

Active Perception Meets Rule-Guided RL: A Two-Phase Approach for Precise Object Navigation in Complex Environments

Liang Qin^{1,2}, Min Wang^{3*}, Peiwei Li^{1,2}, Wengang Zhou^{1,2*}, and Houqiang Li^{1,2}

¹MoE Key Laboratory of Brain-inspired Intelligent Perception and Cognition, China

²University of Science and Technology of China, China

³Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, China

{qinliang6218, liipeiwei}@mail.ustc.edu.cn; wangmin@iai.ustc.edu.cn; {zhwg, lihq}@ustc.edu.cn;

Abstract

Object Goal Navigation (ObjectNav) in unknown environments presents significant challenges, particularly in Open-Vocabulary Mobile Manipulation (OVMM), where robots must efficiently explore large spaces, locate small objects, and accurately position themselves for subsequent manipulation. Existing approaches struggle to meet these demands: rule-based methods offer structured exploration but lack adaptability, while reinforcement learning (RL)-based methods enhance adaptability but fail to ensure effective long-term navigation. Moreover, both approaches often overlook precise stopping positions, which are critical for successful manipulation. To address these challenges, we propose APRR (Active Perception meets Rule-guided RL), a two-phase framework, which designs a new rule-guided RL policy for the exploration phase and a novel active target perception policy for the last-mile navigation phase. Inspired by human search behavior, our rule-guided RL policy enables efficient and adaptive exploration by combining structured heuristics with learning-based decision-making. In the last-mile navigation phase, we introduce an RL-based policy enhanced with active target perception, allowing the robot to refine its position dynamically based on real-time detection feedback. Experimental results demonstrate that APRR improves the success rate by 13%, significantly outperforming existing methods. Furthermore, real-world experiments validate the practicality and effectiveness of APRR in real-world mobile manipulation scenarios, offering a robust and adaptable solution for precise object navigation. The code is available at <https://github.com/qinliangql/APRR>.

*Corresponding author.

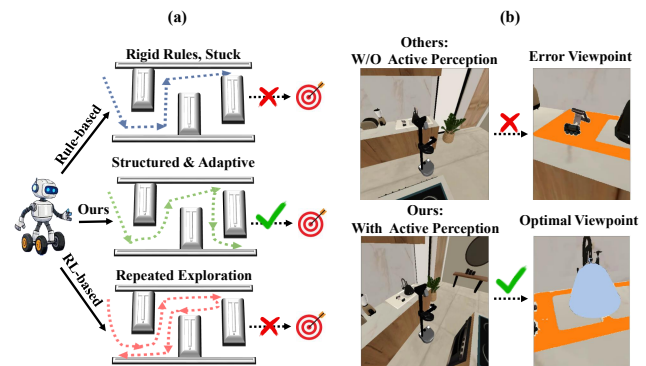


Figure 1. Comparison of our approach with existing methods. (a) illustrates the difference between our approach and rule-based and RL-based methods during the exploration phase. Our rule-guided RL strategy enables a structured yet adaptive search, effectively overcoming the limitations of previous approaches. (b) demonstrates the advantages of employing active perception in the last-mile navigation stage, where our method helps the agent position itself at an optimal viewpoint for subsequent manipulation.

1. Introduction

Navigation in unknown environments is extensively studied [1–3, 7, 12, 24] in the field of robotics, with most research focusing on moving from point A to point B. However, in real-world applications, particularly in household environments, robots must not only navigate efficiently but also position themselves correctly to perform downstream manipulation tasks. Open-Vocabulary Mobile Manipulation (OVMM) [27] extends this challenge by requiring an agent to execute instructions such as “Move the apple from the table to the sofa.” This involves a sequential process: first navigating to find the apple on the table, then grasping it, transporting it to the sofa, and finally placing it.

This work focuses on the most critical sub-task of OVMM—navigating to the object before grasping it, re-

ferred to as OVMM ObjectNav. While this task falls within the broader scope of Object Goal Navigation (ObjectNav), it differs significantly from conventional ObjectNav settings. Prior benchmarks such as HM3D [15] and MP3D [14] assume navigation targets that are large, visually salient, and easily detectable (e.g., beds, sofas) within moderate-scale indoor environments. In contrast, OVMM ObjectNav requires searching small, often occluded objects (e.g., apples, cups) in large, complex, and cluttered environments. More critically, the task requires not only locating the object but also ensuring that the robot stops at an appropriate position and orientation for subsequent interaction. These factors significantly increase the difficulty of efficient and precise navigation.

As shown in Fig. 1, existing approaches to this task are broadly categorized into rule-based and reinforcement learning (RL)-based methods, both of which have notable limitations. Rule-based approaches [11, 22], while offering structured and systematic exploration, often suffer from inefficiencies, rigid behavior, and an inability to adapt dynamically to the agent’s state, leading to failures such as getting stuck. In contrast, RL-based navigation methods [9] provide greater adaptability but lack long-term memory, resulting in inefficient exploration patterns where the agent repeatedly searches the same areas. More critically, existing approaches fail to explicitly address the problem of stopping at an optimal position for manipulation, often leading to suboptimal final poses that hinder subsequent interactions with the target object.

To address these challenges, we propose APRR (Active Perception Meets Rule-Guided RL), a novel two-phase framework that decouples OVMM ObjectNav into distinct exploration and last-mile navigation stages. In real life, humans do not solely rely on instruction instructions or prior experience but rather integrate both to effectively navigate unknown spaces. Inspired by this, we propose a rule-guided reinforcement learning strategy for the exploration stage. The policy aims to combine the strength of two fundamental components: following explicit guidance and leveraging prior experience. In other words, the robot uses rule-guided reinforcement learning to explore the environment efficiently, integrating structured exploration with adaptive RL capabilities to localize the target region.

In the last-mile navigation phase, we introduce a new reinforcement learning strategy for active target detection. This approach mirrors human behavior when reaching for an object, humans dynamically adjust their position and viewpoint based on feedback from their surroundings. Likewise, our model learns active target detection policy with real-time detection feedback—such as object masks, confidence scores, and spatial context—from a fine-tuned YOLO-World [4] detector combined with SAM2 segmentation [16]. This enables the robot to refine its position and

orientation, ensuring it stops at an optimal viewpoint for subsequent manipulation.

Experimental results demonstrate that APRR outperforms existing approaches, achieving a 13% improvement in success rate in challenging OVMM ObjectNav tasks. Furthermore, real-world experiments validate the practicality and effectiveness of our approach in physical robotic systems, demonstrating its robustness and adaptability for real-world mobile manipulation. By decoupling exploration from last-mile navigation, and integrating rule-guided reinforcement policy for exploration with active target detection for precise stopping, APRR offers a robust and adaptable solution for precise object navigation. This lays a strong foundation for more effective mobile manipulation in real-world environments.

2. Related Work

2.1. Open-Vocabulary Mobile Manipulation

The Open-Vocabulary Mobile Manipulation (OVMM) task is first formally defined in the Home-Robot [27] framework, encompassing not only open-vocabulary object goal navigation but also complex object interaction tasks such as grasping and placement. Since its introduction, numerous efforts have been made to tackle the OVMM challenge [26], with most approaches [9, 11] in this challenge extending rule-based baselines by refining heuristics and addressing corner cases. For instance, UniTeam [11] refined navigation heuristics to prevent infinite loops and improved object detection by adjusting confidence thresholds. Team KuzHum [9] enhanced the place skill policy through reward engineering and optimized the high-level heuristic to improve overall task success.

One notable work, PoLo [22] shares our focus on the ObjectNav sub-task of OVMM. This method utilizes prior knowledge of object co-occurrence patterns to estimate probable object locations, thereby enhancing navigation efficiency by reducing search time. However, this approach heavily relies on learned spatial priors, which may struggle to generalize to unseen environments. Moreover, PoLo does not address last-mile navigation, which is crucial for precise object localization and interaction. Our approach differs in that we simultaneously optimize both global exploration and explicit last-mile localization, leading to a more robust and adaptable ObjectNav solution within OVMM.

2.2. Object Goal Navigation

Object Goal Navigation (ObjectNav) has been extensively studied in the context of embodied AI, where the objective is to navigate toward a specific object category (e.g., bed, sofa) within structured indoor environments [14, 15, 20]. Existing approaches in this domain can be broadly classified into reinforcement learning (RL)-based and rule-based

methods.

RL-based approaches train policies through interactions with the environment, optimizing navigation efficiency via reward-driven learning [1, 12, 29]. While these methods exhibit adaptability, they often suffer from sample inefficiency and struggle in large-scale environments due to the absence of long-term memory, leading to repetitive and inefficient exploration. Conversely, rule-based approaches [3, 10, 28] rely on pre-defined heuristics to guide navigation, ensuring interpretable and consistent behavior. However, these methods cannot dynamically adapt to action outcomes, limiting their flexibility in unstructured environments.

Furthermore, conventional ObjectNav tasks [1, 12] typically assume targets that are large and easily detectable. In OVMM, however, the ObjectNav task requires locating small, occluded objects within substantially larger and more cluttered environments, thereby demanding both efficient global exploration and precise local positioning—challenges that remain inadequately addressed by existing methods.

2.3. Active Object Detection

Active object detection focuses on actively adjusting an agent’s viewpoint to improve perception and recognition. Several prior works explore this concept by enabling embodied agents to modify their position or orientation to obtain better visual information. Fang et al. [6] introduce an iterative viewpoint refinement strategy to enhance detection accuracy. Ding et al. [5] propose a decision-transformer-based framework for viewpoint selection, optimizing observation efficiency. Similarly, Shen et al. [19] present a collaborative model that jointly optimizes viewpoint adjustment and object detection. Other notable works, such as Kotar et al. [8] explore adaptive movement strategies to improve detection reliability.

While these works primarily focus on enhancing object detection through active viewpoint selection, our approach extends active perception to facilitate precise last-mile navigation. Instead of solely improving detection confidence, we leverage active perception techniques to refine the agent’s stopping position, ensuring an optimal pose for subsequent object interactions. By incorporating active object detection strategies into our last-mile navigation model, we effectively bridge the gap between active perception and navigation decision-making, leading to more robust and task-efficient precise object navigation.

3. Method

In this section, we introduce APRR (Active Perception Meets Rule-Guided RL), our novel two-phase framework designed for Object Navigation in Open-Vocabulary Mobile Manipulation (OVMM). APRR effectively addresses the challenges associated with locating small, occluded ob-

jects in large, complex environments. The framework consists of two distinct stages: an initial exploration phase, responsible for efficient global search, and a subsequent last-mile navigation phase, dedicated to precise local positioning for downstream manipulation tasks.

3.1. Task Setup

This study focuses on OVMM ObjectNav, a sub-task within the 2023 Habitat OVMM Challenge [27]. Unlike conventional ObjectNav, where the agent navigates toward large, visually salient objects (e.g., beds, sofas), OVMM ObjectNav requires searching for small, often occluded objects (e.g., apples, cups) in complex and cluttered environments. At each timestep, the agent receives egocentric RGB-D images, localization information (e.g., GPS or pose estimates), and compass readings, and takes discrete actions: MOVE FORWARD, TURN LEFT, TURN RIGHT, and STOP. A key distinction of OVMM ObjectNav is that success is not solely determined by reaching the target object but also by stopping at an optimal position and orientation to facilitate subsequent manipulation. These additional constraints significantly increase the complexity of the task, requiring both efficient exploration and precise last-mile navigation.

3.2. Overall Pipeline

Our method, APRR, decouples the OVMM ObjectNav task into two distinct phases: exploration and last-mile navigation, as shown in Fig. 2.

In the exploration phase, the agent follows a rule-guided RL framework, leveraging RGB-D inputs, positional data, and semantic cues from YOLO-World [4] and SAM2 [16] to build a semantic map. A rule-based model proposes initial actions, refined by a learned policy combining state encoding and category-based action selection. The last-mile phase begins when the agent triggers the STOP action. For last-mile navigation, the agent refines its position using active target detection, dynamically adjusting based on detection feedback (confidence scores, IoU, semantic masks, and spatial cues). This two-phase approach ensures both efficient exploration and precise final positioning.

3.3. Semantic Detection and Mapping

OVMM Object Navigation requires open-set object detection, for which we use YOLO-World due to its efficiency and generalization. To bridge the domain gap between real-world training data and simulated objects in Habitat [13, 17], we fine-tune YOLO-World with frontier-based exploration [25], enabling systematic semantic data collection. Ground-truth labels from Habitat enhance detection performance and confidence reliability for active perception in last-mile navigation. Since YOLO-World provides only bounding boxes, we integrate SAM2 for instance segmentation, generating precise masks for spatial reasoning. The

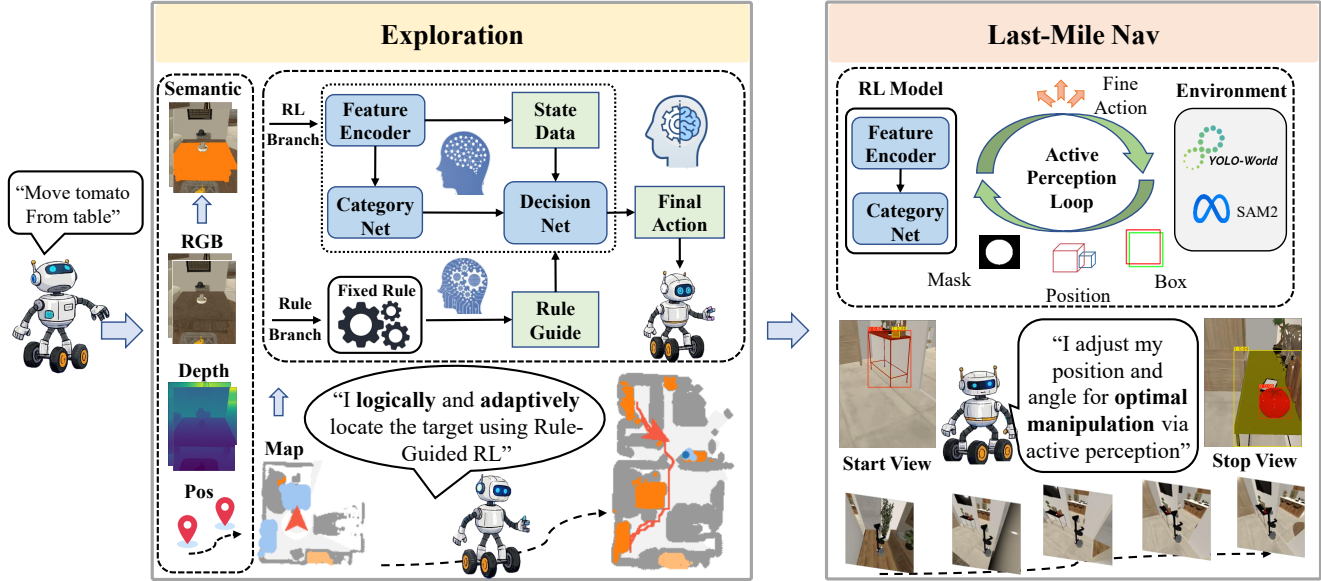


Figure 2. Overview of the APRR pipeline. The figure illustrates the complete process from input acquisition to action selection. The agent receives RGB-D observations and positional data, which are processed to extract semantic information and construct an environmental map, serving as inputs for both the Exploration and Last-Mile Navigation models. During the exploration phase, a rule-guided reinforcement learning model computes a final action tailored to the current scene, searching for the object logically and adaptively. The system transitions to the last-mile navigation phase when the final action is STOP. In this phase, the active target detection strategy employs the Active Perception Loop to guide the agent to the optimal position near the target, facilitating subsequent manipulation tasks. The sequence of images in the last-mile navigation phase visualizes the agent’s progressive adjustments, illustrating how it fine-tunes its position and orientation to optimize the final stopping viewpoint.

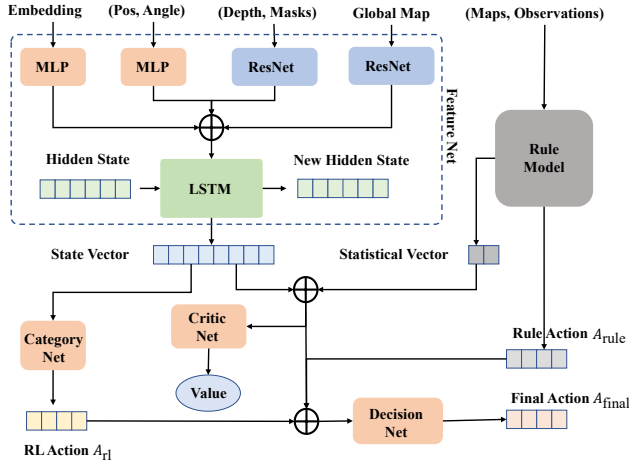


Figure 3. The Network of Exploration Model. The model consists of a Feature Encoder, Category Net, Critic Net, Rule Model, and Decision Net. The Rule Model provides heuristic action guidance, while the Decision Net dynamically balances RL and heuristic strategies for efficient and adaptive exploration.

agent constructs a 2D occupancy map by projecting position, depth, and semantic data, forming a Local Map for last-mile navigation and a Global Map for high-level exploration.

3.4. Exploration

The agent’s goal in exploration is to systematically cover unexplored areas to locate the target efficiently. In OVMM ObjectNav, small target objects are placed within larger “start receptacles”, requiring structured search strategies. Rule-based methods sequentially search receptacles but suffer from hyperparameter sensitivity, lack of adaptability, and inefficient navigation in dynamic environments. RL-based approaches improve adaptability but often struggle with systematic exploration due to limited memory and spatial awareness.

To address these issues, we introduce a rule-guided RL exploration model, which integrates the strengths of both rule-based heuristics and RL-based adaptability while mitigating their respective weaknesses. As shown in Fig. 3, the exploration model consists of five key components: the Feature Encoder, Category Net, Critic Net, Rule Model, and Decision Net. The Rule Model, derived from predefined OVMM ObjectNav heuristics, remains fixed and does not require training; instead, it provides heuristic-based action recommendations.

The model processes various sensory inputs, including the agent’s position, orientation, depth observations, semantic embeddings for both target objects and receptacles, and a down-sampled Global Map. These inputs are processed

through ResNet or MLP architectures before being combined with the recurrent state from an LSTM layer and the encoded representation of previous actions. The resulting feature representation is further refined using LSTM layers to produce an updated state representation.

Training consists of two stages: pretraining the RL policy and refining the Decision Net to fuse rule-based and RL-based actions. Both stages are optimized with DDPPO (Distributed Proximal Policy Optimization) [18, 23] algorithm. In the first stage, the Decision Net is excluded, and the Critic Net directly estimates the value function from state representation, while the Category Net samples action probabilities to guide movement. The reward function encourages efficient exploration, penalizes stagnation, and reinforces receptacle-based search behaviors:

$$R_{\text{exp-s1}} = \alpha_s + \Delta d_{\text{obj}} + \beta_{\text{rv}} \cdot \text{sign}(\Delta S_{\text{recep-visited}}) + R_{\text{stop-exp}}, \quad (1)$$

where $\alpha_s = -0.005$ is a penalty term for stagnation (*i.e.*, when the agent remains stationary without progress), and $\beta_{\text{rv}} = 0.2$ is the weight assigned to receptacle visited process. Δd_{obj} represents the reduction in shortest-path distance to the target object, computed using the ground-truth map in the Habitat simulator. $\Delta S_{\text{recep-visited}}$ denotes the increase in the explored area of receptacles within the agent’s perceptual field. The stopping reward, $R_{\text{stop-exp}}$, is applied only when the agent executes the stop action and is defined as:

$$R_{\text{stop-exp}} = \begin{cases} 10, & \text{if } \text{dis}(\text{agent}, \text{target}) < 3, \\ -5, & \text{otherwise.} \end{cases} \quad (2)$$

A successful stop within 3m of the target ends exploration; otherwise, failure occurs if the agent stops prematurely or runs out of time.

The second training stage focuses on refining the Decision Network. The state representation generated by the Feature Encoder, action probabilities A_{rl} produced by the Category Network, and categorical rule-based actions A_{rule} from the Rule Model are integrated into the Decision Network. Additionally, statistical features derived from the exploration map are incorporated, such as the exploration frequency F , as defined in Eq.3, which quantifies the cumulative number of visits to a given location. Another key feature is the path density D , which measures the concentration of visited locations within a specified neighborhood.

$$F(x, y) = \sum_{t=0}^T (\mathbb{I}(P_{\text{agent}}^t = (x, y))), \quad (3)$$

$$D(x, y) = \sum_{i=-\tau}^{\tau} \sum_{j=-\tau}^{\tau} (\text{sign}(F(x+i, y+j))), \quad (4)$$

where P_{agent}^t represents the position of the agent at time step t , τ represents the region size considered for computing the path density, and $\mathbb{I}(\cdot)$ represents the indicator function.

This Decision Net then computes an adaptive weighting factor λ that dynamically balances RL-based decision-making with rule-based heuristics. The final action probability vector A_{final} is obtained as Eq.5, from which a discrete action is selected via sampling.

$$A_{\text{final}} = \lambda A_{\text{rl}} + (1 - \lambda) A_{\text{rule}}. \quad (5)$$

In this phase, the reward function $R_{\text{exp-s2}}$ incorporates an additional term that encourages the agent to expand the explored area systematically, thus reinforcing structured exploration:

$$R_{\text{exp-s2}} = \alpha_s + \Delta d_{\text{obj}} + \beta_{\text{rv}} \cdot \text{sign}(\Delta S_{\text{recep-visited}}) + \beta_v \Delta S_{\text{visited}} + R_{\text{stop-exp}}, \quad (6)$$

where $\Delta S_{\text{visited}}$ denotes the overall expansion of the explored region, and $\beta_v = 0.005$ is the weight assigned to region visited process. The Critic Model further integrates statistical metrics from the exploration map to refine value function estimations.

Through this new rule-guided RL exploration framework, our model balances the strengths of heuristic-driven navigation with the flexibility and adaptability of RL-based decision-making. By enabling systematic and efficient exploration, our approach significantly enhances the performance and robustness of ObjectNav agents in OVMM environments, leading to more effective and scalable solutions for real-world exploration tasks.

3.5. Last-Mile Navigation with Active Detection

After exploration, the agent enters the Last-Mile Navigation stage, requiring precise adjustments for accurate alignment with the target. Unlike broad navigation, which only aims to reach the target’s vicinity, this phase fine-tunes viewpoint to optimize perception and interaction. Existing OVMM ObjectNav methods rely on rule-based heuristics for stopping but struggle with fine-grained positioning, leading to sub-optimal viewpoints. RL-based approaches often treat navigation as a single-stage process, neglecting the unique challenges of this stage.

To address these challenges, we introduce a new reinforcement learning-based Last-Mile Navigation model with active target detection. The key innovation of our approach is the integration of real-time target detection feedback into the navigation process. Unlike conventional methods that rely solely on spatial metrics (*e.g.*, Euclidean distance) to determine stopping conditions, our model continuously refines its actions based on dynamic visual feedback. We observe that stopping positions with higher detection confidence and improved alignment between predicted and

ground-truth object masks lead to more effective manipulation. Thus, our method leverages detection quality as an implicit stopping criterion, ensuring the agent halts at an optimal viewpoint.

The network architecture, adapted from the exploration model, is optimized for fine control in constrained spaces. The module takes as input the agent’s position, orientation, depth, semantic masks, target embeddings, detection confidence, and a local spatial map. Unlike the global map used in exploration, this local map enables precise adjustments. Inputs are processed through ResNet and MLP modules, concatenated with an LSTM hidden state and the previous action’s encoding. The state representation is fed to the Critic Net for value estimation and the Category Net for action prediction. Actions remain the same as in exploration (forward, left turn, right turn, stop), but turning angles are reduced from 30° to 15° for finer positioning.

The reward function for Last-Mile Navigation is designed to optimize both spatial proximity to the target and observation quality. It is formulated as follows:

$$R_{\text{last-mile}} = \alpha_s + \beta_{\text{obj}} \Delta d_{\text{obj}} + \beta_{\text{mi}} \text{sign}(\Delta R_{\text{mask-iou}}) + \beta_{\text{bi}} \text{sign}(\Delta R_{\text{box-iou}}) + \beta_{\text{c}} \text{sign}(\Delta R_{\text{conf}}) + R_{\text{stop-last}}, \quad (7)$$

where $\Delta R_{\text{mask-iou}}$, $\Delta R_{\text{box-iou}}$, and ΔR_{conf} represent changes in mask IoU, bounding box IoU, and detection confidence, respectively. The weighting coefficients β_{mi} , β_{bi} and β_{c} are all set to 0.2, while β_{obj} is set to 0.5 to reduce the influence of distance-based criteria. The stopping reward, $R_{\text{stop-last}}$ is only applied when the agent issues a stop action, defined as:

$$R_{\text{stop-last}} = \begin{cases} 10, & \text{if } C_{\text{stop}}, \\ -5, & \text{otherwise.} \end{cases} \quad (8)$$

The stopping condition C_{stop} is satisfied when both of the following criteria hold:

$$\begin{cases} \min_{v \in \mathcal{V}} \text{dis}(\text{agent}, v) < 0.1, \\ |\theta_{\text{agent}} - \theta_v| < 15^\circ, \end{cases} \quad (9)$$

where \mathcal{V} represents the set of predefined candidate viewpoints, $\text{dis}(\text{agent}, v)$ denotes the Euclidean distance between the agent’s stopping position and the closest viewpoint v , θ_{agent} and θ_v denote the agent’s orientation and the viewpoint’s orientation, respectively.

By incorporating real-time perception feedback, our model optimizes object localization and stopping behavior, improving observation quality and interaction success. Leveraging the reinforcement learning framework, the model utilizes detection-driven rewards during training to refine decision-making but operates independently of these rewards and without retraining during the test phase. This active perception refinement sets our method apart, significantly enhancing OVMM ObjectNav performance.

4. Experiments

In this section, we present comprehensive experiments of APRR, to the OVMM ObjectNav task within the Habitat simulator [15, 17] using the OVMM dataset [27]. We benchmark APRR against multiple baselines across 400 test episodes drawn from 12 unseen scenes. Additionally, we conduct ablation studies to analyze the contributions of our rule-guided reinforcement learning exploration model and active target detection-based last-mile navigation module. An episode is deemed successful if the agent identifies the target object and stops at a correct viewpoint within 800 simulation steps; otherwise, it is considered unsuccessful. Additionally, we perform real-world experiments to demonstrate the practical applicability of APRR. Further visualizations and details on experiments are provided in the supplementary materials.

4.1. Experimental Setup

Model Training. Both the exploration model and last-mile navigation module are trained using episodes collected from 38 training scenes. We leverage the DD-PPO framework [18, 23] for training, utilizing 8 NVIDIA 3090 GPUs with 12 simulation environments per node to ensure efficient interaction-driven learning. Each model undergoes training over 6000+ episodes.

Baselines. We evaluate our method against six baseline agents, each tested under two conditions: with a standard semantic detector and with ground-truth (GT) semantic masks (i.e., perfect object and receptacle masks). The baselines include:

- **FBE** [25]: A frontier-based exploration agent that constructs a 2D semantic map and navigates toward the target upon detection.
- **POLO** [22]: This method estimates the probability of target presence in unexplored areas using prior exploration data and selects the next region to investigate.
- **Uniteam** [11]: A baseline from the Home-Robot framework that dynamically adjusts perception thresholds and incorporates a target map decision mechanism.
- **Rule***: Our re-implementation of Uniteam with optimized collision handling and map updates.
- **Kuzhum** [9]: Uses MobileSAM segmentation and enhances Home-Robot RL with continuous rewards.
- **RL***: Our DD-PPO agent predicts actions from depth images and segmentation masks.

Evaluation Metrics. To assess navigation performance, we employ the following metrics:

- **SR (Success Rate)**: The fraction of successful episodes relative to the total test episodes.
- **DTG (Distance-to-Goal)**: The mean Euclidean distance from the agent’s final stopping position to the target object center, measuring proximity upon termination.

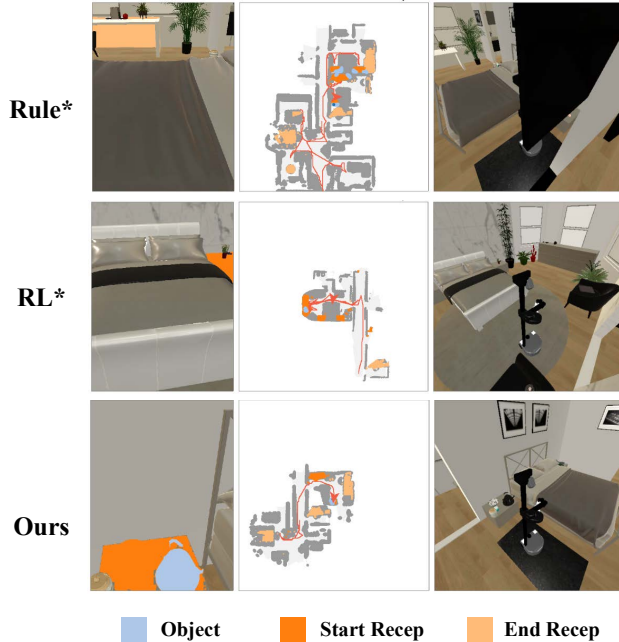


Figure 4. Comparison of our method with Rule* (rule-based) and RL* (RL-based) approaches during the exploration phase. The first column shows semantic detection results, the second shows the agent’s exploration map and trajectory, and the third shows third-person views. The rule-based agent exhibits significant hesitation but fails to adjust itself promptly, while the RL-based agent becomes trapped in repetitive search behavior. In contrast, our approach effectively integrates structured search with dynamic feedback, enabling the agent to reach the target successfully.

Method	Percep	SR(%) \uparrow	DTG \downarrow	PIC \uparrow
FBE [25]	Detic	10.9	/	/
PoLo [22]	Detic	20.3	/	/
Uniteam [11]	Detic*	49.2	/	/
KuzHum [9]	YS*+Detic	40.2	/	/
Rule*	YW*+S2	51.1	4.01	0.0155
RL*	YW*+S2	40.8	4.40	0.0223
APRR(Ours)	YW*+S2	64.5	3.36	0.0279
PoLo [22]	GT	56.7	/	/
Rule*	GT	54.4	3.04	0.0162
RL*	GT	48.5	3.94	0.0274
APRR(Ours)	GT	74.2	1.96	0.0447

Table 1. Comparison of Our Method to other baselines “Detic*” means “Detic finetune”, “YS*+Detic” means “Yolo-SAM finetune + Detic”, “YW*+S2” means “Yolo-world finetune + Sam2”.

- **PIC (Pick-Goal IoU Coverage):** IoU between the target’s ground-truth mask and the agent’s final grasping region, indicating optimal stopping positions.

4.2. Comparisons with Baselines

Tab. 1 compares our approach with baseline methods. Our method APRR, achieves consistent improvements over both

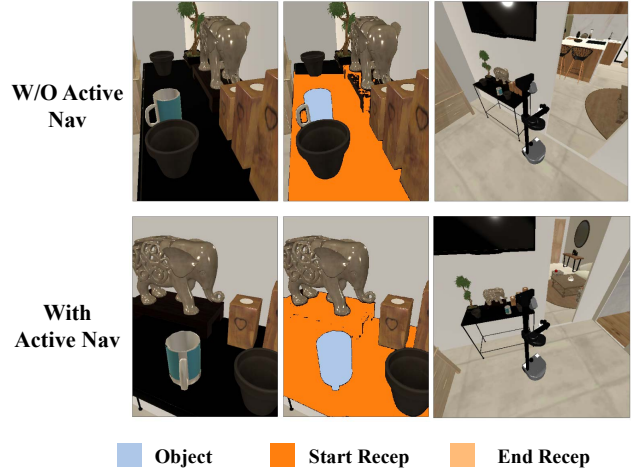


Figure 5. Comparison of last-mile navigation performance with and without the active navigation policy. The first column shows RGB observations, the second shows semantic detection, and the third shows third-person views. The left panel (“W/O Active Nav”) illustrates the performance when the active detection-based navigation strategy is not employed, while the right panel (“With Active Nav”) demonstrates our method. The results show that our active detection-driven navigation policy enables the agent to select a more suitable viewpoint, thereby enhancing the overall effectiveness of the navigation for subsequent manipulation tasks.

rule-based (FBE, PoLo, Uniteam, Rule*) and RL-based (Kuzhum, RL*) baselines. Notably, when using a standard semantic detector, our approach improves success rate by 13% over the state-of-the-art rule-based baseline (Rule*). With ground-truth semantic masks, this improvement increases to 17%.

Conventional methods such as FBE struggle with OVMM ObjectNav due to occlusion and the small size of targets. PoLo, while more recent and effective than naive strategies, remains limited, though it performs best among prior methods under oracle conditions. Uniteam and Kuzhum leverage perceptual fine-tuning but treat ObjectNav as a single-stage task, overlooking the differing requirements of exploration and precise localization.

In contrast, APRR explicitly separates these phases. Our rule-guided RL exploration framework combines the adaptability of RL with the structure of heuristic policies, guiding the agent efficiently to the target vicinity. This design is validated by consistently lower DTG values and qualitative results in Fig. 4.

Furthermore, our last-mile navigation module, driven by active perception, optimizes the agent’s stopping viewpoint. As illustrated in Fig. 5, this refinement significantly boosts PIC scores (Tab. 1), enabling more precise and manipulation-ready final positioning.

Method	Percep	SR(%) \uparrow	DTG \downarrow	PIC \uparrow
Rule*	YW*+S2	51.1	4.01	0.0155
Rule*+L-M Nav	YW*+S2	60.2	4.00	0.0258
RL*	YW*+S2	40.8	4.40	0.0223
RL*+L-M Nav	YW*+S2	43.6	4.37	0.0248
Ours(conf=0.5)	YW*+S2	64.5	3.36	0.0279
Ours(conf=0.3)	YW*+S2	63.3	3.84	0.0267
Rule*	GT	54.4	3.04	0.0162
Rule*+L-M Nav	GT	69.3	2.20	0.0421
RL*	GT	48.5	3.94	0.0274
RL*+L-M Nav	GT	56.9	3.08	0.0337
Ours($\alpha = 15$)	GT	74.2	1.96	0.0447
Ours($\alpha = 30$)	GT	68.3	2.39	0.0343

Table 2. Results of Ablation Experiment. “YW*+S2” means “Yolo-world finetune + Sam2”. “conf” is the threshold of Yolo-World detection, α is the angle of “TURN LEFT” and “TURN RIGHT” in the Last-Mile Stage.

4.3. Ablation Studies

We perform ablation studies to evaluate the contributions of each component in APRR, as shown in Tab. 2. The variants “Rule* + Last-Mile Nav” and “RL* + Last-Mile Nav” replace the original last-mile modules with our active target detection model. In both cases, this replacement significantly improves success rates and PIC scores, indicating more precise stopping behaviors. APRR further outperforms these variants, highlighting the effectiveness of our rule-guided RL exploration strategy in integrating heuristic and learned policies.

We also analyze two key hyperparameters in the last-mile phase: the detector confidence threshold and the rotation angle α . A threshold of 0.5 and a 15° rotation yield the best performance. Lower thresholds increase false positives, while larger rotations (e.g., 30°) reduce success, underscoring the importance of fine-grained control during final positioning.

Overall, the ablation results validate that both our rule-guided RL exploration model and the active detection-based last-mile navigation module significantly enhance exploration efficiency and stopping precision, addressing key challenges in OVMM ObjectNav navigation.

4.4. Real-World Experiment

To evaluate real-world performance, we deploy APRR and baseline methods on a custom mobile manipulation platform. The robot comprises a Ranger Mini v2.0 base with a Jetson Nano for onboard control, a Mid-360 LiDAR for localization, and an Intel RealSense D455 RGB-D camera for perception. Spatial-semantic mapping is built from RGB-D inputs using the YOLOv8-seg model [21].

For real-world testing, we conduct experiments in a large office environment, placing objects such as bottles, cups, and books on items like couches, tables, and chairs. The

Method	Path Length	SR(%)
PoLo* for Real-World	15m	5.0
Rule* for Real-World	15m	5.0
RL* for Real-World	15m	0.0
Ours for Real-World	15m	30.0

Table 3. Real-World Experiment Results. PoLo* refers to our reproduction of PoLo.

robot starts approximately 15 meters away from the target. A trial is successful if the agent stops within 1 meter of the target, facing it, with an object detection confidence above 0.9. As summarized in Tab.3, our method is evaluated across 20 trials, achieving a 30% success rate. Although this is lower than the performance observed in the simulation, it demonstrates the feasibility of our approach in real-world settings. In contrast, all baseline methods nearly failed, as they do not explicitly handle last-mile navigation, often stopping at suboptimal positions far from the target. The performance gap between simulation and real-world deployment highlights the challenges of sim-to-real transfer, emphasizing the need for improved adaptation techniques.

5. Conclusion

In this work, we propose a novel approach to OVMM ObjectNav by decoupling the task into two phases: exploration and last-mile navigation. In the exploration phase, we introduce a rule-guided RL model that combines the structured efficiency of rule-based strategies with the adaptability of reinforcement learning, enabling effective navigation in large, cluttered environments. In the last-mile navigation phase, we develop an RL model to achieve active target detection, allowing the agent to dynamically refine its position and orientation based on real-time feedback for subsequent manipulation. Experimental results show that our approach yields a 13% improvement in success rate over existing methods. Real-world experiments further validate the practicality of our approach. In summary, our method enhances both efficient exploration and precise navigation in OVMM ObjectNav. Future work will focus on improving sim-to-real transfer for real-world deployment.

Acknowledgement

This work is supported by National Key R&D Program of China under Contract 2022ZD0119802, and National Natural Science Foundation of China under Contract 62472141, Key Laboratory of Target Cognition and Application Technology under Contract 2023-CXPT-LC-005, the Youth Innovation Promotion Association CAS, and the GPU cluster built by MCC Lab of USTC and the Supercomputing Center of USTC.

References

- [1] Alexander Amini, Guy Rosman, Sertac Karaman, and Daniela Rus. Variational end-to-end navigation and localization. In *International Conference on Robotics and Automation*, pages 8958–8964, 2019. 1, 3
- [2] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural slam. In *International Conference on Learning Representations*, 2020.
- [3] Devendra Singh Chaplot, Dhiraj Prakashchand Gandhi, Abhinav Gupta, and Russ R Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. *Advances in Neural Information Processing Systems*, 33:4247–4258, 2020. 1, 3
- [4] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xingang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16901–16911, 2024. 2, 3
- [5] Wenhao Ding, Nathalie Majcherczyk, Mohit Deshpande, Xuewei Qi, Ding Zhao, Rajasimman Madhivanan, and Arnie Sen. Learning to view: Decision transformers for active object detection. In *IEEE International Conference on Robotics and Automation*, pages 7140–7146. IEEE, 2023. 3
- [6] Zhaoyuan Fang, Ayush Jain, Gabriel Sarch, Adam W Harley, and Katerina Fragkiadaki. Move to see better: Self-improving embodied object detection. *arXiv preprint arXiv:2012.00057*, 2020. 3
- [7] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, et al. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017. 1
- [8] Klemen Kotar and Roozbeh Mottaghi. Interactron: Embodied adaptive object detection. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 14860–14869, 2022. 3
- [9] Volodymyr Kuzma, Vladyslav Humennyi, and Ruslan Partsey. Homerobot open vocabulary mobile manipulation challenge 2023 participant report (team kuzhum). *arXiv preprint arXiv:2401.12048*, 2024. 2, 6, 7
- [10] Haokuan Luo, Albert Yue, Zhang-Wei Hong, and Pulkit Agrawal. Stubborn: A strong baseline for indoor object navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3287–3293, 2022. 3
- [11] Andrew Melnik, Michael Büttner, Leon Harz, Lyon Brown, Gora Chand Nandi, Arjun PS, Gaurav Kumar Yadav, Rahul Kala, and Robert Haschke. Uniteam: Open vocabulary mobile manipulation challenge. *arXiv e-prints*, pages arXiv–2312, 2023. 2, 6, 7
- [12] Claudia Pérez-D’Arpino, Can Liu, Patrick Goebel, Roberto Martín-Martín, and Silvio Savarese. Robot navigation in constrained pedestrian environments using reinforcement learning. In *IEEE International Conference on Robotics and Automation*, pages 1140–1146, 2021. 1, 3
- [13] Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dal-laire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander Clegg, Michal Hlavac, So Yeon Min, et al. Habitat 3.0: A co-habitat for humans, avatars, and robots. In *International Conference on Learning Representations*, 2023. 3
- [14] Adel Rahmoune, Pierre Vanderghenst, and Pascal Frossard. Mp3d: Highly scalable video coding scheme based on matching pursuit. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages iii–133. IEEE, 2004. 2
- [15] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. In *Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. 2, 6
- [16] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 2, 3
- [17] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *IEEE/CVF International Conference on Computer Vision*, pages 9339–9347, 2019. 3, 6
- [18] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 5, 6
- [19] Lingdong Shen, Chunlei Huo, Nuo Xu, Chaowei Han, and Zichen Wang. Learn how to see: Collaborative embodied learning for object detection and camera adjusting. In *AAAI Conference on Artificial Intelligence*, pages 4793–4801, 2024. 3
- [20] Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. *Advances in neural information processing systems*, 34:251–266, 2021. 2
- [21] Ultralytics. Yolov8, 2024. Accessed: 2024-08-26. 8
- [22] Jiaming Wang and Harold Soh. Probable object location (polo) score estimation for efficient object goal navigation. In *IEEE International Conference on Robotics and Automation*, pages 5221–5227. IEEE, 2024. 2, 6, 7
- [23] Erik Wijmans, Abhishek Kadian, Ari Morcos, Stefan Lee, Irfan Essa, Devi Parikh, Manolis Savva, and Dhruv Batra. Ddppo: Learning near-perfect pointgoal navigators from 2.5 billion frames. In *International Conference on Learning Representations*, 2019. 5, 6
- [24] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9068–9079, 2018. 1
- [25] Brian Yamauchi. Frontier-based exploration using multiple robots. In *International conference on Autonomous agents*, pages 47–53, 1998. 3, 6, 7

- [26] Sriram Yenamandra, Arun Ramachandran, Mukul Khanna, Karmesh Yadav, Devendra Singh Chaplot, Gunjan Chhablani, Alexander Clegg, Theophile Gervet, Vidhi Jain, Ruslan Partsey, et al. The homerobot open vocab mobile manipulation challenge. In *Conference on neural information processing systems: competition track*, 2023. [2](#)
- [27] Sriram Yenamandra, Arun Ramachandran, Karmesh Yadav, Austin S Wang, Mukul Khanna, Theophile Gervet, Tsung-Yen Yang, Vidhi Jain, Alexander Clegg, John M Turner, et al. Homerobot: Open-vocabulary mobile manipulation. In *Conference on Robot Learning*, pages 1975–2011. PMLR, 2023. [1](#), [2](#), [3](#), [6](#)
- [28] Albert J Zhai and Shenlong Wang. Peanut: Predicting and navigating to unseen targets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10926–10935, 2023. [3](#)
- [29] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In *IEEE international conference on robotics and automation*, pages 3357–3364. IEEE, 2017. [3](#)