

COOL: Curiosity-Driven Object Ownership Learning for Personalized Robotic Assistance

Samira Huber* Ruben Hammele* Sören Pirk

Department of Computer Science, Kiel University, Germany

{samira.huber, ruben.hammele}@cs.uni-kiel.de, soeren.pirk@gmail.com



Figure 1: COOL enables personalized object search by combining long-term memory, ownership inference, and spatial grounding. Given a natural-language request such as “Find my backpack” (left), the robot first retrieves prior observations linking people and object instances, then infers the likely owner from accumulated human–object interaction evidence (middle), and finally grounds the selected object in the map to navigate to its most likely location (right).

Abstract: Robots are increasingly expected to provide personalized services in everyday environments. To do so, they must ground natural-language commands such as *Where is my backpack?* or *Find my bottle* and execute them by reasoning about object instances, people, locations, and ownership. This is challenging because ownership is rarely labeled explicitly and must be inferred from long-term, behavioral evidence of human–object interactions. To address this, we present *COOL*, a novel robotic framework for autonomously learning object ownership from everyday observations and maintaining a long-term spatial memory of its environment. To keep its memory current, *COOL* uses an agent-based curiosity-driven data collection strategy that guides the robot toward the most promising locations to gain information and refresh stale observations. Offline experiments, ablation studies, and real-world evaluations show that *COOL* can infer ownership relations from real-world interactions and use this knowledge for ownership-conditioned navigation and task execution. Code, prompts, and evaluation data are available at: <https://samirahuber.github.io/cool>.

Keywords: Spatial Memory, Object Ownership, Robotic Assistance

1 Introduction

In everyday human environments, robots are increasingly tasked with providing personalized services through natural-language commands. In such settings, users may issue instructions such as *Where is Simba’s backpack?* or *Find my bottle*. Executing such commands requires more than object detection and navigation: the robot must identify the intended object instance, determine who it is associated with, and know where it is likely to be located. This is particularly challenging in shared environments such as offices or homes, where ownership is often ambiguous because objects may be visually similar, shared, borrowed, or moved over time. Ownership is rarely directly observable: people do not continuously label their belongings, and ownership can become ambiguous

*Equal contribution

when objects are shared, borrowed, or moved. Object ownership must therefore be inferred indirectly from repeated observations, spatial context, and human–object interactions. A service robot must collect and maintain long-term evidence about people, objects, places, and interactions.

Existing work addresses parts of this problem, but not the full ownership-learning pipeline. Memory-based navigation systems store spatial, semantic, or temporal observations for object search and embodied question answering, but do not model persistent ownership relations [1, 2, 3, 4, 5, 6, 7]. Conversely, personalized object-search systems often require explicit user input, demonstrations, or visual references [8, 9, 10, 11, 12, 13, 14, 15]. What remains missing is a robotic system that autonomously learns ownership from long-term observations and uses it for task execution.

In this paper, we address this gap by treating ownership as a latent relation inferred from long-term robot observations rather than as an explicit label. Our framework associates repeated detections of people and objects, records human–object interactions, and converts accumulated evidence into ownership-conditioned actions for personalized object search. Specifically, we present *COOL* (*Curiosity-Driven Object Ownership Learning for Personalized Robotic Assistance*), an integrated framework for ownership-aware object search in shared human environments. *COOL* maintains a long-term spatial memory that links people, object instances, locations, and interaction histories over time. It continuously clusters repeated observations using re-identification, detects human–object interactions as ownership evidence, and records where people and objects have been observed. A curiosity-driven agent guides the robot toward under-observed areas to refresh stale or incomplete memory. At query time, a user-facing agent resolves commands such as *Find Simba’s backpack* by retrieving relevant observations, inferring likely person–object relations, and selecting the corresponding object instance (see Fig. 1).

We evaluate *COOL* through offline experiments, ablations, and real-world tests in an office environment. These experiments assess the framework’s ability to associate repeated observations of the same people and objects and infer ownership relations from human–object interaction evidence. It updates object locations over time and uses this knowledge for object search and navigation. Combining long-term spatial memory, re-identification, interaction detection, and active data collection enables robots to autonomously acquire ownership knowledge and apply it to personalized tasks.

In summary, the main contributions of this paper are: (1) **Ownership inference**: a dynamic framework for inferring person–object ownership from long-term robot observations, without explicit labels, demonstrations, or dialogue. (2) **Persistent spatial memory**: a structured memory linking people, object instances, locations, and human–object interaction histories. (3) **Ownership-conditioned task execution**: a tool-using agent that answers ownership-conditioned questions and executes navigation commands from memory. (4) **Memory refresh**: an autonomous data-collection agent that seeks informative locations and refreshes stale observations.

2 Related Work

Memory-based Navigation. Recent work gives embodied agents long-term memory for object search, exploration, and embodied question answering. These methods store prior observations as maps, graphs, vector memories, or scene graphs, and often combine them with large language models (LLMs) or vision-language models (VLMs) for retrieval and reasoning [1, 2, 3, 4, 5, 6]. Retrieval-augmented navigation and multimodal retrieval-augmented generation (RAG) reuse prior episodes or perceptual streams for navigation and temporal reasoning, while general LLM memory systems focus on textual recall and memory consolidation [7, 16, 17, 18, 19, 20]. Unlike prior work, *COOL* links persistent people, objects, locations, and interaction histories to infer person–object ownership relations for personalized robot assistance.

Re-Identification. Visual re-identification links observations of the same entity across time using image embeddings. Many general-purpose backbones provide strong visual features [21, 22, 23, 24], while Person ReID models such as OSNet are optimized for identity matching [25, 26, 27]. Tracking-by-detection methods link detections across nearby frames using motion, appearance, and confidence cues [28, 29, 30]. Recent work improves robustness under deployment shift, hetero-

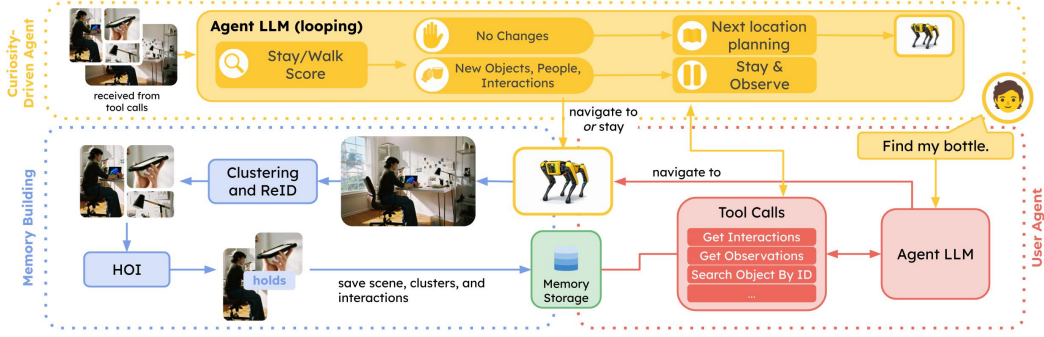


Figure 2: Our framework integrates curiosity-driven data collection with persistent spatial memory and ownership-conditioned reasoning. A curiosity-driven agent (top) decides when to stay and observe or move to more informative locations, the memory module (blue) clusters repeated person and object detections and stores human–object interactions, while a user agent (red) retrieves this evidence to infer ownership and execute personalized commands such as *Find my bottle*.

genous sensors, and multimodal object appearance [31, 32, 33, 34, 35, 36, 37, 38, 39]. COOL differs by using tracking and re-identification as components of long-term robot memory, linking repeated observations of people and object instances to spatial observations and interaction histories for ownership-conditioned navigation.

Object Ownership. Recent work has studied social relations and personalization cues for robotic assistance. Early systems model owner–object relations with probabilistic inference, visual cues, or demonstrations [8, 9]. Recent personalized search and navigation methods use dialogue, user feedback, open-vocabulary grounding, or image/language-specified object instances [10, 11, 12, 13]. Other methods localize personal objects from limited examples or on-device visual matching [14, 15]. COOL infers owner–object associations from long-term robot observations and uses them to decide where to search for or navigate to objects associated with a specific person.

Human–Object Interaction (HOI). HOI recognition identifies how people interact with objects. Recent work extends HOI beyond closed-set triplets toward open-vocabulary and open-world recognition with vision-language or multimodal language models [40, 41, 42, 43]. Related embodied work uses interaction reasoning for long-horizon hand-object manipulation from language [44]. In contrast, COOL stores detected interactions with person, object, place, and time information, and uses these histories as evidence for ownership-conditioned navigation.

3 Method

The key contribution of COOL is ownership inference: enabling mobile robots to reason about which objects belong to which people from situated, long-term observations (Fig. 2). During operation, the robot grounds visual observations in a metric map and stores them in a spatial memory (Sec. 3.1). The system’s reasoning is driven by two specialized language agents (Fig. 3). First, a *user agent* (Sec. 3.2) queries this memory to resolve entities, infer ownership relations, answer ownership-related questions, and issue navigation commands. Second, a *curiosity agent* (Sec. 3.3) actively expands and refreshes the memory through curiosity-driven data collection.

3.1 Persistent Person and Object Memory

A persistent memory links repeated detections of the same people and object instances over time. This association is necessary because ownership evidence is accumulated across multiple observations. At each observation step, the robot detects visible people and objects in its RGB-D camera stream using YOLO11m-seg [45], selected based on the results of an ablation study reported in App. F.1. We then assign these observations to persistent clusters with a pipeline that consists of three stages: First, the system performs *short-term association* by preserving identity through BoT-Sort [46] track continuity. If the track ID is lost, a short-term spatial cache reassigns nearby same-



Figure 3: Agent-based reasoning in COOL: we use two language agents to operate on a shared long-term spatial memory. A user agent (red) retrieves people, objects, locations, and interaction histories to infer ownership and generate navigation goals, while a curiosity-driven exploration agent (blue) decides where to go next to collect ownership-relevant evidence and refresh stale observations.

class detections to the previous cluster. This stage handles frame-to-frame continuity and brief detector dropouts. Second, we perform *nearest-neighbor retrieval* based on embeddings of people and objects. Specifically, we represent people with OSNet $\times 1.0$ body-appearance embeddings and objects with ConvNeXt-Tiny [21] visual embeddings. These feature extractors were selected based on ablation studies comparing several embedding models and input preprocessing variants. For person re-identification, OSNet provided the strongest performance among the off-the-shelf models. We fine-tuned it on our data to further improve its clustering performance. For object re-identification, ConvNeXt-Tiny achieved the highest F1 score among the evaluated feature extractors (App. F.2, App. F.3). For the current observation embedding z_t , we retrieve the k nearest stored observations across all clusters using cosine similarity. We then group these neighbors by their assigned cluster k and compute a score

$$s_k = w \cdot \bar{s}_k + (1 - w) \cdot s_k^{\max}, \quad (1)$$

where $\bar{s}_k$ is the average cosine similarity to cluster k 's neighbors, $s_k^{\max}$ is the best similarity among them, and w controls the trade-off between the average and the peak similarity. The best cluster is accepted only if it satisfies the embedding similarity threshold τ_{emb} , receives at least $h_{\min}$ neighbor hits, and has a margin of at least δ over the runner-up. Appendix A lists the clustering parameters. Third, for predefined static classes, such as chairs and tables, the system performs *position-based matching*. If an object of the same class is observed again at approximately the same location, the observation is assigned to the existing cluster. This stage helps separate visually similar static objects by location.

Online association can fragment one physical entity across clusters due to viewpoint changes, occlusion, or embedding noise. The system therefore periodically compares same-class cluster centroids and merges clusters when their accumulated appearance evidence is sufficiently similar. Person identity is estimated from both face and body appearance. The association pipeline is extended with InsightFace embeddings [47] for face-visible detections. Face embeddings provide high-specificity matches, while body embeddings preserve continuity when the face is not captured.

3.2 Agent-Based Ownership Inference and Navigation

We formulate ownership as a latent, dynamically inferred relation between a person cluster p and an object-instance cluster o . Unlike explicit labels, this relation is not stored as a fixed database attribute. Instead, COOL infers it at query time from the robot's accumulated memory of observations and human-object interactions. This design allows ownership estimates to update when new evidence is collected, ambiguous histories become clearer, or ownership changes.

Ownership Inference. Given a query q , such as *Who owns the mug?* or *Go to Simba's mug*, the system first resolves the referenced entities using structured memory tools (see Appendix B). Let $\mathcal{P}$ denote the set of persistent person clusters and $\mathcal{O}$ the set of persistent object-instance clusters. For each candidate person-object pair $(p, o) \in \mathcal{P} \times \mathcal{O}$, COOL retrieves an ownership evidence set

$$\mathcal{E}_{p,o} = \{e_i = (p_i, o_i, a_i, \ell_i, t_i, c_i)\}, \quad (2)$$

where each event records the involved person and object clusters, the detected interaction type a_i , the map location ℓ_i , timestamp t_i , and an optional caption c_i . Interactions are inferred from detected person–object bounding-box pairs using a VLM, inspired by recent HOI pipelines [40]. These events capture direct human–object interactions, repeated person and object co-occurrences, and their observed locations. Direct physical interactions, such as drinking, using, or picking up, provide stronger ownership evidence than holding or weak co-presence.

A tool-using language agent reasons over the retrieved evidence through a constrained multi-turn retrieval loop. Using structured tool calls, the agent retrieves candidate entities and evidence records and must define the ownership state: *owned by person p* , *shared*, or *unknown*. It is instructed to weigh evidence according to interaction specificity, repetition, recency, and consistency across locations. Repeated direct interactions by the same person increase confidence in ownership, while a single weak interaction or mere spatial co-presence is insufficient. Evidence distributed across people is labeled as *shared*, while weak or conflicting evidence is labeled as *unknown*.

Ownership-conditioned Navigation. The inferred relation is used to select an object instance and map it to a reachable navigation goal. Detections are grounded in a pre-recorded metric map using `slam_toolbox` [48]. Each detection is assigned a 3-D position by back-projecting the median depth inside the YOLO bounding box. Let ${}^w\mathbf{p}_r(t)$ and ${}^w\mathbf{R}_r(t)$ denote the robot position and orientation in the map at time t , and let ${}^r\mathbf{p}_{obj}(t)$ denote the object position in the robot frame. The corresponding global object position is

$${}^w\mathbf{p}_{obj}(t) = {}^w\mathbf{p}_r(t) + {}^w\mathbf{R}_r(t){}^r\mathbf{p}_{obj}(t). \quad (3)$$

For possessive commands such as *Go to Simba’s mug*, the agent first infers the relevant owner–object relation, selects the object instance most strongly associated with the named person, and retrieves its most recent reliable map position. For generic commands without an owner reference, such as *Go to the blue mug*, the agent selects the most recently observed matching instance unless additional disambiguating evidence is available.

3.3 Curiosity-Driven Data Collection

Since ownership is inferred from long-term evidence rather than explicit labels, the robot must decide where to spend its sensing time to observe useful events: people interacting with objects, objects appearing in new locations, and previously stored observations becoming stale. We therefore formulate curiosity-driven data collection as an active memory-refresh problem. The robot must balance three competing goals: revisiting locations where ownership evidence is likely to occur, checking locations where new changes are expected, and refreshing under-observed areas.

To this end, we use a language-agent-based exploration loop rather than random wandering or a fixed patrol schedule. At defined intervals, the agent observes the current room and determines whether the scene has changed compared to its recent history. Changes include variations in object counts, visible people, object locations, and human–object interactions. If the agent expects near-term ownership-relevant activity, or has recently arrived and needs more observations, it issues a *stay* command. Staying allows the perception pipeline to accumulate repeated detections, improve person and object association, and capture interaction events that may provide ownership evidence.

Conversely, if the scene remains static, the agent issues a *move* command. To select the next room, the agent queries spatial memory with three tools and scores candidate rooms using three signals: (1) **Historical Hotspots** identify rooms with a high empirical rate of scene changes or human–object interactions. They are likely to contain direct ownership evidence. (2) **Expected Change** returns the time elapsed since the last detected change in each room. Rooms that have not changed recently but historically change often are likely to provide new evidence if revisited. (3) **Staleness** identifies rooms whose observations are old or sparse. Visiting stale rooms prevents the robot from relying on outdated object locations and helps update ownership-conditioned navigation targets.

This makes curiosity explicitly tied to the ownership inference objective: the robot preferentially collects evidence that can strengthen, revise, or disambiguate person–object relations, while also maintaining fresh object-location estimates for navigation. When the robot navigates to a selected

Table 1: Results on the offline reasoning benchmark. We report overall and category-specific accuracy over three seeds, together with the average number of tool calls and wall-clock time per query.

Variant	Accuracy (%)	Interaction (%)	Location (%)	Lookup (%)	Ownership (%)	Navigation (%)	Func. calls	Time (sec)
Plain LLM	37.0 $\pm$ 0.0	61.0 $\pm$ 0.0	26.0 $\pm$ 0.0	33.0 $\pm$ 0.0	0.0 $\pm$ 0.0	0.0 $\pm$ 0.0	0.00 $\pm$ 0.00	4.69 $\pm$ 0.07
Heuristic	21.0 $\pm$ 0.0	0.0 $\pm$ 0.0	78.0 $\pm$ 0.0	0.0 $\pm$ 0.0	0.0 $\pm$ 0.0	11.0 $\pm$ 0.0	0.25 $\pm$ 0.00	0.01 $\pm$ 0.00
No Interaction	36.0 $\pm$ 2.0	6.0 $\pm$ 3.0	96.0 $\pm$ 0.0	33.0 $\pm$ 0.0	33.0 $\pm$ 12.0	15.0 $\pm$ 6.0	4.60 $\pm$ 0.15	6.85 $\pm$ 0.68
Rule-based	15.0 $\pm$ 0.0	18.0 $\pm$ 0.0	37.0 $\pm$ 0.0	0.0 $\pm$ 0.0	0.0 $\pm$ 0.0	11.0 $\pm$ 0.0	0.25 $\pm$ 0.00	0.01 $\pm$ 0.00
COOL	89.0 $\pm$ 2.0	94.0 $\pm$ 0.0	96.0 $\pm$ 0.0	33.0 $\pm$ 0.0	100.0 $\pm$ 0.0	89.0 $\pm$ 0.13	2.18 $\pm$ 0.05	2.35 $\pm$ 0.10

destination, the perception and memory pipeline remains active. Newly detected people, objects, object locations, and human–object interactions are grounded in the metric map, associated with existing clusters when possible, and inserted into the long-term spatial memory. Through this closed-loop process, COOL continuously enriches and refreshes the evidence used for ownership inference.

4 Experiments and Results

We implement the hardware-agnostic framework in ROS 2 and evaluate it on a Boston Dynamics Spot robot [49] equipped with RGB-D perception and LiDAR-based SLAM. Navigation is executed with Nav2 [50, 51] and uses the SmacPlanner2D global planner [52] over the 2-D occupancy grid. We use Qwen3-VL-4B [53] for both agents as the backbone, which is commonly used in robotics [54, 55, 56, 57]. The model runs on an RTX 4090. For the longest queries (2k input tokens), we observe an end-to-end latency of ~ 2 s. We store embeddings and text rather than raw images, minimizing storage requirements to 11 KB per observation. Since embeddings can enable re-identification, we obtained consent from all subjects.

4.1 Offline Reasoning Benchmark: Synthetic Dataset

Synthetic Dataset. We evaluate the memory-reasoning capabilities of COOL on a synthetic office benchmark to systematically test ownership reasoning components. The benchmark represents one day of robot perception in a shared office (see Appendix C). It contains rooms, people, objects, timestamped observations, and human–object interactions. We generated an initial trace using an LLM and manually reviewed it for plausibility. The log was evaluated by examining the recorded activities and their temporal sequence to determine whether they were consistent with a realistic office workday. Implausible activities, sequences, or transitions were identified and corrected based on human judgment. We use this synthetic dataset because, to our knowledge, no existing dataset combines navigation coordinates with ground-truth ownership data and recurring people and objects [58, 59]. From this state, we derive 79 natural-language questions. These questions probe the system’s ability to reason over time and space, aggregate interaction histories, infer owner–object associations, and resolve navigation targets, helping identify the strengths and failure modes of each subcomponent. All questions were reserved for final evaluation.

We compare COOL with four variants. Semantic similarity is used to evaluate the answers. (1) **Plain LLM** receives a flattened database snapshot. (2) **Rule-based** maps questions to pre-defined SQL queries using keywords and regular expressions. (3) **No interaction** runs the agent without interaction-related tools. (4) **Heuristic** uses recency and proximity rules.

COOL achieves the highest overall accuracy (Table 1). This shows the benefit of structured tool-based memory access over flattened database prompting.

Limited ownership reasoning in baselines. *Rule-based*, *Plain LLM*, and *Heuristic* score 0% on ownership, while *No Interaction* reaches 33% with high variance. COOL achieves 100% in this oracle-memory setting, showing that reliable ownership inference requires structured access to interaction and observation histories.

Importance of interaction information. The *No Interaction* ablation performs well on location questions, but fails on interaction and ownership questions. This confirms that human–object interactions are necessary evidence for object-use and ownership reasoning.

Table 3: Real-world navigation tasks and evaluated capabilities. Accuracy is measured over 10 runs per task. A “+” denotes a capability that is required for solving the task, whereas “−” denotes a capability that is not involved in the task. Details are provided in Appendix D.

Task Type	Ownership	Person Distinction	Person ReID	Object ReID	Long-Term Tracking	Visual Ambiguity	Reachability Reasoning	Accuracy (%)
Room Navigation	−	−	−	−	−	−	+	100
Object Relocation	+	−	+	+	+	−	−	90
Owner-based Distinction	+	+	+	+	+	−	−	100
Visual Ambiguity	+	+	−	−	−	+	−	90
Ownership-Preserving Update	+	+	−	+	+	+	−	90

4.2 Video-Based Evaluation: Sitcom Dataset

Sitcom Dataset. To validate ownership reasoning, we evaluate COOL on three hours of multi-person sitcom footage. It captures realistic scenarios with object passing, identity changes due to clothing or viewpoint shifts, and ambiguous ownership. Existing datasets lack the combination of recurring identities, realistic interactions, and ground-truth ownership labels [58, 59].

We manually inspect the videos to identify objects and formulate 30 ownership questions. The system processes the video end-to-end with no oracle inputs, including missed detections and identity errors from the perception pipeline. A question is correct if the VLM identifies the owner or correctly labels an object as shared.

Robustness to Perception Errors. We evaluate robustness to perception errors by perturbing the sitcom dataset observations with failure modes. We simulate eight failure modes (Table 2): detection dropout (40%, 60%, 80%), false-positive injection (10%, 50%), face occlusion (50%), track fragmentation (20%), and identity swaps (50%). Each condition is evaluated over 10 seeds.

The method maintains strong performance under moderate perception noise, with accuracy of 81.48% under 40% dropout, 78.51% with 10% false positives, and 79.63% with 50% face occlusion. Performance degrades more substantially under severe dropout and identity swaps, reaching 61.12% and 50.01% respectively. These results suggest that the approach is generally tolerant to noisy or incomplete observations, while consistent identity information is particularly important for reliable ownership matching. Details can be found in Appendix C.1.

Table 2: Robustness test on sitcom footage. Accuracy (%) as mean $\pm$ standard deviation (10 seeds).

Robustness Test	Accuracy (%)
Baseline	84.33 $\pm$ 2.75
Detection Dropout (40%)	81.48 $\pm$ 5.30
Detection Dropout (60%)	61.12 $\pm$ 5.78
Detection Dropout (80%)	47.78 $\pm$ 5.77
False-Positive Injection (10%)	78.51 $\pm$ 5.78
False-Positive Injection (50%)	66.67 $\pm$ 5.99
Face Occlusion (50%)	79.63 $\pm$ 2.00
Track Fragmentation (20%)	63.68 $\pm$ 6.95
Identity Swap (50%)	50.01 $\pm$ 12.47

4.3 Real-World Experiments

To test practical applicability, we evaluate whether COOL can perform real-world, ownership-conditioned navigation. We conduct the experiments in an office environment. In each experiment, the robot receives a natural-language command, such as *Go to Simba’s ball*. To succeed, the robot must identify the correct object instance from its memory and navigate to that object’s most recently observed location. Each trial starts from a random position in the office. We define a trial as successful if the robot reaches the navigation goal, stops, and finishes within a 1 m radius of the target. We evaluate five distinct navigation tasks with 10 runs each. Table 3 summarizes the tested capabilities.

Results. Table 3 and Figure 4 show that COOL successfully completes most real-world navigation trials. The system achieves a success rate of at least 90% in every scenario, and reaches 100% for *Room Navigation* and *Owner-based Distinction*. This indicates that inferred ownership relations can be used for navigation commands in a real office environment.

Disambiguation and location updating. *Visual Ambiguity* tasks show that COOL can disambiguate visually similar objects using interaction-derived ownership evidence. The *Ownership-Preserving*

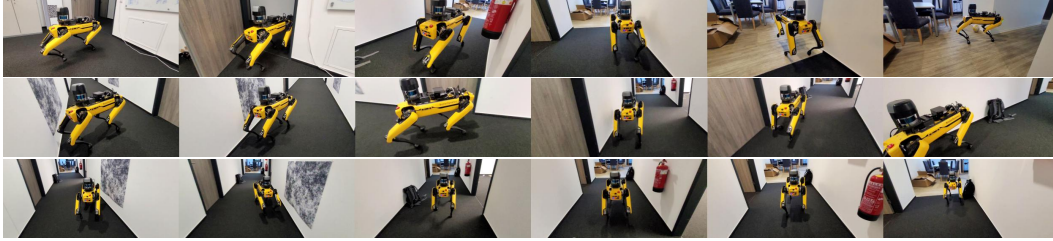


Figure 4: Three object ownership inference examples. 1) Object Relocation: Find Simba’s ball (top), 2) Owner-based Distinction: Find Nemo’s backpack (middle), 3) Visual Ambiguity: Find Simba’s backpack (bottom).

Update task further shows that the system preserves the ownership association while updating the target location when the object is later observed elsewhere.

4.4 Curiosity-Driven Navigation and Exploration

We evaluated the curiosity agent on 15 synthetic office logs spanning diverse environmental conditions and change distributions. Synthetic data is necessary because existing datasets do not provide long-term, multi-person interactions with recurring ownership evidence and controlled navigation [60, 61, 62]. We set the LLM temperature to 0 to reduce sampling variability. Each log contains 183 samples for each room, visible people, visible objects, and a scene description. The task is to detect changes. A change is any difference between two consecutive observations of a room: a person, object, or room fixture appearing or disappearing, or a change in what people are doing. Per-dataset details and results are presented in Appendix E.

Curiosity evaluation. In this experiment, the agent detects scene changes and decides whether to remain in its current room or relocate. Recall is the fraction of changes detected among those occurring while the agent is present. Precision is the fraction of reported changes that are correct. Detection yield is the number of observed changes per relocation. Change-detection F1 measures whether a room changed since the agent’s previous observation, evaluated only at observed steps. Baseline strategies select rooms using observation statistics such as visit recency, change frequency, or estimated change probability. In contrast, COOL uses a VLM to determine whether continued observation is warranted.

Table 4: Navigation strategy comparison across all logs. *Moves* denotes room-to-room relocations, *Observed* denotes change events that occurred while the agent was in the room, and *Yield* is observed changes per move. *Recall* and *Precision* are measured over the agent’s per-visit change judgements. *Best* counts logs on which a strategy attains the highest F1 among those listed. All values are means over 15 logs $\times$ 10 seeds.

Strategy	F1	Recall	Prec.	Yield	Moves	Observed	Best
Random	65.0%	88.2%	52.0%	0.36	182	65	0/15
Round-Robin	67.9%	82.9%	57.8%	0.65	91	59	0/15
Frequency	69.9%	86.2%	59.5%	0.50	146	73	0/15
Staleness	63.1%	90.6%	48.7%	0.36	182	66	0/15
Active Mapping	63.2%	90.5%	48.8%	0.36	182	65	0/15
Greedy Hazard	76.5%	83.3%	71.3%	0.64	108	69	0/15
Entropy	80.1%	70.7%	93.3%	5.15	14	72	1/15
Beta-Entropy	72.0%	83.4%	63.8%	0.38	182	69	0/15
COOL	85.0%	85.0%	86.0%	2.11	30	63	14/15

Table 4 shows that COOL achieves the highest F1 score of 85.0% and is the only strategy with both precision and recall above 80%. Sweeping strategies reach high recall but poor precision (90.6%/48.7%), since they visit constantly and most visits have nothing to report, whereas Entropy shows the opposite pattern (93.3%/70.7%). COOL efficiently prioritizes active rooms, requiring only 30 moves compared to 182 for sweeping strategies and 91 for Round-Robin, while observing 2.11 changes per move versus 0.36–0.65 for coverage-driven policies. The advantage further increases with world size, reaching 85.4% F1 on 24-room logs versus 78.1% for the strongest baseline on those logs.

Navigator ablation. COOL follows the top-ranked destination in over 99% of relocations, but its navigator additionally determines when to relocate. Replacing this relocation policy with unconditional score argmax increases moves from 30 to 105 and reduces F1 from 85.0% to 76.4%, while observing fewer interaction steps per move (1.27 vs. 3.52). The argmax variant does leave fewer changes unobserved in absolute terms (107 vs. 400 per log) at 3.5 $\times$ the movement cost, so in this

configuration the navigator’s contribution is deciding when to relocate under a movement budget rather than where to go.

Trapping. Entropy attains the highest precision of any strategy (93.3%) with only 14 moves and the highest yield (5.15), but the lowest recall (70.7% vs 85.0% for COOL, Table 4). It is accurate about the single room it occupies and blind to the rest of the building, leaving twice as many distinct change episodes unobserved as COOL (818 vs 400 per log). Trapping is an efficient way to exploit known activity, whereas COOL distributes its observations across the environment.

Real-world run. We deployed the curiosity-driven agent on the robot in a four-room office environment with a kitchen and three offices. Figure 5 shows the resulting trajectories, where the color gradient indicates time. Across 4.5 hours of exploration, the robot autonomously navigated the environment, selecting when to move between rooms and when to remain in place, with charging breaks between runs. The robot spent more time in rooms with observed human activity, suggesting that the curiosity policy directed data collection toward interaction-rich areas.

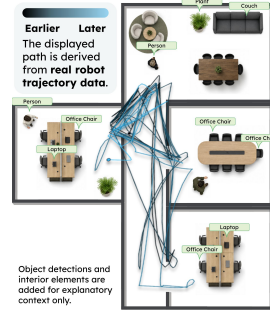


Figure 5: Real-world curiosity-driven exploration on the Spot robot. Trajectories over a 4.5-hour deployment in a four-room office environment are shown.

5 Limitations

COOL infers person–object ownership relations from observed behavior rather than explicit ownership labels. This enables autonomous ownership learning, but ownership remains uncertain when evidence is sparse or ambiguous, for example when objects are borrowed, moved, or only briefly handled. In such cases, the system can report ownership as unknown or shared, but additional cues such as clarification questions, visible name labels, or explicit user confirmations may be needed. The system stores long-term observations of people, objects, locations, and interactions, which raises privacy and social acceptability concerns. Deployments in shared spaces should therefore include clear consent procedures, data minimization, access control, and retention policies. Finally, our real-world evaluation is limited to a single office environment with a small number of people, objects, and ownership-conditioned navigation tasks. Broader studies across more environments, longer deployments, and more diverse shared-object scenarios are needed to assess generalization.

6 Conclusion

We presented COOL, a robotic framework for learning object ownership in shared human environments from long-term observations. Rather than relying on explicit labels, demonstrations, or dialogue, COOL infers ownership from accumulated evidence about people, object instances, and human–object interactions. This allows the system to adapt to ownership changes and incorporate new evidence. It combines persistent spatial memory with two agents for data collection and task execution. The curiosity-driven agent autonomously explores the environment to refresh stale or incomplete memory and gather ownership-relevant evidence. The user agent interprets ownership-conditioned requests by grounding them in spatial memory, inferring the relevant owner–object relation, and directing the robot toward the inferred target location. Our evaluations show that interaction memory, ownership-based navigation, and active observation jointly support ownership-aware assistance. On a physical robot, COOL improved ownership reasoning, resolved personalized search requests, collected relevant evidence more efficiently, and handled visually similar objects and changing object locations in an office environment. This suggests that persistent, interaction-aware memory and active observation can support personalized robotic assistance. Future work will explore interactive clarification, broader multi-user deployments, and mobile manipulation, extending the system beyond object localization toward retrieval and handover.

Acknowledgement

The authors thank the anonymous reviewers for their valuable feedback. The research was conducted under the RobOdin project funded by the European Union through its Interreg Germany-Denmark funding program. We acknowledge technical support by the Embodied AI Center at Kiel University.

References

- [1] H. Niu, L. Zhang, X. Ji, R. Ying, P. Liu, and F. Wen. SSMG-Nav: Enhancing Lifelong Object Navigation with Semantic Skeleton Memory Graph. *arXiv preprint arXiv:2603.01813*, 2026.
- [2] A. Anwar, J. Welsh, J. Biswas, S. Pouya, and Y. Chang. ReMEMBR: Building and Reasoning Over Long-Horizon Spatio-Temporal Memory for Robot Navigation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2838–2845. IEEE, 2025.
- [3] S. Huber, K. Pelzer, D. M. Nguyen, X. Xiao, and S. Pirk. Humembr: Learning human routines for predictive embodied navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2026.
- [4] M. F. Ginting, D.-K. Kim, X. Meng, A. M. Reinke, B. J. Krishna, N. Kayhani, O. Peltzer, D. Fan, A. Shaban, S.-K. Kim, M. Kochenderfer, A. akbar Agha-mohammadi, and S. Omidshafiei. Enter the Mind Palace: Reasoning and Planning for Long-term Active Embodied Question Answering. In *9th Annual Conference on Robot Learning*, 2025.
- [5] S. Saxena, B. Buchanan, C. Paxton, P. Liu, B. Chen, N. Vaskevicius, L. Palmieri, J. Francis, and O. Kroemer. GraphEQA: Using 3D Semantic Scene Graphs for Real-time Embodied Question Answering. In *9th Annual Conference on Robot Learning*, 2025.
- [6] H. Zhang, N. Savaliya, F. Siddiqui, and E. Sachdeva. FAST-EQA: Efficient Embodied Question Answering with Global and Local Region Relevancy. In *Winter Conference on Applications of Computer Vision (WACV)*, 2026, 2026.
- [7] G. Monaci, R. S. Rezende, R. Deffayet, G. Csurka, G. Bono, H. Déjean, S. Clinchant, and C. Wolf. RANa: Retrieval-Augmented Navigation. *Transactions on Machine Learning Research*, 2025.
- [8] H. Wu, Z.-W. Chen, G.-H. Tian, Q. Ma, and M.-L. Jiao. Item Ownership Relationship Semantic Learning Strategy for Personalized Service Robot. *Int. J. Autom. Comput.*, 17(3):390–402, June 2020. ISSN 1476-8186.
- [9] Y. Hu, Y. Ge, T. Yang, and X. Sun. An Interactive Learning Framework for Item Ownership Relationship in Service Robots. *Human Factors in Robots, Drones and Unmanned Systems*, 2023.
- [10] Y. Dai, R. Peng, S. Li, and J. Chai. Think, act, and ask: Open-world interactive personalized robot navigation. *ICRA*, 2024.
- [11] A. Majumdar, F. Xia, B. Ichter, D. Batra, and L. Guibas. FindThis: Language-Driven Object Disambiguation in Indoor Environments. In J. Tan, M. Toussaint, and K. Darvish, editors, *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pages 1335–1347. PMLR, 06–09 Nov 2023.
- [12] L. Barsellotti, R. Bigazzi, M. Cornia, L. Baraldi, and R. Cucchiara. Personalized Instance-based Navigation Toward User-Specific Objects in Realistic Environments. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 11228–11250. Curran Associates, Inc., 2024.

- [13] M. Khanna, R. Ramrakhya, G. Chhablani, S. Yenamandra, T. Gervet, M. Chang, Z. Kira, D. S. Chaplot, D. Batra, and R. Mottaghi. GOAT-Bench: A Benchmark for Multi-Modal Lifelong Navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16373–16383, June 2024.
- [14] J. Kim, G.-C. Kang, J. Kim, S. Yang, M. Jung, and B.-T. Zhang. PGA: Personalizing Grasping Agents with Single Human-Robot Interaction. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9277–9284, 2024.
- [15] K. Paramonov, J.-X. Zhong, U. Michieli, J. Moon, and M. Ozay. Swiss DINO: Efficient and Versatile Vision Framework for On-device Personal Object Search. *IROS*, 2024.
- [16] Y. Luo, X. Zheng, G. Li, S. Yin, H. Lin, C. Fu, J. Huang, J. Ji, F. Chao, J. Luo, and R. Ji. Video-RAG: Visually-aligned Retrieval-Augmented Long Video Comprehension. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.
- [17] Z. Pan, Q. Wu, H. Jiang, X. Luo, H. Cheng, D. Li, Y. Yang, C.-Y. Lin, H. V. Zhao, L. Qiu, and J. Gao. SeCom: On Memory Construction and Retrieval for Personalized Conversational Agents. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [18] Y. Du, B. Wang, Y. He, B. Liang, B. Wang, Z. Li, L. Gui, J. Z. Pan, R. Xu, and K.-F. Wong. MemGuide: Intent-Driven Memory Selection for Goal-Oriented Multi-Session LLM Agents. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(36):30584–30592, Mar. 2026.
- [19] W. Xu, Z. Liang, K. Mei, H. Gao, J. Tan, and Y. Zhang. A-Mem: Agentic Memory for LLM Agents. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.
- [20] P. Chhikara, D. Khant, S. Aryan, T. Singh, and D. Yadav. Mem0: Building Production-Ready AI Agents with Scalable Long-Term Memory. *arXiv preprint arXiv:2504.19413*, 2025.
- [21] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie. A convnet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [22] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*, 2021.
- [23] M. Tan and Q. Le. EfficientNetV2: Smaller Models and Faster Training. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10096–10106. PMLR, 18–24 Jul 2021.
- [24] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, F. Massa, D. Haziza, L. Wehrstedt, J. Wang, T. Darcet, T. Moutakanni, L. Sentana, C. Roberts, A. Vedaldi, J. Tolan, J. Brandt, C. Couprie, J. Mairal, H. Jégou, P. Labatut, and P. Bojanowski. DINOv3, 2025. *arXiv preprint arXiv:2508.10104*.
- [25] K. Zhou and T. Xiang. Torchreid: A Library for Deep Learning Person Re-Identification in Pytorch. *arXiv preprint arXiv:1910.10093*, 2019.
- [26] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang. Omni-Scale Feature Learning for Person Re-Identification. In *ICCV*, 2019.
- [27] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang. Learning Generalisable Omni-Scale Representations for Person Re-Identification. *TPAMI*, 2021.

- [28] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang. ByteTrack: Multi-Object Tracking by Associating Every Detection Box. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.
- [29] Y. Du, Z. Zhao, Y. Song, Y. Zhao, F. Su, T. Gong, and H. Meng. Strongsort: Make deepsort great again. *IEEE Transactions on Multimedia*, 2023.
- [30] M. M. Khalil, S. N. Saleh, N. S. Tawfik, and M. N. Elagamy. EF-StrongSORT: An Enhanced Feature StrongSORT Model for Multi-Object Tracking. *IEEE Access*, 13:53608–53620, 2025.
- [31] C. Yuan, G. Zhang, C. Ma, T. Zhang, and G. Niu. From poses to identity: Training-free person re-identification via feature centralization. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24409–24418, 2025.
- [32] H. Ye, J. Zhao, Y. Zhan, W. Chen, L. He, and H. Zhang. Person Re-Identification for Robot Person Following With Online Continual Learning. *IEEE Robotics and Automation Letters*, 9(11):9151–9158, 2024.
- [33] R. Ha, S. Jiang, B. Li, B. Pan, Y. Zhu, J. Zhang, X. Zhu, S. Gong, and J. Wang. Multi-modal Multi-platform Person Re-Identification: Benchmark and Method. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10251–10261, October 2025.
- [34] F. Yingying, L. Jie, H. Jie, Z. Yukang, T. Lei, and J. Jiayi. MDReID: Modality-Decoupled Learning for Any-to-Any Multi-Modal Object Re-Identification. In *Advances in Neural Information Processing Systems*, 2025.
- [35] X. Xu, Z. Liu, W. Zhou, Y. Gao, J. Cao, Y. Wang, J. Luo, and D. Zhang. STMI: Segmentation-Guided Token Modulation with Cross-Modal Hypergraph Interaction for Multi-Modal Object Re-Identification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(14):11433–11441, Mar. 2026.
- [36] Y. Wang, Y. Liu, A. Zheng, and P. Zhang. DeMo: Decoupled Feature-Based Mixture of Experts for Multi-Modal Object Re-Identification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(8):8141–8149, Apr. 2025.
- [37] Y. Feng, J. Li, C. Xie, L. Tan, and J. Ji. Multi-Modal Object Re-identification via Sparse Mixture-of-Experts. In *Forty-second International Conference on Machine Learning*, 2025.
- [38] Y. Wang, X. Liu, T. Yan, Y. Liu, A. Zheng, P. Zhang, and H. Lu. MambaPro: Multi-Modal Object Re-Identification with Mamba Aggregation and Synergistic Prompt. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.
- [39] Y. Wang, Y. Lv, P. Zhang, and H. Lu. IDEA: Inverted Text with Cooperative Deformable Aggregation for Multi-Modal Object Re-Identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
- [40] T. Lei, S. Yin, Q. Chen, Y. Peng, and Y. Liu. Open-Vocabulary HOI Detection with Interaction-aware Prompt and Concept Calibration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 23945–23957, October 2025.
- [41] Q. Lei, B. Wang, and T. Robby T. HOLa: Zero-Shot HOI Detection with Low-Rank Decomposed VLM Feature Adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2025.
- [42] B. Yang, Y. Zhang, H.-Y. Zhou, and S. Yang. No More Sibling Rivalry: Debiasing Human-Object Interaction Detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22707–22717, October 2025.

- [43] F. Tonini, A. Conti, L. Vaquero, C. Beyan, and E. Ricci. Towards Unconstrained Human-Object Interaction, 2026. *arXiv preprint arXiv:2604.14069*.
- [44] Z. Zhang, Y. Shi, L. Yang, S. Ni, Q. Ye, and J. Wang. OpenHOI: Open-World Hand-Object Interaction Synthesis with Multimodal Large Language Model. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.
- [45] G. Jocher and J. Qiu. Ultralytics yolo11, 2024. URL <https://github.com/ultralytics/ultralytics>.
- [46] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky. BoT-SORT: Robust Associations Multi-Pedestrian Tracking. *arXiv preprint arXiv:2206.14651*, 2022.
- [47] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4685–4694, 2019.
- [48] S. Macenski and I. Jambrecic. Slam toolbox: Slam for the dynamic world. *Journal of Open Source Software*, 6(61):2783, 2021. doi:10.21105/joss.02783.
- [49] Boston Dynamics. Spot: The agile mobile robot, 2026. Boston Dynamics product documentation.
- [50] S. Macenski, F. Martín, R. White, and J. Ginés Clavero. The marathon 2: A navigation system. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. URL <https://github.com/ros-planning/navigation2>.
- [51] S. Macenski, T. Moore, D. V. Lu, A. Merzlyakov, and M. Ferguson. From the desks of ros maintainers: A survey of modern & capable mobile robotics algorithms in the robot operating system 2. *Robotics and Autonomous Systems*, 2023.
- [52] S. Macenski, M. Booker, J. Wallace, and T. Fischer. Cost-aware kinematically feasible planning for mobile and surface robotics. *IEEE Robotics and Automation Practice*, 2026.
- [53] S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge, W. Ge, Z. Guo, Q. Huang, J. Huang, F. Huang, B. Hui, S. Jiang, Z. Li, M. Li, M. Li, K. Li, Z. Lin, J. Lin, X. Liu, J. Liu, C. Liu, Y. Liu, D. Liu, S. Liu, D. Lu, R. Luo, C. Lv, R. Men, L. Meng, X. Ren, X. Ren, S. Song, Y. Sun, J. Tang, J. Tu, J. Wan, P. Wang, P. Wang, Q. Wang, Y. Wang, T. Xie, Y. Xu, H. Xu, J. Xu, Z. Yang, M. Yang, J. Yang, A. Yang, B. Yu, F. Zhang, H. Zhang, X. Zhang, B. Zheng, H. Zhong, J. Zhou, F. Zhou, J. Zhou, Y. Zhu, and K. Zhu. Qwen3-VL Technical Report. *arXiv preprint arXiv:2511.21631*, 2025.
- [54] X. Zhang, A. Konno, T. Yamasaki, and L. Xiao. Enhancing Lightweight Vision Language Models through Group Competitive Learning for Socially Compliant Navigation. *arXiv preprint arXiv:2603.11447*, 2026.
- [55] Y. Zhang, Y. Chen, C. Liu, Z. Ding, J. Xu, S. Zou, J. Liao, J. Hu, X. Ren, X. Zhang, Y. Liu, H. Shi, Z. Tang, H. Sun, R. Cui, K. Wu, W. Liu, Y. Xu, Y. Zhang, Y. Wang, S. Hu, J. Lu, N. T. Chan, Y. Wu, Z. Liu, X. Hou, Y. Dai, J. Tang, and X. Ju. Pelican-Unify 1.0: A Unified Embodied Intelligence Model for Understanding, Reasoning, Imagination and Action. *arXiv preprint arXiv:2605.15153*, 2026.
- [56] Z. Huang, M. Liu, X. Lin, M. Zhu, C. Zhao, Z. Du, Y. Lin, X. Li, Y. Jia, H. Zhong, H. Chen, and C. Shen. NoTVLA: Semantics-Preserving Robot Adaptation via Narrative Action Interfaces. *arXiv preprint arXiv:2510.03895*, 2025.
- [57] X. Zeng, X. Zhou, Y. Li, J. Shi, T. Li, L. Chen, L. Ren, and Y.-L. Li. Diagnose, correct, and learn from manipulation failures via visual symbols. *arXiv preprint arXiv:2512.02787*, 2025.

- [58] M. Deitke, E. VanderBilt, A. Herrasti, L. Weihs, J. Salvador, K. Ehsani, W. Han, E. Kolve, A. Farhadi, A. Kembhavi, and R. Mottaghi. ProcTHOR: Large-Scale Embodied AI Using Procedural Generation. In *NeurIPS*, 2022. Outstanding Paper Award.
- [59] M. Khanna, Y. Mao, H. Jiang, S. Haresh, B. Shacklett, D. Batra, A. Clegg, E. Undersander, A. X. Chang, and M. Savva. Habitat synthetic scenes dataset (hssd-200): An analysis of 3d scene scale and realism tradeoffs for objectgoal navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16384–16393, June 2024.
- [60] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu, M. Martin, T. Nagarajan, I. Radosavovic, S. K. Ramakrishnan, F. Ryan, J. Sharma, M. Wray, M. Xu, E. Z. Xu, C. Zhao, S. Bansal, D. Batra, V. Cartillier, S. Crane, T. Do, M. Doulaty, A. Erapalli, C. Feichtenhofer, A. Fragomeni, Q. Fu, C. Fuegen, A. Gebreselasie, C. González, J. Hillis, X. Huang, Y. Huang, W. Jia, W. Khoo, J. Kolár, S. Kottur, A. Kumar, F. Landini, C. Li, Y. Li, Z. Li, K. Mangalam, R. Modhugu, J. Munro, T. Murrell, T. Nishiyasu, W. Price, P. R. Puentes, M. Ramazanova, L. Sari, K. K. Somasundaram, A. Southerland, Y. Sugano, R. Tao, M. Vo, Y. Wang, X. Wu, T. Yagi, Y. Zhu, P. Arbeláez, D. Crandall, D. Damen, G. M. Farinella, B. Ghanem, V. K. Ithapu, C. V. Jawahar, H. Joo, K. Kitani, H. Li, R. A. Newcombe, A. Oliva, H. S. Park, J. M. Rehg, Y. Sato, J. Shi, M. Z. Shou, A. Torralba, L. Torresani, M. Yan, and J. Malik. Ego4d: Around the world in 3, 000 hours of egocentric video. *CoRR*, abs/2110.07058, 2021. URL <https://arxiv.org/abs/2110.07058>.
- [61] J. Yang, S. Liu, H. Guo, Y. Dong, X. Zhang, S. Zhang, P. Wang, Z. Zhou, B. Xie, Z. Wang, B. Ouyang, Z. Lin, M. Cominelli, Z. Cai, Y. Zhang, P. Zhang, F. Hong, J. Widmer, F. Gringoli, L. Yang, B. Li, and Z. Liu. Egolife: Towards egocentric life assistant, 2025. URL <https://arxiv.org/abs/2503.03803>.
- [62] Y. Kant, A. Ramachandran, S. Yenamandra, I. Gilitschenski, D. Batra, A. Szot, and H. Agrawal. Housekeep: Tidying virtual households using commonsense reasoning, 2022.
- [63] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.
- [64] M. Minderer, A. Gritsenko, and N. Houlsby. Scaling Open-Vocabulary Object Detection. *arXiv preprint arXiv:2306.09683*, 2024.
- [65] C. E. Duchon. Lanczos Filtering in One and Two Dimensions. *Journal of Applied Meteorology and Climatology*, 18(8):1016 – 1022, 1979.
- [66] X. Wang, L. Xie, C. Dong, and Y. Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *International Conference on Computer Vision Workshops (ICCVW)*, 2021.
- [67] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou. Training data-efficient image transformers & distillation through attention. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR, 18–24 Jul 2021.
- [68] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 18–24 Jul 2021.

- [69] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollar, and C. Feichtenhofer. SAM 2: Segment Anything in Images and Videos. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [70] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.
- [71] L. Wei, S. Zhang, W. Gao, and Q. Tian. Person transfer gan to bridge domain gap for person re-identification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 79–88, 2018.
- [72] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

A Object and Person Clustering

The system uses a fixed set of parameters for persistent person and object memory. These parameters control embedding-based re-identification, face matching, nearest-neighbor scoring, spatial and temporal gating, and periodic cluster consolidation. They define when observations are linked to existing clusters, when face evidence is accepted, and when cached or spatially constrained matches are allowed. The same values are used across all experiments unless stated otherwise. Table 5 summarizes these parameters.

Table 5: Clustering parameters used at inference time for person/object re-identification.

Parameter	Symbol	Value	Description
<i>Embedding similarity</i>			
Visual embedding threshold	τ_{emb}	0.643	Cosine similarity; runtime override (code default 0.85)
Face embedding threshold	τ_{face}	0.60	Cosine similarity; runtime override (code default 0.68)
Similarity metric	—	Cosine	pgvector <=> cosine distance; $\text{sim} = 1 - \text{dist}$
Centroid update	—	Arithmetic mean	SQL AVG(<code>oo.embedding</code>) over cluster observations
Attribute part embeddings	—	false	Part-level attribute embeddings are disabled
<i>K-NN match scoring</i>			
K-NN neighbors	k	12	Neighbors considered for match scoring
Average weight	w	0.65	Weight on $\bar{s}_k$ vs. s_k^{max} in score
Min. score margin	δ	0.03	Required gap to second-best cluster
Face score margin	δ_{face}	0.03	Min. margin over second-best face match
Min. neighbor hits	h_{min}	1	Min. neighbors that must exceed threshold
Min. observation quality	q_{min}	0.05	Quality floor for observations entering ReID
<i>Spatial / temporal constraints</i>			
Max. ReID spatial distance	d_{max}	2.0 m	Spatial veto if observation > 2 m from cluster centroid
Track ID TTL	Δt_{track}	8.0 s	Time-only; track-to-object mapping expires after 8 s
Face-track TTL	$\Delta t_{\text{face-track}}$	20.0 s	Face-to-track association expires after 20 s
Face-track DB lookup window	$\Delta t_{\text{face-db}}$	180.0 s	Window for looking up face tracks in the database
Recent detection cache TTL	Δt_{cache}	30.0 s	Hardcoded; suppresses duplicate object creation
Recent detection cache distance	d_{cache}	1.0 m	$d_{\text{max}} \times 0.5$; spatial radius for cache lookup
Static object position distance	d_{static}	0.8 m	Fallback match radius for non-person static objects
Position source preference	—	depth	Preferred source for observation position estimates
<i>Face detection and embedding</i>			
Minimum face size	—	16 px	Minimum accepted face crop size
Minimum face confidence	—	0.45	Minimum confidence for face detections
Maximum faces per person	—	1	Maximum number of face crops used per person detection
Face crop margin	—	0.24	Relative margin added around face crop
InsightFace model	—	buffalo_l	Face embedding model
InsightFace device	—	cuda	Device used for face embedding inference
InsightFace detection size	—	640	Input detection size for InsightFace
<i>Cluster consolidation (periodic merge)</i>			
Consolidation interval	N_{consol}	120 obs	Runs every 120 new observations
Consolidation similarity threshold	τ_{consol}	0.643	Cosine similarity for merge candidates (code default 0.80)
Min. observations per cluster	n_{min}	2	Clusters below this are skipped
Max. pairs evaluated per run	P_{max}	6	Caps merge computation per consolidation pass
Min. observation quality (consolidation)	—	0.05	Same as main ReID quality floor

B Structured Database Tools

The agents interact with the memory system through a fixed set of structured tools. Each tool defines typed inputs and outputs, allowing the agent to search tracked entities, inspect observations, query interaction histories, and resolve navigation targets. These tools also provide access to the robot’s current pose, enabling decisions to be grounded in the robot’s spatial context. Navigation is executed only after a target has been resolved to a reachable position on the map.

B.1 Tool Calls for the User Agent

The user agent uses the tools listed in Table 6 to answer ownership-related navigation commands. These tools allow the agent to identify people and objects, retrieve their observation histories, inspect object locations, query interaction evidence, and execute navigation commands once a valid target has been selected.

Table 6: Structured tools exposed to the user agent for object search, inspection, interaction retrieval, scene understanding, and navigation.

Tool	Input	Output	Purpose
<i>Discovery and search</i>			
<code>search_objects_by_class_id</code>	<code>class_id</code> , <code>name</code> , <code>min_obs</code> , <code>limit</code>	object list	Find tracked objects or people by class or name
<code>list_objects</code>	<code>class_name</code> (opt.), <code>limit</code>	object list	Enumerate all tracked objects, optionally filtered
<code>find_person_by_name</code>	<code>person_name</code>	person entity	Resolve a named person to their latest location
<i>Object inspection</i>			
<code>get_object_observations</code>	<code>object_id</code> , time filters, <code>limit</code>	observation history	Retrieve the location history of one entity
<code>get_object_last_location</code>	<code>object_id</code>	latest pose	Retrieve the most recent recorded location
<code>get_object_first_location</code>	<code>object_id</code>	earliest pose	Retrieve the first recorded location of an object
<code>get_object_summary</code>	<code>object_id</code>	spatial summary	Summarize all places where an object has been observed
<code>get_object_image</code>	<code>object_id</code>	image data	Fetch the latest cropped object image for display
<i>Location and navigation</i>			
<code>get_robot_location</code>	—	current pose + room	Ground decisions relative to the robot’s current position
<code>get_room_navigation_target</code>	<code>room_name</code>	navigable pose	Resolve a room to a reachable entry point
<code>get_objects_in_room</code>	<code>room_name</code> , <code>limit</code>	object list	List entities currently assigned to a room
<code>move_to_position</code>	<code>x</code> , <code>y</code> , <code>reason</code> , <code>standoff_m</code> , <code>yaw</code>	nav status	Execute navigation to a resolved map target
<i>Interaction history</i>			
<code>get_interaction_events</code>	<code>person</code> , <code>object</code> , <code>time</code> , <code>room</code> , <code>action</code> , <code>limit</code>	event list	Retrieve interaction evidence using filters
<code>get_object_interactions</code>	<code>object_id</code> , <code>action</code> , <code>limit</code> , <code>sort</code>	interaction list	Retrieve the history of people interacting with an object
<code>get_object_person_interactions</code>	<code>object_id</code> , <code>action</code> (opt.)	aggregated list	List people who interacted with an object, including counts
<code>get_person_interaction_summary</code>	<code>person_name</code> , <code>other</code> , <code>time</code> , <code>room</code> , <code>limit</code>	co-participants	Rank interaction partners for a person
<code>get_recent_interactions</code>	<code>minutes</code> , <code>limit</code>	recent events	Retrieve recent interactions with room context
<i>Room and scene context</i>			
<code>get_current_scene_description</code>	<code>minutes</code> (opt.)	scene + context	Retrieve the latest caption, objects, people, and interactions

B.2 Tool Calls for the Curiosity-Driven Agent

The curiosity-driven agent uses a smaller set of tools to decide which room should be visited next. These tools provide information about room visitation history, recent interaction density, and rooms that have not been observed for a longer period. This allows the agent to prioritize navigation targets based on spatial coverage and expected information gain. Table 7 provides an overview.

Table 7: Structured tools used by the curiosity-driven navigation strategy to select the next room to visit.

Tool	Input	Output	Purpose
<code>get_room_visit_history</code>	<code>limit</code>	per-room statistics	Retrieve visit counts, last-visit times, and event counts per room
<code>get_room_event_density</code>	<code>hours</code> , <code>limit</code>	event counts	Count recent interaction events per room over a time window
<code>get_stale_rooms</code>	<code>threshold_min</code> , <code>limit</code>	stale room list	Identify rooms that have not been visited recently

C Offline Reasoning Benchmark

The synthetic office benchmark is used to evaluate reasoning abilities over structured robot memory. It consists of a memory trace for one simulated workday. The trace contains rooms, people, objects, timestamped observations, and human–object interaction events. Each object and person has a stable identity. Each observation is associated with a time and a metric position in the map. The question set is derived from this trace and is used only for final evaluation. Table 8 shows the distribution of questions across evaluation categories.

Table 8: Test set composition.

Evaluation Category	Questions
Interaction	33
Location	22
Navigation	13
Lookup	6
Ownership	5
Total	79

How it is used. The benchmark is used to evaluate the user agent. For each question, the agent can call structured tools to access memory. The tools expose observations, interactions, entities, rooms,

and navigation targets. The agent then returns a natural-language answer. Answers are scored with semantic similarity against manually specified reference answers. All questions are held out until final evaluation.

Question categories. Interaction questions ask about human–object interaction events. Location questions ask where people or objects were observed. Navigation questions require resolving a room, person, or object to a reachable goal. Lookup questions require retrieving an entity by name or identifier. Ownership questions require inferring likely ownership from interaction and observation histories.

Example questions. Examples include:

- Where was notebook_002 at 16:20 on 2026-04-27?
- Who picked up headphones_001?
- What interaction happened with whiteboard_001 at 09:58?
- Who used notebook_001, and how many times?
- Navigate to the person who used the coffee machine most frequently.
- Navigate to Simba’s bottle.

C.1 Robustness Evaluation Details

To evaluate robustness to perception errors, we perturb the results returned by the memory database at query time while leaving the underlying database unchanged. This isolates the effect of perception errors from errors introduced during data collection or database construction. Unless otherwise specified, perturbations are applied independently for each evaluation seed.

Table 9 summarizes the simulated failure modes. Detection dropout randomly removes a fraction of observation rows, simulating missed detections. False-positive injection adds observations with incorrect identities, simulating spurious detections or identity hallucinations. Embedding noise adds Gaussian noise to stored feature vectors, reducing the reliability of similarity retrieval. Track fragmentation splits observations associated with an identity into synthetic track identifiers and removes the associated identity names, simulating loss of temporal identity consistency.

Face occlusion removes the available face image information from selected observations. Temporal jitter randomly perturbs timestamps within the specified range, simulating synchronization errors. Action noise replaces interaction/action labels with incorrect labels, while class confusion swaps object categories between visually similar classes and removes the associated captions. Finally, systematic identity swapping applies a consistent bijection between identities throughout the evaluation, simulating a persistent identity-assignment error.

Table 9: Perturbations used in the robustness evaluation.

Failure Mode	Parameter	Perturbation
Detection Dropout	$p \in \{0.4, 0.6, 0.8\}$	Randomly remove observation rows
False-Positive Injection	$p \in \{0.1, 0.5\}$	Add observations with incorrect identities
Embedding Noise	$\sigma = 0.10$	Add Gaussian noise to embeddings
Track Fragmentation	$p = 0.2$	Split tracks and remove identity names
Face Occlusion	$p = 0.5$	Remove face image information
Temporal Jitter	± 300 s	Perturb observation timestamps
Action Noise	$p = 0.3$	Replace action/interaction labels
Class Confusion	$p = 0.3$	Swap similar object classes
Systematic Identity Swap	$p = 0.5$	Apply a consistent identity permutation

D Real-World Tasks

The real-world experiments consist of five task setups. Each setup targets a specific capability of the robot memory system and the tool-using agent.

D.1 Detailed Tasks

Task 1: Room Navigation. The robot receives a target room name. The target room is selected from four different rooms. The robot starts from multiple positions across the environment. The agent must ground the room name to a spatial region on the map. It must then resolve a reachable navigation goal inside the room. The robot must navigate at least 1 m into the target room.

Task 2: Object Relocation. The robot observes a ball at five different positions. A person is observed with the ball at two different locations. The agent must re-identify both the ball and the person across observations. It must infer the correct owner–object association. It must then query the latest known position of the ball and navigate to it.

Task 3: Owner-based Distinction. The robot observes two bottles in different locations. One person interacts with one bottle in the kitchen, another person interacts with the other bottle at his workplace. The agent must distinguish between the two bottles and the two people. It must infer the correct owner–object association. It must then query the latest known position of the target bottle and navigate to it.

Task 4: Visual Ambiguity. The robot observes two visually similar backpacks in different locations. Each backpack is associated with a different person through prior interaction evidence. The agent must distinguish between the two backpacks and the two people. It must infer the correct owner–object association. It must then query the latest known position of the target backpack and navigate to it.

Task 5: Ownership-Preserving Update. The robot observes two visually similar backpacks in different locations. Each backpack is associated with a different person through prior interaction evidence. One backpack is later moved to a new location. The agent must re-identify the moved backpack without changing its ownership association. It must distinguish between the two backpacks and the two people. It must infer the correct owner–object association. It must then query the latest known position of the target backpack and navigate to it.

D.2 Success criterion

A run is counted as successful if the agent selects the correct target entity or room, and the robot reaches and stops at the corresponding navigation goal. For room navigation, the robot must stop at least 1 m inside the target room. For object-based tasks, the robot must navigate to the latest known position of the correct target object. A run is counted as failed if the agent selects the wrong entity, queries an outdated location, or navigates to the wrong goal.

D.3 Execution protocol

The task was conducted over 10 hours of naturalistic office activity, involving 10 people and 20 objects across four rooms. Each task is evaluated over 10 runs. Runs differ in the robot’s starting pose and, where applicable, the target instance. The memory state is initialized with the observations and interactions required by the task. The agent receives only the task instruction. It must use the structured tools to query memory and resolve the navigation target.

E Curiosity-Driven Evaluation

The synthetic office evaluation set for the curiosity-driven agent contains 15 worlds at two spatial scales: nine 6-room worlds and six 24-room worlds that evaluate the same generator under increasing scale and movement-budget pressure. All worlds span the same three simulated days, 09:00–14:00 at 5 min intervals, which means 61 steps per day and 183 decision steps per run. Table 10 gives an overview of all the worlds.

All logs are generated by a procedural office simulator in which each person movement is explicitly represented by an intended destination, an optionally carried personal object, and a departure cue rendered in the origin room. Consequently, person–object–place consistency is ensured by construction. Each room–timestep pair is also automatically annotated with a change label and an interest label, distinguishing ownership-relevant events from transient distractors.

Table 10: Dataset specification. The two floor plans differ in size, population and object count; every other component of the simulator is shared.

Aspect	6-room floor plan	24-room floor plan
Rooms	6: kitchen, storage, office 1–3, meeting room	24: 16 offices, 3 meeting rooms, kitchen, lounge, storage, printer room, server room
Room types	social, utility, office, meeting	social, utility, office, meeting
People	8 employees, each with a home office and a role	40 employees, each with a home office and a role
Objects	12 personal objects with a fixed owner (laptop, red cup, backpack, water bottle, notebook, pen, headphones, coffee mug, tablet, jacket, yoga mat, camera), plus activity-spawned temporary items	40 personal objects with a fixed owner, plus activity-spawned temporary items
Fixtures	per room, e.g. coffee machine, fridge, microwave (kitchen); projector, whiteboard, conference table (meeting room)	as left, plus printer, paper shelf, server rack
Geometry	pairwise room distances $d \in \{1, 2, 3, 4\}$ (adjacent offices $d = 1$, kitchen–office 3 $d = 4$)	same distance model over the larger plan
Time resolution	5-minute intervals, 09:00–14:00	5-minute intervals, 09:00–14:00
Structure	3 days $\times$ 61 steps = 183 steps; 6 $\times$ 183 = 1098 room observations	3 days $\times$ 61 steps = 183 steps; 24 $\times$ 183 = 4392 room observations

Activity model. At each timestep, present occupants perform activities sampled from room-specific templates, as summarized in Table 11. Some activities spawn temporary items which persist with a time-to-live of 15–30 min. These dynamic elements are incorporated directly into the natural-language scene descriptions rendered for the VLM, without exposing ground-truth labels.

Table 11: Activities and durations by room type.

Room Type	Activity	Duration
Office	Works at their desk	20–200 min
	Types on their laptop	15–120 min
	Reads a document	10–40 min
	Joins a video call	15–120 min
	Writes notes	10–30 min
	Checks their phone	5 min
Kitchen	Makes coffee	5–10 min
	Prepares a snack	5–15 min
	Washes a mug	–
	Chats near the counter	–
	Fills a water bottle	–
Meeting Room	Sits at the table	–
	Writes on the whiteboard	–
	Points at the screen	–
	Takes notes	–
Utility Room	Organizes boxes	–
	Looks for a tool	–
	Checks inventory	–
	Sweeps the floor	–

Movement is a stochastic decision layer. At the end of an activity, a person moves with a room-type-dependent base probability (0.85 kitchen, 0.70 meeting, 0.60 office, 0.55 utility), modulated by time-of-day multipliers ($\times 1.8$ at lunch, $\times 1.5$ in the late afternoon, $\times 0.7$ in the first half hour) and by per-world room multipliers. On top of this layer, every world runs a small scripted schedule of 3 events per day (e.g. stand-up, lunch, client presentation, architecture discussion).

Room priors. Worlds differ in how informative the marginal visit statistics are. *Hot* priors are the default (kitchen and home offices dominate), *mild* priors partially flatten the affinities, and *flat* priors make every room a priori equally attractive, so visit counts carry almost no information and the language cues become the only usable signal.

E.1 Change Definition

Ground truth is computed per room by comparing consecutive observations of that room, independently of where the robot actually is.

Entity-level changes. For each room we compare the set of people and the set of objects present at timestep t with those at $t + 1$. Any addition or removal is a change. Its severity is classified as *major* if a person entered or left, a fixture changed, or at least two objects changed; *minor* if exactly one object changed or only the activities changed. *Major* changes are the ownership-relevant ones.

Activity-level changes. Activity-level changes are detected by comparing the interactions between consecutive observations, i.e. the set of `subject:action:target` tuples observed in a room. A room is labelled as changed at the activity level if this set differs between t and $t + 1$. The severity of an activity-only change is *minor*.

E.2 The 15 Evaluation Worlds

All logs are generated by a custom office simulator that explicitly defines each person’s destination, optional personal object, and departure cue. This ensures consistency between people, objects, and rooms. The simulator also automatically labels each room at each timestep by change type (none, minor, or major) and interest (ownership-relevant or transient distractor). The released logs are generated with fixed random seeds and world layouts. Table 12 gives an overview of the generated worlds.

Table 12: Evaluation worlds. Each world spans 3 simulated days, 09:00–14:00 at 5 min intervals (183 decision steps/run). **Obs.** = logged room observations; **Maj.** = ownership-relevant scene changes; **I/D** = v2 interesting/distractor changes; **Ep.** = scripted intent episodes; p_{head} = probability that a departure cue names its destination. Occupancy priors are *hot*, *mild*, or *flat*. Each world is fixed; seeds vary policy randomness only.

World	R	P	O	Obs.	Maj.	I/D	Ep.	p_{head}	Prior / property
<i>6-room worlds</i>									
diffused	6	8	21	1098	365	–	–	–	hot; diffused activity
regime	6	8	24	1098	140	–	–	–	hot; daily regime shift, sparse changes
adversarial	6	8	18	1098	296	–	–	–	hot; moving hotspots
base	6	8	9	1098	234	121/267	–	–	hot; v1 parity under v2 generator
distractor	6	8	10	1098	280	135/273	–	–	hot; 4× churn in kitchen/meeting, mostly distractor-class
intent_cued	6	8	15	1098	250	132/256	14	–	hot; ownership cues precede change
intent_cued_heading	6	8	14	1098	275	161/245	19	0.5	hot; cues additionally name destination
agent_cued_6room	6	8	15	1098	255	137/240	22	0.6	mild; denser cues, reduced hotspot advantage
agent_cued_flat_6room	6	8	11	1098	261	147/255	19	0.7	flat; cues are the only predictive signal
<i>24-room worlds</i>									
scaled_base_24room	24	40	44	4392	1041	672/978	–	–	hot; scale without intent episodes
scaled_24room	24	40	45	4392	1046	665/968	31	–	hot; scale with intent episodes
scaled_distractor_24room	24	40	41	4392	1087	632/1056	–	–	hot; scale + 4× distractor hotspots
scaled_heading_24room	24	40	47	4392	1055	660/970	32	0.5	hot; scale + heading cues
agent_cued_24room	24	40	45	4392	1081	684/970	44	0.6	mild; denser cues at scale
agent_cued_flat_24room	24	40	44	4392	1107	695/999	54	0.7	flat; cue-following at scale

Notes. R = rooms, P = people, O = objects, Obs. = $R \times 183$. I/D reports interesting/distractor changes; these labels are unavailable in legacy v1 logs. “–” denotes not applicable. Occupancy priors: *hot* = kitchen/home offices dominant; *mild* = partially flattened; *flat* = rooms equally attractive a priori, making language cues the primary predictive signal. Intent episodes are cue→change events within 10–30 min, with 10% head fakes.

E.3 Change Distribution

The worlds differ mainly in *where* and *when* changes concentrate. Table 13 gives the full per-room distribution of major changes for the nine 6-room worlds and Table 14 the concentration summary for the six 24-room worlds, in both cases as 3-day totals with the Gini coefficient of the per-room counts as a concentration index (0 = uniform, 1 = all changes in one room). Storage is naturally quiet in every hot-prior world (2–16 changes over three days) but is never systematically suppressed, and the flat-prior worlds deliberately remove that gradient (agent_cued_flat_6room: Gini 0.10, storage as busy as an office).

Table 13: Major changes per room, 3-day totals, 6-room worlds. *Gini* is the concentration of changes across the six rooms.

World	Kitchen	Storage	Office 1	Office 2	Office 3	Meeting	Gini
default	96	16	58	70	43	82	0.24
regime	32	13	28	24	25	18	0.15
adversarial	70	3	54	66	48	55	0.22
base_v2	62	9	50	36	28	49	0.25
distractor	88	2	38	46	43	63	0.30
intent_cued	65	10	43	39	31	62	0.25
intent_cued_heading	63	10	57	39	32	74	0.26
agent_cued_6room	57	16	51	32	34	65	0.22
agent_cued_flat_6room	37	35	44	39	47	59	0.10

Table 14: Concentration of major changes in the 24-room worlds, 3-day totals. With 24 rooms no single room dominates: the busiest room carries 7–14% of all major changes, which is what makes camping strategies expensive at scale.

World	Changes	Gini	Three busiest rooms (share of major changes)
scaled_base_24room	1041	0.21	kitchen 11%, meeting room 1 7%, office 3 5%
scaled_24room	1046	0.27	kitchen 11%, meeting room 1 7%, meeting room 3 6%
scaled_distractor_24room	1087	0.28	kitchen 14%, meeting room 1 9%, meeting room 3 7%
scaled_heading_24room	1055	0.24	kitchen 10%, meeting room 1 7%, meeting room 2 6%
agent_cued_24room	1081	0.19	meeting room 1 8%, meeting room 2 7%, kitchen 7%
agent_cued_flat_24room	1107	0.12	meeting room 1 7%, lounge 6%, meeting room 3 5%

E.4 Metrics

Three of the four reported quantities are *conditional on presence*, which we state explicitly because it determines how the numbers must be read.

- **Recall** — major-change recall *given presence*: $TP_{\text{major}} / (TP_{\text{major}} + FN_{\text{major}})$ over the major changes the robot was present for. It answers “when the robot was there, did it notice?”, not “how many changes did it see?”.
- **F1** — change-detection F1 over the steps the robot actually observed, i.e. of the per-decision judgement “has this room changed since I last saw it?”. Also conditional on presence, and therefore *maximised by camping*: a policy that never leaves the busiest room scores very well on it. Every table below therefore prints a move budget and an unconditional column next to it.
- **Det.** — agent-relative detection rate: the fraction of cases in which the room differed from the agent’s *own* last observation of it and the detector said so.
- **Yield** — change events observed per room move.
- **Moves** — room moves per 3-day run (of at most 182).
- **Cov.** (unconditional) — fraction of the reachable change events the strategy witnessed, taking the number of steps at which *any* room changes as 100%.
- **Missed** (unconditional) — change events never observed: the room was not visited after the change, or was visited without a detection.

E.5 Protocol

We compare nine strategies across all 15 worlds. To isolate navigation performance, the policy controls every stay-or-move decision while COOL’s VLM change detector is shared along with its perceived statistics. Each table cell represents the mean across 10 seeds, where each seed alters policy randomness while keeping the environment fixed.

E.5.1 Per-World Results

Each table reports, for one world: major-change recall given presence (*Recall*), change-detection F1 with its standard deviation over seeds, changes observed per move (*Yield*), the confusion counts of the change/no-change decision (*TP*, *FP*, *FN*), the move budget (*Moves*), the changes observed per run (*Changes*), unconditional coverage (*Cov.*) and the number of change events never observed (*Missed*). Bold marks the best F1, the highest yield, the smallest move budget and the fewest missed changes in each world.

Table 15: Navigation-strategy comparison on default. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	89.0%	67.3% ± 3.6	0.43	64	52	10	182	78	45.9%	19
Round-Robin	93.2%	81.4% ± 0.8	0.92	74	28	6	91	84	49.7%	25
Frequency	89.3%	72.8% ± 5.4	0.61	71	42	11	139	85	50.4%	42
Staleness	90.4%	67.1% ± 2.4	0.44	68	58	9	182	81	47.7%	18
Active Mapping	88.0%	66.8% ± 2.4	0.44	68	57	10	182	81	47.8%	15
Greedy Hazard	86.6%	79.4% ± 1.0	0.65	78	28	12	144	93	55.3%	76
Entropy	85.8%	91.7% ± 3.4	14.49	78	1	13	6	94	55.7%	383
Beta-Entropy	93.3%	73.3% ± 3.7	0.46	72	44	8	182	84	49.5%	47
COOL	97.1%	92.5% ± 1.6	2.34	69	9	2	32	74	43.8%	58

Table 16: Navigation-strategy comparison on regime. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	91.3%	72.9% ± 2.8	0.24	36	20	6	182	44	33.0%	6
Round-Robin	93.9%	71.6% ± 1.0	0.40	30	19	4	91	36	26.9%	5
Frequency	85.5%	73.1% ± 2.0	0.34	26	12	8	104	35	26.4%	48
Staleness	90.7%	72.2% ± 3.4	0.22	33	21	5	182	40	29.8%	7
Active Mapping	95.2%	72.9% ± 3.8	0.22	33	20	5	182	40	29.7%	7
Greedy Hazard	80.4%	83.0% ± 2.8	0.61	20	2	6	45	28	20.6%	176
Entropy	81.9%	85.5% ± 1.8	3.71	25	0	8	10	35	26.3%	191
Beta-Entropy	90.3%	74.9% ± 3.7	0.32	44	17	12	182	58	43.6%	9
COOL	96.2%	89.8% ± 3.7	1.45	40	8	1	30	43	32.3%	47

Table 17: Navigation-strategy comparison on adversarial. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	90.7%	71.4% ± 3.8	0.33	52	36	5	182	60	47.4%	19
Round-Robin	86.1%	65.2% ± 0.5	0.57	43	39	8	91	52	40.9%	21
Frequency	88.3%	77.7% ± 2.8	0.56	63	26	10	135	75	59.2%	53
Staleness	91.6%	68.9% ± 2.1	0.33	52	42	5	182	59	46.7%	11
Active Mapping	89.5%	69.8% ± 1.6	0.33	52	40	5	182	60	47.2%	14
Greedy Hazard	88.5%	85.4% ± 4.9	0.55	47	9	8	104	57	44.9%	229
Entropy	88.1%	90.8% ± 2.1	10.45	62	0	12	7	76	60.1%	283
Beta-Entropy	88.8%	78.8% ± 4.6	0.40	61	23	10	182	73	57.5%	43
COOL	96.8%	91.6% ± 4.1	2.24	50	7	1	24	54	42.4%	102

Table 18: Navigation-strategy comparison on base_v2. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	87.9%	63.7% ± 3.4	0.35	51	48	11	182	64	40.3%	11
Round-Robin	92.5%	63.1% ± 3.1	0.62	42	37	12	91	57	35.8%	26
Frequency	90.5%	66.7% ± 4.1	0.50	55	40	14	140	70	44.6%	43
Staleness	88.9%	64.7% ± 3.6	0.36	58	58	5	182	65	41.1%	14
Active Mapping	88.8%	66.2% ± 1.8	0.36	59	55	5	182	65	41.2%	14
Greedy Hazard	91.1%	71.4% ± 3.8	0.65	49	24	15	100	65	41.4%	121
Entropy	79.1%	74.0% ± 7.4	9.09	42	2	25	8	68	43.2%	319
Beta-Entropy	94.1%	68.8% ± 4.2	0.36	53	37	10	182	65	41.0%	45
COOL	99.3%	81.8% ± 4.0	2.27	39	8	10	22	51	32.3%	130

Table 19: Navigation-strategy comparison on distractor. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	92.8%	66.7% ± 4.8	0.37	59	52	6	182	68	43.1%	12
Round-Robin	92.8%	63.3% ± 1.0	0.59	43	40	10	91	54	34.4%	13
Frequency	92.0%	74.2% ± 4.6	0.69	74	38	14	129	89	56.8%	45
Staleness	94.1%	60.7% ± 3.3	0.34	56	68	4	182	62	39.4%	8
Active Mapping	93.2%	59.2% ± 2.7	0.34	55	69	6	182	62	39.7%	8
Greedy Hazard	88.8%	76.8% ± 4.9	0.65	74	30	15	139	90	57.5%	138
Entropy	81.4%	77.1% ± 13.3	13.25	58	2	25	6	85	54.0%	323
Beta-Entropy	93.0%	68.5% ± 1.8	0.38	57	43	10	182	68	43.6%	37
COOL	98.6%	83.0% ± 4.4	2.07	44	8	8	26	55	34.8%	53

Table 20: Navigation-strategy comparison on intent_cued. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	93.7%	66.0% ± 4.2	0.34	54	49	6	182	63	39.4%	12
Round-Robin	93.5%	67.5% ± 3.9	0.65	47	36	9	91	59	37.1%	15
Frequency	89.5%	70.2% ± 3.8	0.56	59	37	13	132	74	46.7%	44
Staleness	90.1%	62.5% ± 3.5	0.34	54	59	6	182	63	39.4%	10
Active Mapping	90.3%	61.8% ± 1.4	0.34	54	61	6	182	62	39.2%	9
Greedy Hazard	88.1%	73.0% ± 3.6	0.68	45	21	12	87	59	37.4%	74
Entropy	88.2%	82.8% ± 4.8	11.58	56	1	22	7	80	50.9%	308
Beta-Entropy	92.1%	68.3% ± 3.6	0.34	52	40	9	182	63	39.4%	30
COOL	98.3%	80.3% ± 6.2	1.92	38	10	8	25	49	30.7%	87

Table 21: Navigation-strategy comparison on intent_cued_heading. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	91.0%	66.3% ± 4.5	0.37	56	47	9	182	67	41.9%	14
Round-Robin	94.1%	65.8% ± 2.3	0.59	44	39	8	91	54	34.0%	34
Frequency	91.9%	72.0% ± 3.6	0.58	64	36	13	135	79	49.6%	44
Staleness	85.4%	63.1% ± 2.9	0.36	54	53	10	182	66	41.8%	11
Active Mapping	86.4%	60.9% ± 2.6	0.37	55	60	10	182	67	42.3%	10
Greedy Hazard	90.1%	77.9% ± 2.2	0.67	66	25	13	120	80	50.6%	121
Entropy	85.1%	82.1% ± 3.4	14.10	60	1	24	6	86	54.1%	319
Beta-Entropy	85.3%	67.7% ± 3.7	0.39	55	40	13	182	70	44.1%	51
COOL	97.9%	82.7% ± 4.6	2.36	44	8	10	24	56	35.5%	95

Table 22: Navigation-strategy comparison on agent_cued_6room. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	95.3%	66.7% ± 3.4	0.35	55	49	6	182	64	41.8%	15
Round-Robin	97.8%	68.1% ± 0.8	0.64	47	35	9	91	58	38.2%	25
Frequency	94.8%	69.9% ± 3.8	0.51	58	41	9	135	69	45.5%	35
Staleness	91.1%	62.8% ± 3.7	0.36	57	60	6	182	65	42.7%	6
Active Mapping	92.3%	65.2% ± 2.8	0.37	59	57	6	182	67	44.0%	7
Greedy Hazard	91.2%	71.9% ± 4.1	0.61	42	24	9	88	54	35.5%	118
Entropy	83.2%	78.4% ± 5.0	9.58	41	2	19	6	62	41.0%	314
Beta-Entropy	94.2%	67.8% ± 4.4	0.34	51	40	9	182	62	40.7%	35
COOL	99.2%	80.2% ± 6.7	2.20	39	9	10	23	52	33.9%	113

Table 23: Navigation-strategy comparison on agent_cued_flat_6room. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	89.1%	65.1% ± 3.9	0.39	59	53	9	182	70	45.1%	10
Round-Robin	99.3%	63.7% ± 0.5	0.52	41	41	6	91	47	30.1%	16
Frequency	91.8%	68.3% ± 4.8	0.46	54	41	9	141	66	42.0%	35
Staleness	90.5%	59.2% ± 2.7	0.35	56	71	6	182	64	40.9%	6
Active Mapping	86.8%	57.5% ± 2.2	0.34	52	70	7	182	62	39.5%	5
Greedy Hazard	85.9%	69.1% ± 2.6	0.62	39	26	9	82	50	32.3%	135
Entropy	85.6%	79.8% ± 5.0	10.82	44	2	21	6	67	43.0%	335
Beta-Entropy	92.0%	65.3% ± 2.9	0.36	53	45	11	182	66	42.0%	36
COOL	99.2%	81.1% ± 4.9	2.39	51	11	12	28	66	42.4%	102

Table 24: Navigation-strategy comparison on scaled_base_24room. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	90.4%	62.1% ± 3.6	0.38	55	62	5	182	68	37.7%	227
Round-Robin	95.1%	71.4% ± 1.4	0.77	50	27	12	91	70	38.7%	211
Frequency	92.7%	67.8% ± 6.0	0.45	61	51	6	168	75	41.5%	584
Staleness	89.2%	61.8% ± 2.9	0.41	58	67	6	182	74	40.8%	148
Active Mapping	91.7%	60.1% ± 3.0	0.38	56	70	4	182	69	38.3%	142
Greedy Hazard	94.9%	80.8% ± 6.7	0.68	67	20	8	122	83	45.6%	1422
Entropy	82.8%	78.0% ± 7.2	3.18	48	5	20	25	78	43.2%	1565
Beta-Entropy	91.3%	75.1% ± 3.7	0.40	52	24	11	182	73	40.2%	965
COOL	99.5%	84.2% ± 5.1	2.32	50	6	13	30	70	39.0%	1033

Table 25: Navigation-strategy comparison on scaled_24room. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	92.1%	61.5% ± 3.3	0.37	54	62	5	182	68	37.5%	269
Round-Robin	92.8%	67.1% ± 1.7	0.57	37	29	7	91	52	28.7%	327
Frequency	95.9%	69.1% ± 4.9	0.46	63	50	5	167	77	42.3%	568
Staleness	92.4%	61.5% ± 3.8	0.38	57	68	4	182	69	38.0%	150
Active Mapping	93.6%	62.5% ± 3.1	0.38	59	67	4	182	70	38.6%	157
Greedy Hazard	88.7%	76.7% ± 9.7	0.64	57	20	10	116	74	40.6%	1277
Entropy	79.4%	75.4% ± 4.6	2.90	42	5	22	24	71	39.1%	1559
Beta-Entropy	91.3%	75.2% ± 3.2	0.38	52	22	12	182	70	38.5%	936
COOL	99.4%	84.4% ± 4.9	1.98	54	9	12	38	75	41.6%	730

Table 26: Navigation-strategy comparison on scaled_distractor_24room. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	91.5%	62.2% ± 3.1	0.39	56	63	6	182	71	39.0%	263
Round-Robin	95.7%	67.6% ± 1.2	0.74	50	38	10	91	67	37.1%	252
Frequency	96.2%	72.3% ± 6.3	0.49	70	48	4	167	82	45.2%	594
Staleness	92.3%	60.9% ± 3.8	0.39	58	70	4	182	71	39.3%	154
Active Mapping	95.1%	63.2% ± 3.8	0.40	60	66	4	182	72	40.0%	152
Greedy Hazard	94.1%	84.4% ± 6.5	0.69	88	22	8	152	105	57.8%	1353
Entropy	75.4%	72.9% ± 7.3	2.79	38	6	22	25	69	38.2%	1615
Beta-Entropy	94.5%	76.8% ± 4.0	0.39	54	23	10	182	71	39.1%	979
COOL	99.8%	87.6% ± 6.7	2.21	56	7	10	34	74	41.2%	937

Table 27: Navigation-strategy comparison on scaled_heading_24room. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	91.0%	60.3% ± 4.4	0.36	52	63	5	182	65	36.0%	246
Round-Robin	95.2%	65.3% ± 2.5	0.69	43	36	9	91	63	34.8%	270
Frequency	91.5%	66.5% ± 4.5	0.45	61	53	8	169	76	41.9%	535
Staleness	93.3%	60.9% ± 3.8	0.38	56	67	5	182	69	38.1%	158
Active Mapping	92.3%	60.0% ± 4.9	0.36	54	67	5	182	66	36.4%	153
Greedy Hazard	93.0%	75.5% ± 4.9	0.63	49	22	9	104	65	36.1%	1442
Entropy	81.2%	78.7% ± 4.9	2.78	43	6	17	25	68	37.7%	1559
Beta-Entropy	93.6%	74.0% ± 3.6	0.37	48	23	10	182	66	36.7%	1052
COOL	99.2%	85.9% ± 2.9	1.96	50	7	10	36	70	38.7%	819

Table 28: Navigation-strategy comparison on agent_cued_24room. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	88.2%	61.1% ± 5.3	0.37	54	63	6	182	67	37.2%	244
Round-Robin	89.9%	67.1% ± 3.7	0.76	46	32	13	91	69	38.1%	255
Frequency	90.4%	66.5% ± 4.7	0.44	59	52	7	168	74	40.7%	583
Staleness	90.4%	61.3% ± 2.4	0.38	57	66	5	182	69	38.2%	152
Active Mapping	92.3%	61.2% ± 3.3	0.37	55	65	5	182	67	37.0%	152
Greedy Hazard	91.5%	76.8% ± 4.1	0.61	62	26	11	131	80	44.5%	1236
Entropy	82.7%	79.2% ± 5.7	3.11	50	6	19	24	76	42.2%	1574
Beta-Entropy	93.1%	74.1% ± 1.8	0.40	51	24	12	182	72	39.7%	972
COOL	99.8%	84.1% ± 4.0	1.90	57	10	12	41	77	42.7%	713

Table 29: Navigation-strategy comparison on agent_cued_flat_24room. Mean over 10 seeds; $\pm$ is the sample standard deviation of F1 over seeds.

Strategy	Recall	F1	Yield	TP	FP	FN	Moves	Changes	Cov.	Missed
Random	92.5%	61.6% ± 3.9	0.38	55	64	5	182	68	37.5%	272
Round-Robin	90.5%	69.5% ± 1.3	0.76	50	34	10	91	69	37.9%	221
Frequency	91.2%	63.7% ± 3.9	0.42	56	56	7	169	70	38.7%	586
Staleness	86.0%	58.4% ± 4.2	0.39	56	73	6	182	70	38.7%	129
Active Mapping	88.3%	60.0% ± 4.1	0.38	57	70	6	182	70	38.5%	139
Greedy Hazard	92.5%	74.5% ± 3.7	0.65	47	21	11	98	63	34.8%	1489
Entropy	77.9%	75.8% ± 7.6	2.71	39	6	19	24	66	36.4%	1626
Beta-Entropy	90.4%	71.0% ± 4.1	0.39	50	28	12	182	72	39.3%	1032
COOL	99.8%	86.4% ± 4.1	2.28	55	6	12	32	74	40.7%	982

E.6 Ablations

Table 30 removes the navigator, while the destination is taken as the argmax of the code-computed room-score table instead of being chosen by the VLM. Also it shows the results if one of the three signals are removed from the prompt.

Table 30: Ablations of COOL, per world: change-detection F1 (%) with the move budget in parentheses, mean over 10 seeds. All variants share the same detector and the same dwell policy. Mean F1 *rises* when a signal is removed, because F1 is conditional on presence and the ablated variants collapse into camping: at 7.7 moves the *all three* variant enters 22% of rooms and never observes 812 changes per run, against 30 moves, 96% of rooms and 400 missed for the full policy. The signals therefore buy worst-case robustness and coverage, not mean F1: the worst case is adversarial, where removing staleness drops F1 from 91.6% at 24 moves to 62.3% at 6 moves.

World	COOL (full)	Score	argmax	— hotspots	— expected change	— staleness	— all three
default	92.5 (32)	87.5 (81)	91.7 (42)		93.6 (26)	93.9 (6)	95.2 (3)
regime	89.8 (30)	89.8 (93)	90.8 (44)		89.6 (24)	97.5 (8)	97.1 (4)
adversarial	91.6 (24)	83.0 (109)	90.8 (35)		88.4 (20)	62.3 (6)	61.3 (4)
base_v2	81.8 (22)	71.2 (110)	77.6 (42)		82.5 (19)	85.3 (6)	85.7 (3)
distractor	83.0 (26)	74.0 (102)	80.3 (46)		78.7 (25)	88.8 (4)	90.4 (2)
intent_cued	80.3 (25)	73.9 (108)	80.6 (44)		80.8 (29)	89.3 (6)	90.0 (2)
intent_cued_heading	82.7 (24)	75.3 (107)	82.0 (40)		83.8 (25)	85.2 (6)	86.4 (4)
agent_cued_6room	80.2 (23)	74.4 (109)	78.5 (43)		81.7 (28)	85.1 (7)	86.5 (5)
agent_cued_flat_6room	81.1 (28)	71.4 (104)	77.8 (44)		79.4 (31)	87.5 (9)	88.0 (2)
scaled_base_24room	84.2 (30)	71.9 (111)	82.3 (49)		85.9 (22)	87.2 (20)	85.9 (16)
scaled_24room	84.4 (38)	74.6 (111)	83.0 (46)		86.0 (23)	85.5 (17)	84.8 (16)
scaled_distractor_24room	87.6 (34)	80.5 (92)	86.3 (45)		88.2 (18)	85.4 (22)	85.3 (17)
scaled_heading_24room	85.9 (36)	72.2 (111)	84.4 (39)		87.4 (19)	86.6 (15)	85.9 (12)
agent_cued_24room	84.1 (41)	72.6 (112)	82.3 (47)		86.7 (24)	86.3 (18)	85.3 (14)
agent_cued_flat_24room	86.4 (32)	74.0 (110)	82.6 (53)		86.4 (20)	86.5 (18)	86.7 (14)
Mean F1	85.0	76.4	83.4		85.3	86.2	86.3
Mean moves	30	105	44		24	11	8
Rooms entered	0.96	1.00	0.98		0.86	0.27	0.22
Changes missed	400	107	302		542	798	812

E.7 Isolating the destination choice

In Table 4 each baseline uses its own detector prompt and its own rule for when to move, so the comparison is between complete exploration systems. To isolate the destination choice we re-ran every baseline with COOL’s detector and COOL’s stay-or-move decision, varying only the rule that selects the next room.

Granting this decision improves every baseline, by 0.120 F1 on average (range 0.056–0.162). The effect is not simply “move less”: it releases Entropy from its statistical camp (13 → 45 moves) while restraining the sweeping policies (182 → 74 moves) and the hazard policy (112 → 8).

Under this condition COOL’s destination rule is no longer the strongest on F1 (Table X): a hazard-greedy destination reaches 90.7% F1 at 8 moves, and four rules clear 80% on both precision and recall. Those gains come either from camping — the hazard-greedy variant leaves 715 changes unobserved, and Entropy 668 — or from moving two to four times more. COOL leaves the fewest changes unobserved among strategies with a comparable movement budget (389 at 28 moves). We therefore attribute most of the benefit to the detector’s stay-or-move decision, which is transferable, and the remainder to the navigator’s additional restraint on movement: argmax retains the same stay decision yet still makes 102 moves against COOL’s 28, because it always relocates once the detector releases the robot, whereas the navigator frequently elects to remain.

Destination rule	Recall	Prec.	F1	Moves	Missed
Greedy Hazard	84.9%	98.2%	90.7%	8	715
COOL navigator	87.6%	87.0%	86.8%	28	389
Entropy	87.0%	87.1%	86.5%	45	668
Frequency	90.5%	81.8%	85.7%	67	408
Beta-Entropy (cost)	89.9%	76.8%	82.7%	125	358
Beta-Entropy	88.9%	75.2%	81.3%	61	285
Round-Robin	89.6%	73.8%	80.8%	70	124
Staleness	90.9%	72.6%	80.6%	74	127
Active Mapping	90.9%	71.9%	80.2%	75	122
Random	91.0%	70.9%	79.5%	104	163
Score argmax	91.6%	68.6%	78.3%	102	95

F Perception and Re-Identification Ablations

F.1 Object Detection

We compare several object detection models to identify the most suitable model for low-quality images captured by the robot. The evaluation is conducted on a self-labeled dataset collected in the office environment in which the robot is deployed. For each model, we report the best-performing confidence threshold based on multiple test runs.

Since the downstream task requires reliable detections, precision is prioritized over recall. In this setting, false positives are more problematic than false negatives, because incorrect detections may lead to erroneous robot behavior. Therefore, we use the $F_{0.5}$ score as the main selection criterion, as it gives more weight to precision than recall. Table 31 shows the results.

Table 31: Object detection performance on the self-labeled office dataset. The confidence threshold is selected based on the best performance observed across multiple test runs.

Model	Conf	TP	FP	FN	Precision	Recall	$F_{0.5}$
GroundingDINO [63]	0.55	11.0	2.0	147.0	0.846	0.070	0.262
YOLO11m-seg	0.55	88.0	2.3	70.0	0.975	0.557	0.848
YOLO11l-seg	0.55	88.0	3.0	70.0	0.967	0.557	0.843
YOLO11s-seg	0.40	87.0	3.0	71.0	0.967	0.551	0.840
OWLv2 [64]	0.40	19.3	0.3	138.7	0.974	0.122	0.402

F.2 Person Re-Identification

We evaluate several feature extractors for person re-identification on a self-labeled test set containing 20 identities. For each model and input variant, the threshold with the best performance across multiple test runs is reported. The input variants include the original image, Lanczos upscaling [65], Real-ESRGAN upscaling [66], and a preprocessing variant using white balancing and brightness adjustment.

The results are shown in Table 32. Overall, the fine-tuned OSNet model achieves the best clustering performance. In particular, using the original whole-image input yields the highest Pairwise F1 score, purity, and precision, while Real-ESRGAN upscaling produces the highest recall.

Table 32: Person re-identification clustering ablation results on a test set with 20 identities. For each model and input variant, the best-performing threshold across multiple test runs is reported. Bold values indicate the best result per metric, and underlined values indicate the second-best result.

Model	Input	Threshold	Clusters	Pairwise F1	Purity	Precision	Recall
ConvNeXt-Tiny [21]	Whole Image	0.500	60	0.151	0.500	0.159	0.144
ConvNeXt-Tiny [21]	Lanczos Upscaled	0.500	60	0.147	0.496	0.154	0.141
ConvNeXt-Tiny [21]	Whole Image + WB + Bright	0.500	85	0.122	0.572	0.166	0.097
ConvNeXt-Tiny [21]	Real-ESRGAN Upscaled	0.500	93	0.105	0.572	0.129	0.089
DINOv3 [24]	Lanczos Upscaled	0.400	6	0.180	0.228	0.099	0.925
DINOv3 [24]	Real-ESRGAN Upscaled	0.450	6	0.177	0.228	0.098	0.893
DINOv3 [24]	Whole Image + WB + Bright	0.400	7	0.170	0.228	0.095	0.833
DINOv3 [24]	Whole Image	0.400	5	0.168	0.225	0.094	0.813
EfficientNetV2 [23]	Real-ESRGAN Upscaled	0.500	89	0.146	0.580	0.128	0.170
EfficientNetV2 [23]	Whole Image	0.500	88	0.145	0.609	0.224	0.107
EfficientNetV2 [23]	Lanczos Upscaled	0.500	89	0.145	0.616	0.227	0.106
EfficientNetV2 [23]	Whole Image + WB + Bright	0.500	95	0.142	0.634	0.209	0.108
ViT [22]	Whole Image + WB + Bright	0.500	44	0.178	0.388	0.128	0.292
ViT [22]	Real-ESRGAN Upscaled	0.450	13	0.168	0.254	0.100	0.534
ViT [22]	Whole Image	0.450	<u>18</u>	0.164	0.279	0.099	0.482
ViT [22]	Lanczos Upscaled	0.450	13	0.161	0.261	0.096	0.483
DeiT [67]	Whole Image + WB + Bright	0.500	38	0.160	0.428	0.125	0.223
DeiT [67]	Whole Image	0.500	39	0.143	0.417	0.120	0.175
DeiT [67]	Lanczos Upscaled	0.550	61	0.135	0.482	0.134	0.136
DeiT [67]	Real-ESRGAN Upscaled	0.500	50	0.127	0.417	0.097	0.185
OSNet [27]	Whole Image + WB + Bright	0.625	53	0.358	0.739	0.481	0.286
OSNet [27]	Lanczos Upscaled	0.625	61	0.358	0.779	0.556	0.264
OSNet [27]	Whole Image	0.625	63	0.358	0.783	0.552	0.265
OSNet [27]	Real-ESRGAN Upscaled	0.625	56	0.345	0.757	0.467	0.274
OSNet Fine-tuned	Whole Image	0.643	19	0.831	0.895	0.806	0.858
OSNet Fine-tuned	Real-ESRGAN Upscaled	0.643	17	<u>0.815</u>	0.895	0.775	<u>0.860</u>
OSNet Fine-tuned	Lanczos Upscaled	0.643	<u>18</u>	0.815	0.884	<u>0.786</u>	0.845
OSNet Fine-tuned	Whole Image + WB + Bright	0.643	19	0.806	<u>0.888</u>	0.770	0.847

F.3 Object Re-Identification

We evaluate object re-identification using a labeled object dataset collected in the robot’s office environment. The dataset composition is summarized in Table 33. Since the detections contain some annotation noise, several identity labels appear under multiple detector classes. For example, the same physical object may be associated with different detector categories across samples.

The clustering results are reported in Table 34. We compare different feature extraction models using three input variants: the original whole image, Lanczos upscaling [65], and the original image with white balance and brightness adjustment. The results are averaged over multiple runs using greedy clustering and are reported as mean $\pm$ standard deviation.

Table 33: Class-wise composition of the labeled object dataset. Some identity labels appear under multiple detector classes due to annotation noise. All person names were replaced with pseudonymous labels.

Class	Unique IDs	Labelled boxes	Identities
Chair	6	59	Chair-Kitchen (25), Chair-Alex (14), Chair-Blake (13), Chair-Casey (3), Chair-Dana (3), 3D-Printer (1)
Couch	2	20	Couch (19), Chair-Blake (1)
TV	3	7	Desktop-Alex (5), Desktop-Evan (1), Desktop-Casey (1)
Sports Ball	1	8	Volleyball (8)
Bed	1	6	Couch (6)
Oven	2	4	Chair-Kitchen (3), Rubbish (1)
Refrigerator	1	2	Rubbish (2)
Bottle	2	3	Dish-Soap (2), Bottle-Cola (1)
Cat / Car / Mouse / Toilet	1 each	1 each	Couch, Backpack-Evan, Volleyball, Rubbish-Blake

Note: Some identity labels appear under multiple detector classes, e.g., Couch is labelled under both Couch and Bed detections, and Chair-Kitchen under Chair and Oven detections. This reflects annotation noise in the ground truth.

Table 34: Object re-identification clustering results grouped by model and input variant. Values report mean $\pm$ standard deviation over runs using greedy clustering. Bold values indicate the best result per metric and underlined values indicate the second-best result.

Model	Version	F1	Purity	Precision	Recall
CLIP [68]	Whole Image	0.284 $\pm$ 0.067	0.297 $\pm$ 0.115	0.171 $\pm$ 0.063	0.973 $\pm$ 0.065
CLIP [68]	Whole Image + WB + Bright	0.283 $\pm$ 0.058	0.294 $\pm$ 0.106	0.167 $\pm$ 0.044	<u>0.990 $\pm$ 0.024</u>
CLIP [68]	Lanczos Upscaled	0.271 $\pm$ 0.033	0.280 $\pm$ 0.084	0.159 $\pm$ 0.028	<u>0.975 $\pm$ 0.071</u>
ConvNeXt-Tiny [21]	Whole Image	0.734 $\pm$ 0.102	0.914 $\pm$ 0.057	0.878 $\pm$ 0.083	0.659 $\pm$ 0.151
ConvNeXt-Tiny [21]	Lanczos Upscaled	<u>0.731 $\pm$ 0.102</u>	0.899 $\pm$ 0.073	0.868 $\pm$ 0.101	0.666 $\pm$ 0.160
ConvNeXt-Tiny [21]	Whole Image + WB + Bright	0.720 $\pm$ 0.127	0.929 $\pm$ 0.048	0.901 $\pm$ 0.071	0.628 $\pm$ 0.164
DeiT [67]	Whole Image	0.551 $\pm$ 0.222	0.929 $\pm$ 0.045	0.867 $\pm$ 0.074	0.460 $\pm$ 0.242
DeiT [67]	Lanczos Upscaled	0.555 $\pm$ 0.224	0.929 $\pm$ 0.046	0.870 $\pm$ 0.075	0.463 $\pm$ 0.242
DeiT [67]	Whole Image + WB + Bright	0.483 $\pm$ 0.173	0.946 $\pm$ 0.042	0.882 $\pm$ 0.080	0.365 $\pm$ 0.172
DINOv3 [24]	Whole Image Embedding	0.723 $\pm$ 0.202	0.878 $\pm$ 0.075	0.816 $\pm$ 0.107	0.744 $\pm$ 0.280
DINOv3 [24]	Lanczos Upscaled	0.722 $\pm$ 0.202	0.868 $\pm$ 0.080	0.804 $\pm$ 0.107	0.755 $\pm$ 0.288
DINOv3 [24]	Whole Image + WB + Bright	0.720 $\pm$ 0.198	0.885 $\pm$ 0.076	0.824 $\pm$ 0.104	0.735 $\pm$ 0.290
EfficientNetV2 [23]	Whole Image	0.458 $\pm$ 0.193	0.938 $\pm$ 0.049	0.857 $\pm$ 0.086	0.354 $\pm$ 0.203
EfficientNetV2 [23]	Lanczos Upscaled	0.456 $\pm$ 0.197	0.941 $\pm$ 0.044	0.871 $\pm$ 0.079	0.347 $\pm$ 0.201
EfficientNetV2 [23]	Whole Image + WB + Bright	0.465 $\pm$ 0.152	0.960 $\pm$ 0.037	0.914 $\pm$ 0.061	0.333 $\pm$ 0.146
OSNet [27]	Whole Image	0.565 $\pm$ 0.177	0.794 $\pm$ 0.156	0.637 $\pm$ 0.259	0.723 $\pm$ 0.301
OSNet [27]	Lanczos Upscaled	0.557 $\pm$ 0.177	0.793 $\pm$ 0.160	0.637 $\pm$ 0.266	0.712 $\pm$ 0.299
OSNet [27]	Whole Image + WB + Bright	0.404 $\pm$ 0.149	0.649 $\pm$ 0.296	0.562 $\pm$ 0.364	0.631 $\pm$ 0.323
SAM2 [69]	Whole Image	0.361 $\pm$ 0.000	0.469 $\pm$ 0.000	0.222 $\pm$ 0.000	0.973 $\pm$ 0.000
SAM2 [69]	Lanczos Upscaled	0.449 $\pm$ 0.000	0.478 $\pm$ 0.000	0.289 $\pm$ 0.000	0.998 $\pm$ 0.000
SAM2 [69]	Whole Image + WB + Bright	0.361 $\pm$ 0.000	0.469 $\pm$ 0.000	0.222 $\pm$ 0.000	0.973 $\pm$ 0.000
Swin [70]	Whole Image	0.655 $\pm$ 0.212	0.926 $\pm$ 0.057	0.905 $\pm$ 0.064	0.564 $\pm$ 0.225
Swin [70]	Lanczos Upscaled	0.655 $\pm$ 0.213	0.926 $\pm$ 0.056	<u>0.907 $\pm$ 0.067</u>	0.564 $\pm$ 0.226
Swin [70]	Whole Image + WB + Bright	0.591 $\pm$ 0.227	0.938 $\pm$ 0.055	<u>0.903 $\pm$ 0.078</u>	0.493 $\pm$ 0.232
ViT [22]	Whole Image	0.581 $\pm$ 0.224	0.828 $\pm$ 0.117	0.658 $\pm$ 0.176	0.633 $\pm$ 0.269
ViT [22]	Lanczos Upscaled	0.563 $\pm$ 0.200	0.816 $\pm$ 0.106	0.629 $\pm$ 0.161	0.629 $\pm$ 0.263
ViT [22]	Whole Image + WB + Bright	0.513 $\pm$ 0.228	0.900 $\pm$ 0.049	0.737 $\pm$ 0.083	0.459 $\pm$ 0.266

G Fine-Tuning

We fine-tune OSNet from an MSMT17-pretrained [71] checkpoint using manually labeled person detections collected from the Spot robot’s front fisheye camera. The training dataset contains 808 labeled detections from 19 identities. We split the dataset into 605 training samples and 203 validation samples. Training uses identity-balanced batches, AdamW [72], a cosine learning-rate schedule, and a combined cross-entropy and triplet loss. We select the best checkpoint at epoch 24 using validation pairwise F1, with validation loss as a tie-breaker.

Table 35: Training setup and hyperparameters used for fine-tuning OSNet on the person re-identification dataset.

Parameter	Value
Base checkpoint	MSMT17-pretrained OSNet x1.0
Camera source	Spot front fisheye camera
Input size	128×256 (W×H)
Feature dimension	512
Dataset size	808 labeled detections
Identities	19
Training samples	605
Validation samples	203
Best checkpoint	Epoch 24
Checkpoint selection	validation pairwise F1; validation loss as tie-breaker
Optimizer	AdamW
Learning-rate schedule	Cosine
Loss	Cross-Entropy (label smoothing 0.1) + triplet loss
Batch sampling	Identity-balanced
batch_size	32
bbox_margin	0.08
epochs	40
freeze_epochs	40 (early layers frozen for all epochs)
instances_per_id	4
lr	0.0003
min_samples_per_id	2
num_workers	2
seed	42
triplet_margin	0.3
triplet_weight	1.2
val_frac	0.25
weight_decay	0.0005