector weights. By comparing pseudo-target metrics across variants, we analyze which types of query formulations tend to improve detector responses under different failure modes. This analysis is not used as a test-set evaluation protocol or as supervision for final evaluation. Instead, it serves as an upper-bound probe and provides practical guidance for designing the state-specific refinement prompts used by our agents.

B. State Definition and Routing

B.1. Plan Agent as a Diagnostic Router

The Plan Agent serves as a diagnostic router rather than a query generator. Given the target class, current query, target evidence, YOLOE prediction summary, evidence-based target scores, and compact refinement history, it predicts the current detector state and describes the specific failure manifest in the current iteration. Its role is to identify not only whether the current query fails or succeeds but also why it fails, producing a descriptive handoff that guides the subsequent state-specific specialist.

The Plan Agent is explicitly instructed not to generate the refined query itself. Instead, it analyzes the relationship between the target evidence and the detector response. For example, it checks whether the detector completely misses the target region, produces a weak but target-like response, or responds strongly to distractor regions. It also uses the compact history to avoid repeating previously failed query attempts. This design separates state diagnosis from query generation: the Plan Agent provides the reasoning basis for refinement, while the specialist agents generate candidate queries conditioned on the diagnosed state.

B.2. State Definitions

We define one success state and three failure states. A sample is assigned to GOOD when the detector response is sufficiently aligned with the target evidence. In this case, the refinement loop terminates.

MISS indicates that the detector does not produce a reliable response near the target region. This state is typically associated with cases where the class-name query is too generic, visually mismatched, or insufficient to activate the detector on the target object.

WEAK indicates that the detector produces a target-like response, but the response is low-confidence or poorly localized. Unlike MISS, the detector already shows partial evidence of the target, so the refinement should strengthen the existing response rather than broadly rewrite the query.

NOISY indicates that the detector responds to the target but also produces many high-confidence false positives. In this case, the refinement should introduce target-positive distinguishing cues that separate the target from nearby distractors, while avoiding negative phrases such as “not” or “without”.

B.3. Planner Handoff to Specialist Agents

After predicting the state, the Plan Agent produces a compact handoff for the corresponding specialist agent. The handoff contains three types of information: (1) the diagnosed state and the reason for the diagnosis, (2) target-grounded visual cues extracted from the target evidence, and (3) history-aware constraints indicating which query attempts should not be repeated.

For MISS cases, the handoff emphasizes the visual appearance of the pseudo target and asks the Explorer Agent to recover a missing detector response using class-preserving query variants. For WEAK cases, the handoff summarizes the weak detection evidence and asks the Strengthener Agent to add minimal visible cues that reinforce the same target. For NOISY cases, the handoff highlights the difference between the pseudo target and false-positive regions, and asks the Inspector Agent to propose target-positive cues that suppress distractor responses.

This handoff mechanism allows each specialist to generate queries using state-specific reasoning while remaining grounded in the same evidence. It also prevents the agents from repeatedly proposing query variants that have already failed in previous iterations.

C. Auxiliary Query-Axis Analysis

To understand how different query formulations affect a frozen open-vocabulary detector, we conduct an auxiliary analysis on a separate 1,000-image development split. For each annotated target, we use a VLM to generate class-preserving descriptions along several query axes, including color, geometry, material, distinctive parts, and everyday lexical variants. Each description is combined with the target class name to form a candidate query, and YOLOE is evaluated with each candidate. This analysis is used only to study how query axes influence detector responses under different failure modes. It is not used as a held-out test evaluation protocol, and no real annotations from the final evaluation split are used during

refinement. In the main experiments, query candidates are selected using target evidence and detector feedback on the held-out split. The dev-set analysis only informs the design of prompt constraints and the set of query axes available to the agents.