2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

VLION: Vision-Language Guided Interactive Object Navigation with Mobile Manipulation

Renming Liu, Hao Ren, Lanxiang Zheng, Yiming Zeng, Ying Wu, Hui Cheng*

***Abstract*— Object navigation for mobile robots typically assumes that targets are visible and paths are unobstructed. However, real-world scenarios often involve occluded targets like objects hidden behind doors or inside containers. Such scenarios require interactive navigation and manipulation by mobile manipulators. To address this challenge, we propose VLION, a vision-language model-guided framework for interactive object navigation (ION) that enables robots to locate and access such targets efficiently. VLION constructs a probabilistic occupancy map and dynamically identifies frontiers for efficient exploration. It leverages vision-language models (VLMs) to perform joint semantic reasoning at both the scene and object levels, generating Scene-Target and Object-Target Value Maps from egocentric observations. These maps are adaptively fused based on spatial entropy to guide target selection and dynamically balance navigation and manipulation priorities for multi-step decision-making. A hybrid A* planner ensures safe and feasible navigation, while star-convex manipulation regions enable interaction with objects. Extensive experiments in iGibson simulations and real-world environments demonstrate the effectiveness of VLION in zero-shot transfer and on-board deployment, advancing the state of the art in ION.**

I. INTRODUCTION

Object navigation (ObjectNav) is a fundamental task in embodied AI, requiring robots to explore environments and navigate to objects of specified categories autonomously. While ObjectNav methods have made significant progress [1], [2], [3], it assumes that environments are unobstructed and target objects are directly visible and easily accessible [4], [5], [6]. These assumptions rarely hold in the real world, as objects hidden inside containers or behind doors often block the navigation paths.

Operating effectively in such scenarios requires robots not only to perceive and navigate but also to actively interact with the environment, thereby revealing and accessing hidden objects rather than merely conducting passive searches. This motivates **Interactive Object Navigation (ION)**, as illustrated in Fig. 1, where a robot must open containers or doors, navigate around occlusions, and infer object locations from visual contextual cues, which is similar to how a human searches for out-of-sight items at home.

Recent advances in large language models (LLMs) and vision-language models (VLMs) provide a promising foundation for tackling these challenges [7], [8], [9]. These models encode commonsense knowledge and enable robots to make informed decisions even in unfamiliar scenarios. Zero-shot methods built upon LLMs and VLMs have demonstrated impressive generalization and adaptability in navigation tasks [10], [11], [12].

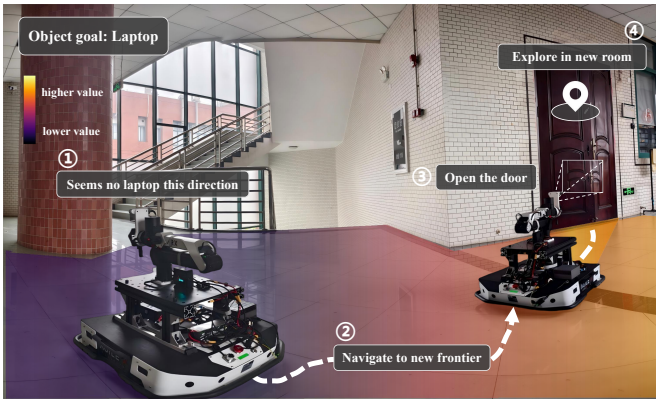

Fig. 1. **An Example of Interactive Object Navigation (ION).** The robot navigates to a target that is not directly visible. Here, the laptop is hidden behind a closed door. To succeed, the robot must open the door and continue navigating into the room, highlighting the core challenge of ION: actively interacting with the environment to reveal and access hidden targets.

Despite these advances, current methods predominantly focus on Object Navigation tasks, primarily emphasizing frontier exploration without adequately addressing interactive elements. In contrast, ION requires longer-horizon reasoning that considers both frontiers and objects as potential targets. Agents must decide not only where to go, but also what to interact with.

Additionally, many existing approaches rely on converting visual inputs into discrete textual representations, such as category labels [13] or semantic ground truth [14], prior to language-based reasoning. This intermediate conversion leads to the loss of detailed visual information and introduces computational overhead, negatively impacting the efficiency and effectiveness of navigation tasks.

To address the challenge of interactive object navigation in occluded and complex environments, we propose VLION, a novel unified Vision-Language-guided framework for ION. VLION constructs a probabilistic occupancy map and identifies frontier regions using only egocentric RGB-D observations. By leveraging the cross-modal alignment capabilities of vision-language models, VLION extracts both scene-level and object-level semantic cues. These are used in an adaptive strategy to construct a visual-language-grounded integrated value map, while preserving the richness of visual inputs by avoiding intermediate conversions into discrete textual representations. Our contributions are as follows:

This work was supported by the National Natural Science Foundation of China (U22A2095). *Corresponding to chengh9@mail.sysu.edu.cn

Renming Liu, Hao Ren, Lanxiang Zheng, Yiming Zeng, Ying Wu, and Hui Cheng are with the School of Computer Science and Engineering, Sun Yat-sen University.

1) **General framework for ION:** The proposed VLION unifies semantic reasoning and geometric planning in a general framework, enabling efficient long-horizon navigation for interactive object navigation.
2) **Visual-language-guided integrated value map:** We introduce a novel value mapping method that adaptively fuses scene-level and object-level semantic cues extracted from vision-language models, enhancing semantic understanding during navigation.
3) **Comprehensive evaluation of VLION:** Extensive experiments in simulation and real-world scenarios demonstrate VLION's superiority over baselines and its zero-shot transfer for real-world deployment.

II. RELATED WORK

A. Object Navigation

Object Goal Navigation (ObjectNav) aims to guide robots to locate specific object categories within previously unseen environments. Early approaches are predominantly learning-based, including reinforcement learning (RL) [15] and imitation learning [16], [17], where agents learn action policies directly from egocentric observations. While effective within training distributions, these end-to-end models often suffer from limited generalization and high sample complexity. To address these challenges, modular methods [1], [18] decompose the navigation pipeline into interpretable submodules, typically involving semantic mapping, object detection, and classical planning. By projecting perceptual inputs into top-down spatial representations, these approaches enable more structured decision-making. However, their reliance on predefined object categories and handcrafted components limits adaptability and scalability in open-world settings.

Recently, zero-shot ObjectNav methods [19], [2], [13], [20] have emerged by leveraging the open-vocabulary and cross-modal reasoning capabilities of foundation models [7], [8], [21]. These approaches enable semantic navigation without task-specific training, allowing robots to generalize to novel categories and real-world scenarios. Building upon these insights, our framework tightly integrates vision-language reasoning, spatial navigation, and interaction capabilities to achieve robust zero-shot Interactive Object Navigation (ION).

B. Embodied Mobile Manipulation

Mobile manipulation robots combine locomotion and manipulation to address complex tasks in cluttered indoor or multi-room environments. These scenarios demand integrated scene understanding, long-horizon planning, and adaptability to dynamic, uncertain conditions.

In structured environments with known topology, methods such as BUMBLE [22] and SayPlan [23] rely on vision-language models or pre-built scene graphs with LLMs for planning but typically assume static, known environments, limiting their applicability to unknown or dynamic scenarios.

For pick-and-place tasks in household environments, HomeRobot [24] introduces the Open-Vocabulary Mobile Manipulation (OVMM) benchmark. OK-Robot [25] tackles OVMM by integrating VLMs with navigation and grasping, enabling zero-shot execution in real-world. DovSG [26] employs LLMs to update local 3D scene graphs, supporting dynamic perception and task decomposition. Qiu et al. [27] combine SLAM, VLMs, and LLMs to build semantic BEV maps for area-aware reasoning. COME-Robot [28] further improves robustness by leveraging GPT-4V for closed-loop failure detection and recovery in OVMM tasks. MoMa-LLM [14] considers scenarios where objects cannot be freely accessed. It explores unknown environments use ground-truth semantic segmentation to build scene graphs, and using LLMs to classify rooms and select skills.

However, these works assume partial environmental priors, such as spatial topology or pre-existing object detection and semantic segmentation, to encode spatial and semantic information for LLM-based reasoning. Moreover, these LLM-based approaches often suffer from a key bottleneck: visual information must be converted into text by object detectors, losing fine-grained details. LLM inference is also computationally expensive, limiting decision frequency and often requiring remote servers.

In contrast, our proposed VLION framework directly leverages pre-trained vision-language models without converting visual input to text. By utilizing vision-language alignment, VLION constructs a unified environment representation that combines the Scene-Target Value Map and the Object-Target Value Map, meeting the semantic understanding requirements for mobile manipulation. Additionally, VLION enables onboard deployment, fast inference, and real-time decision-making for mobile manipulation.

III. PROBLEM FORMULATION

We address the task of Interactive Object Navigation (ION), where a robot must navigate to a specified target object in an unknown environment, while actively interacting with the environment to reveal or access occluded objects. The robot uses an egocentric RGB-D camera and odometry to explore, plan, and navigate effectively toward the target, minimizing both path length and interaction times. The action space consists of the following: MOVE FORWARD (0.075 m), TURN LEFT (35°), TURN RIGHT (35°), and OPEN OBJECT.

IV. METHOD

The overview of VLION is shown in Fig. 2. VLION consists of two key modules: Multi-Layer Value Mapping and Mobile Manipulation. The former constructs a probabilistic map $\mathcal{O}$ and frontiers $\mathcal{F}$ from egocentric observations, and employs BLIP-2 to generate semantic value maps (V_{st}, V_{ot}). These are fused via an adaptive strategy based on spatial entropy D into an integrated map V_{int} for selecting the most valuable target. The latter takes the selected frontier or closed object and performs goal-directed interaction using Hybrid A* navigation and star-convex region planning.

A. Multi-Layer Value Mapping

1) Frontier Detection and Occupancy Mapping: We represent the geometric structure of the environment using a probabilistic occupancy grid map $\mathcal{O}$. Each grid cell is

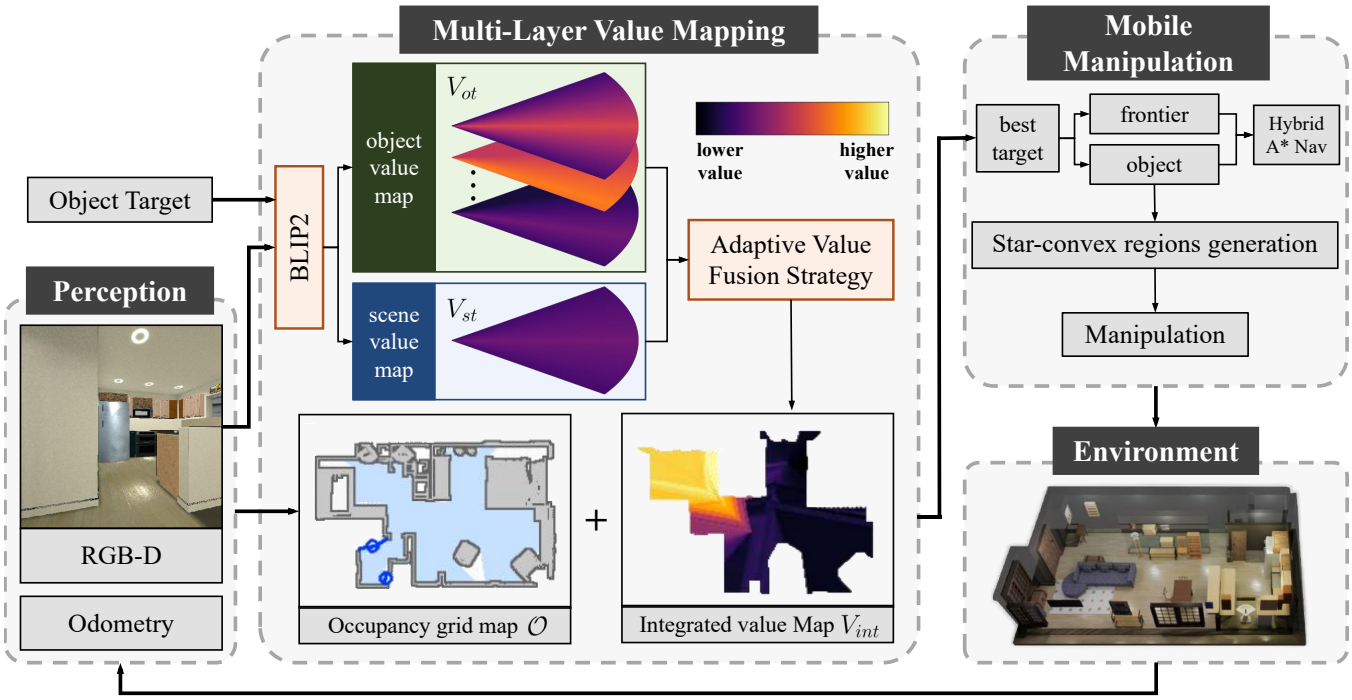

Fig. 2. **System Architecture of VLION.** Given an object target, egocentric RGB-D observations and odometry, VLION first builds a probabilistic map $\mathcal{O}$ and detects frontiers $\mathcal{F}$. Next, BLIP-2 generates Scene-Target and Object-Target value maps (V_{st}, V_{ot}), which are adaptively fused into an Integrated Value Map V_{int} for target selection. The frontier or closed object with the highest value is then passed to the mobile manipulation module, where Hybrid A* is employed to search a feasible navigation path, and star-convex regions are constructed to provide safe and effective interaction region around closed objects.

classified as *free*, *occupied*, or *unknown*, based on the observed data. The resolution of the map is configurable, allowing for flexible representation of environments with varying structural complexity. To build the map, depth images are converted to point clouds and, after being filtered to remove outliers, are probabilistically integrated into the occupancy map. The filtered points are then transformed into the global coordinate frame using odometry, and the map is updated in dynamic scenes through raycasting.

To enable efficient exploration, the environment is divided into known and unknown regions. The known region consists of confirmed free space and occupied obstacles. We denote the set of frontier points, which form the boundary between explored (known) and unexplored (unknown) areas, as $\mathcal{F}$. During navigation, the frontier set $\mathcal{F}$ is dynamically detected and continuously updated as the robot gathers sensor data, until the entire environment is thoroughly explored.

2) Semantic Value Map Generation: To infer target-related semantic value from the current egocentric RGB image, we employ the cross-modal reasoning capabilities of vision-language models such as BLIP-2 [21]. These models utilize a contrastive learning framework to align visual and textual embeddings, ensuring that semantically corresponding image-text pairs exhibit high similarity in the shared multimodal feature space, while non-corresponding pairs are effectively separated. Leveraging this alignment, we compute the cosine similarity score between the encoded image and the given prompt, denoted as s, which quantifies the relevance of the visual observation to the specified task:

$$s = \text{BLIP2}(\text{RGB, prompt}) \tag{1}$$

For interactive object navigation tasks, the robot must efficiently approach occluded target objects while minimizing both travel distance and interaction frequency. To facilitate this, we leverage BLIP-2 to compute two types of semantic similarity scores: the scene-target score s_{st} and the object-target score s_{ot}, which enable semantic-level reasoning crucial for interactive mobile manipulation.

For the scene-target score s_{st}, we employ the textual prompt "`There might be a <target object> ahead.`" to assess the semantic relevance between the current scene and the specified target object.

Given the camera's horizontal field of view θ_{fov}, s_{st} is uniformly projected onto grid cells within a sector-shaped region in the robot coordinate frame. The resulting pixel-wise scene-target score at location (i, j) is denoted as $s_{st}(i, j)$.

For the object-target score s_{ot}, we perform class-agnostic object segmentation to extract bounding boxes and the corresponding image patches. Each patch is then paired with the prompt "`It seems to appear together with <target object>`." to compute s_{ot}.

This score is spatially propagated over a directional sector centered at the object's bounding box center, oriented along its azimuth angle ϕ. The sector's angular width α is estimated based on the bounding box's horizontal span.

Beyond this sector, but still within the camera's FOV, an angular exponential decay is applied to attenuate the influence:

$$w_{\text{decay}}(\theta) = \exp\big(-\lambda \cdot |\theta - \phi|\big), \tag{2}$$

where θ is the angle between the pixel ray and the optical axis. The pixel-wise object-target score $s_{ot}(i,j)$ at location (i,j) is then defined as:

$$s_{ot}(i,j) = s_{ot} \cdot w_{\text{decay}}(\theta). \tag{3}$$

The confidence $c(i,j)$ within the robot's field of view depends on the angular offset of each pixel relative to the camera's optical axis. Specifically, pixels closer to the optical center are considered more reliable. The confidence is computed as:

$$c(i,j) = \cos^2\left(\theta/(\theta_{\text{fov}}/2) \cdot \pi/2\right). \tag{4}$$

Accordingly, the final value: confidence-weighted score at (i,j) in the robot's frame given by:

$$v(i,j) = c(i,j) \cdot s(i,j). \tag{5}$$

When the robot's position is updated, overlapping regions between the current and previously FOV must be updated. For each pixel (i,j) value $v^{new}(i,j)$ and the confidence $c^{new}(i,j)$ are updated as:

$$v^{new}(i,j) = \frac{v^{cur}(i,j) + v^{pre}(i,j)}{c^{cur}(i,j) + c^{pre}(i,j)}, \tag{6}$$

$$c^{new}(i,j) = \frac{(c^{cur}(i,j))^2 + (c^{pre}(i,j))^2}{c^{cur}(i,j) + c^{pre}(i,j)}. \tag{7}$$

Finally, leveraging the robot's odometry, the scene-target v_{st} and object-target values v_{ot} are projected onto a 2D plane to construct the global **Scene-Target Value Map** V_{st} and **Object-Target Value Map** V_{ot} in the world coordinate frame. As shown in Fig. 3, Object-Target Value Map V_{ot} offers fine-grained semantic cues that benefit ION task.

3) Adaptive Value Fusion Strategy: With both scene-level and object-level semantic representations, it becomes necessary to adaptively fuse the two value maps to guide navigation and manipulation more effectively.

When local scene values around the robot exhibit high spatial variance, this suggests the presence of a clear target direction, and the robot can primarily rely on scene-level guidance. In contrast, when scene values are uniformly distributed, it implies a lack of clear semantic cues, and object-level values should be weighted more heavily.

To this end, we propose an adaptive value fusion strategy that dynamically adjusts the fusion weights between the Scene-Target Value Map V_{st} and the Object-Target Value Map V_{ot}.

We first compute the Spatial Entropy D of scene values in the local neighborhood around the robot to quantify semantic differentiation:

$$D = \frac{-\sum_{(i,j)\in\Omega} V_{st}(i,j) \cdot \log V_{st}(i,j)}{\log N_\Omega}. \tag{8}$$

The scene value weight is then computed using a linear interpolation strategy with a minimum weight threshold:

Fig. 3. **Illustration of the advantages for Object-target Value Map V_{ot}. Left:** When the scene-target map V_{st} gives ambiguous guidance at junctions, V_{ot} provides cues to guide navigation toward relevant areas. **Right:** Faced with multiple closed objects, V_{ot} prioritizes the one most related to the target, reducing redundant interactions.

$$w_{st} = \delta + (1-\delta) \cdot D, \tag{9}$$

where, N_Ω denotes the total number of valid grids within a $3\,\text{m} \times 3\,\text{m}$ area centered around the robot, and $V_{st}(i,j)$ represents the scene-target value at grid (i,j). $\delta \in [0,1]$ denotes the lower bound for the scene value weight, ensuring that scene-level information always contributes to the final decision. We set $\delta = 0.5$ to allow balanced contributions in ambiguous cases. When D approaches 0 (indicating low spatial variance in scene values and lack of distinct semantic cues), w_{st} remains close to δ, assigning equal importance to both scene and object values. Conversely, when D approaches 1 (indicating high spatial differentiation and a clear semantic direction), w_{st} approaches 1, allowing the decision to rely primarily on scene-level guidance. The object value weight is computed accordingly as $w_{ot} = 1 - w_{st}$, enabling adaptive fusion based on local semantic characteristics. Finally, the **Integrated Value Map** V_{int} is computed as:

$$V_{int} = w_{st} \cdot V_{st} + w_{ot} \cdot \left(\frac{1}{K}\sum_{k=1}^{K} V_{ot}^{(k)}\right), \tag{10}$$

where K denotes the number of detected objects in the scene, with each $V_{ot}^{(k)}$ representing the Object-Target Value Map corresponding to the k-th object.

Based on V_{int}, the agent selects the frontier or closed object with the highest value as the target, which is then passed to the mobile manipulation module for execution.

B. Mobile Manipulation

Mobile manipulation involves two key components: navigation and manipulation. The first step is to safely navigate to the target location. Traditional methods often employ A* to compute the shortest path and follow it directly, but such paths may pass dangerously close to obstacles, increasing the risk of collision. To improve safety, we first inflate the occupancy grid according to the robot's radius, creating a buffer zone around obstacles. This inflated map is then converted into

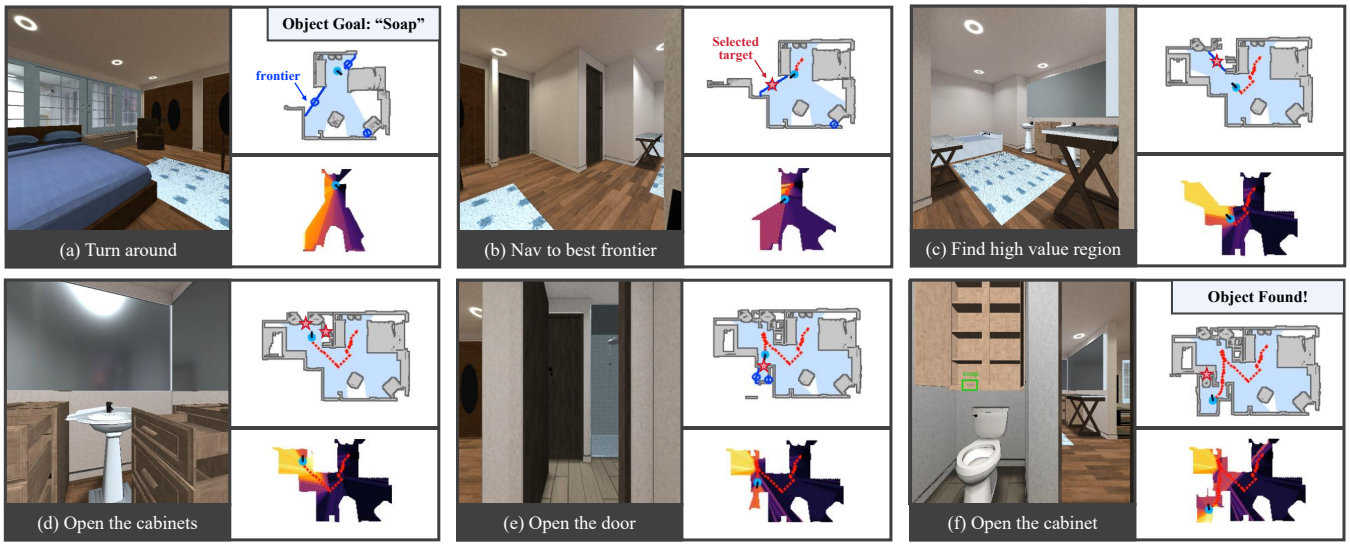

Fig. 4. **An Illustrative Simulation Example of ION.** (a) The robot begins by rotating to initialize its understanding of the environment. (b) It navigates toward the most valuable frontier. (c) Using the Integrated Value Map, it identifies high-value regions near the sink. (d) The robot interacts with two cabinets but fails to find the target. (e) It continues exploration, opens the door, and enters the bathroom. (f) Finally, it opens another cabinet and successfully discovers the target object (soap).

a cost map, where cells closer to obstacles are assigned higher traversal costs. We adopt the Hybrid A* algorithm for path planning, which extends A* by considering the robot's kinematic constraints. It plans in (x, y, θ) state space, where θ denotes the robot's heading, ensuring that the resulting path is both smooth and kinematically feasible.

After successfully navigating to the target location, to ensure feasible and safe interactions, we define the manipulation region as a star-convex area within the free space surrounding the target object. Owing to the star-convex property: any boundary point can be connected to the center by a line entirely contained within the region. The robot is guaranteed a reachable path to interact with the object when positioned inside the region.

To construct the manipulation region efficiently, we first select a point located at a distance of $r/2$ in front of the interaction target as the star center P_{manip}, where r denotes the robot's operational radius. A set of enhancement points is then uniformly sampled within a sphere of radius $r/2$ centered at P_{manip}, and merged with the local point cloud P_{pcl} to form a unified set P.

Next, apply spherical inversion to all points in P, transforming them into $\hat{P}$, and compute the convex hull of $\hat{P}$ using the Quickhull algorithm [29]. The resulting convex surface is then mapped back to the original space via inverse spherical inversion, using geometric partitioning to extract the kernel, defines the final manipulation region [30].

The overall ION process is illustrated in Fig. 4. In the figure, blue circles denote detected frontiers, the red dotted curve represents the robot's executed trajectory, and the star indicates the selected exploration target. The robot first performs an initialization by rotating in place, then proceeds to explore high-value frontiers in the vicinity of the sink, during which it interacts with two cabinets. Subsequently, it opens the door, enters the bathroom, and ultimately finds the target inside a cabinet.

V. EXPERIMENT

In this section, we aim to systematically investigate the effectiveness and generalization capabilities of VLION via the following questions:

1) How does VLION perform on ION tasks compared to existing LLM-based and heuristic baseline methods?
2) What are the individual and combined impacts of the Scene Value Map, Object Value Map, and the adaptive fusion strategy on VLION's performance?
3) Can VLION be effectively deployed in real-world scenarios, illustrating zero-shot generalization capabilities?

A. Simulation

We evaluate our approach in the iGibson [31], which provides interactive, realistic apartment environments with a diverse set of everyday objects. Logical scene configuration is supported via BDDL, allowing automatic placement of objects inside or on top of containers. To evaluate ION task, target objects are placed in different rooms and concealed behind doors or inside containers, requiring the robot to actively interact with the environment for discovery. Experiments are conducted across seven scenes, each repeated for 10 episodes.

The simulated robot is equipped with a differential-drive base and an RGB-D camera (512×512 resolution, 90° FOV) with an effective range of 0-5 meters. Obstacles with heights between 0.1 and 0.8 meters are considered for navigation. This ensures both floors and carpets are correctly recognized as traversable space. The robot has access to ground-truth odometry and constructs a probabilistic occupancy grid map from depth observations. For manipulation, a magic open operation is assumed, directly setting the `object_states.Open` to `True` to complete the open action.

B. Metrics

We evaluate performance using the following four metrics:

- **Success Rate (SR):** The proportion of episodes in which the robot successfully reaches and interacts with the target object.
- **Path Length (PL):** The average distance traveled by the robot during each episode, measured along its executed trajectory.
- **Success weighted by Path Length (SPL)** [32]: Evaluates navigation efficiency by averaging the ratio of the shortest path length to the actual path length for each episode, with failed episodes counted as zero.

$$\text{SPL} = \frac{1}{N}\sum_{i=1}^{N} S_i \cdot \frac{l_i}{\max(p_i, l_i)} \tag{11}$$

  where N is the number of episodes, $S_i \in \{0, 1\}$ indicates success, l_i is the shortest path length, and p_i is the actual path length.
- **Object Interactions (Obj-Interaction):** The average number of object interactions during each episode.

C. Baselines

We compare VLION with following baseline methods:

- **Random:** Randomly select a target from the available list, including detected frontiers and closed objects.
- **Greedy:** Employ the Hybrid A* algorithm to compute the shortest path to all potential targets and select the one closest to the robot's current position.
- **ESC-Interactive [13]:** Employ predefined prompts and DeBERTa v3 [33] for reasoning, estimating co-occurrence scores among the target, detected objects, and room types. The frontier with the highest score is selected for navigation. We extend it by additionally considering closed objects as candidate targets.
- **MoMa-LLM [14]:** Leverage semantic cameras to obtain ground-truth labels for constructing a scene graph, which is transformed into a structured textual representation for LLM-based room classification and target selection. High-level reasoning is performed using GPT-4.1, while room classification is handled by GPT-4.1-mini.

D. Experimental Results and Analyses

Table I reports the average performance of VLION and baseline methods across different scenarios. The best results are highlighted in **bold**, and the second-best are underlined.

For heuristic, non-semantic methods, Greedy prioritizes local optimality, selecting the shortest path to the target at each step, but lacks a global planning perspective. In complex layouts requiring multi-room traversal or interaction with multiple objects, it often follows suboptimal paths, resulting in the lowest SR and the highest number of Obj-Interaction. In contrast, Random avoids local traps and achieves slightly higher SR than Greedy; however, its undirected exploration leads to significantly longer PL and very low SPL (21.44%), reflecting inefficient navigation.

TABLE I
SIMULATION RESULTS OF INTERACTIVE OBJECT NAVIGATION IN IGIBSON ACROSS SEVEN SCENARIOS.

Method	SR↑	PL↓	SPL↑	Obj-Interaction↓
Greedy	84.1	19.90	38.69	8.157
Random	91.7	37.23	21.44	7.014
ESC-Interactive [13]	96.3	15.66	61.40	4.486
MoMa-LLM [14]	96.8	16.85	65.46	3.514
VLION (Ours)	**98.1**	**14.14**	**69.94**	**3.286**

Among semantic-guidance methods, ESC-Interactive [13] employs co-occurrence scores via handcrafted prompts, improves over non-semantic baselines, its reliance on fixed templates and pairwise scoring across limited candidates restricts generalization to diverse or unseen scenarios. MoMa-LLM [14] achieves slightly higher SR (96.8%) and SPL (65.46%) with fewer Obj-Interaction, but its rigid object-room bindings often cause misclassification and error propagation in open or ambiguous environments. In semantically sparse regions, such as corridors or empty rooms, unstable room labels can lead to redundant exploration. In contrast, VLION leverages both scene-level and object-level semantic cues through dense Scene-Target value maps, enabling globally optimized decision-making, guiding robots out of semantically sparse regions, and reducing unnecessary exploration.

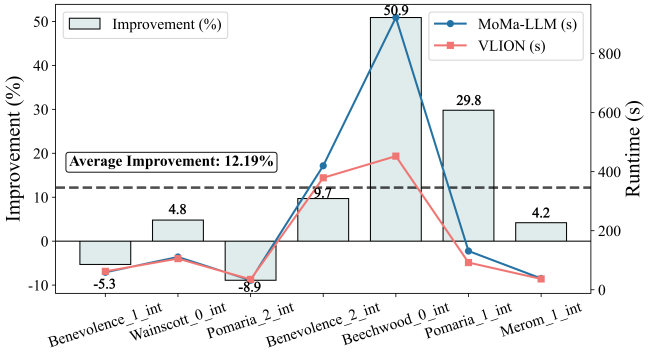

Fig. 5. **Runtime Comparison Across Seven Scenarios.** VLION achieves lower runtime than MoMa-LLM [14], with an average improvement of 12.19%, especially in large-scale scenarios.

Fig. 5 presents the runtime comparison between the two methods across seven scenarios. The left vertical axis indicates the relative improvement, while the right vertical axis shows the absolute runtime. VLION consistently achieves shorter runtimes, with an average improvement of 12.19%. The advantage is particularly evident in large-scale environments such as *Beechwood_0_int*, where runtime is reduced by more than 50%. This efficiency arises from directly aligning egocentric visual observations with target semantics, which enables high-frequency, vision-grounded planning and avoids the inefficiency of two-stage textual conversion. In contrast, the LLM-based episodic planner is limited by its reliance on inference-driven decision cycles, leading to lower responsiveness and delayed adaptation to newly emerging promising targets. Overall, the results demonstrate VLION's stronger scalability and adaptability in complex environments.

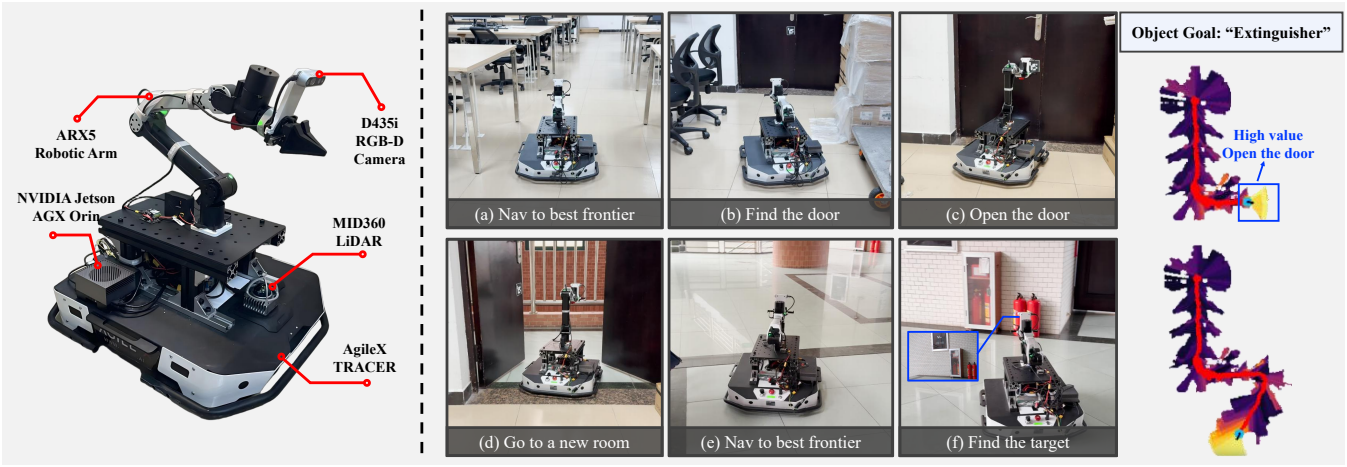

Fig. 6. **Qualitative Results of Real-World Experiments.** The mobile manipulator is deployed in an indoor office environment, tasked with autonomously navigating to a fire extinguisher located behind a closed door. VLION can infer the best frontier and executes the door-opening action, and successfully reaches the target object. The figure illustrates these key steps and the dynamic changes in V_{int}.

TABLE II
EVALUATING THE CONTRIBUTIONS OF DIFFERENT VALUE MAPS AND THE ADAPTIVE VALUE FUSION STRATEGY

Method	SR↑	PL↓	SPL↑	Obj-Interaction↓
VLION w/o s-t Map	94.2	18.60	63.13	4.509
VLION w/o o-t Map	96.9	15.39	65.68	4.688
VLION w/o Strategy	97.4	14.58	67.57	3.611
VLION (Ours)	**98.1**	**14.14**	**69.94**	**3.286**

E. Ablation Study

To assess the contribution of each core component in VLION, we perform ablation studies by selectively disabling the Scene-Target Value Map (s-t Map), Object-Target Value Map (o-t Map), and the Adaptive Fusion Strategy.

As shown in Table II and further qualitatively illustrated in Fig. 7, removing the s-t Map results in the most substantial performance degradation, with SR dropping from 98.1% to 94.2% and PL rising sharply to 18.60 m. This clearly highlights the critical importance of scene-level semantic reasoning, since the s-t Map provides essential global contextual priors that consistently guide efficient and reliable navigation.

Interestingly, retaining only the o-t Map still supports reasonable navigation, suggesting that object-level semantics can partially compensate for the absence of global guidance. However, Obj-Interaction increases from 3.286 to 4.688, reflecting the o-t Map's critical role in enabling precise interaction decisions by leveraging localized semantic cues.

Replacing the Adaptive Fusion Strategy with a fixed weighting scheme ($w_{st} = w_{ot} = 0.5$), still maintains competitive performance, showing that both semantic layers are individually robust. Nonetheless, the full model achieves consistently better results across all metrics, with a 0.7% SR improvement, 2.37% SPL improvement, and 0.325 unnecessary interactions. This confirms that dynamic fusion guided by semantic entropy not only enhances decision-making but also allows the agent to flexibly shift focus between exploration and manipulation based on contextual uncertainty.

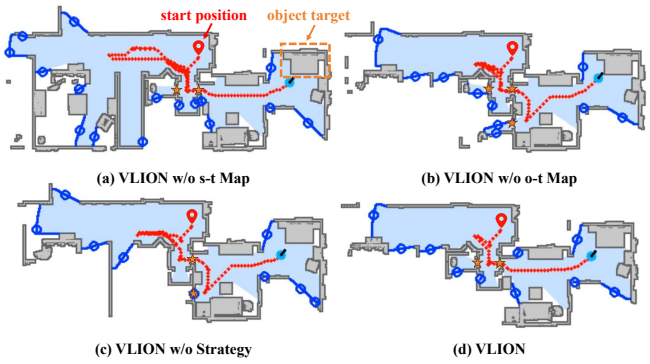

Fig. 7. **Qualitative Results of Ablation Study.** Blue circles indicate frontier centroids, the red dotted curve shows the robot's executed trajectory, pentagrams mark interactive objects, and the red pin denotes the start position.

F. Real-world Deployment

We deploy VLION on a real-world mobile manipulator platform composed of an AgileX TRACER differential-drive base, an ARX5 6-DOF robotic arm, an Intel RealSense D435i RGB-D camera, and a MID360 LiDAR. The hand-eye RGB-D camera provides egocentric visual observations within a 0-3.5 m depth range, while the LiDAR supplies odometry through the Fastlio [34]. All modules run fully onboard an NVIDIA Jetson AGX Orin. To enable interaction, ArUco [35] markers support accurate object detection, and the robotic arm is controlled with MoveIt for motion planning.

Fig. 6 presents a real-world deployment example in an indoor office, where the robot is tasked with autonomously navigating to a fire extinguisher located behind a closed door. Starting from its initial position, VLION explores informative frontiers and identifies the closed door as a high-value region requiring interaction. It then infers the need to open the door and seamlessly performs the action. After entering the new room, the robot continues navigation until it successfully reaches the target object. This demonstrates that VLION can actively seek occluded objects and validate its effectiveness for zero-shot transfer and practical real-world applications.

VI. CONCLUSION

We presented VLION, a unified vision-language-guided framework for Interactive Object Navigation (ION) in unknown and dynamic environments. By integrating geometric frontiers with semantic values from egocentric RGB-D observations, the system achieves fine-grained reasoning at both the scene and object levels, which demonstrates that VLION enables robust and efficient decision-making in obstructed environments. Through extensive evaluation in both simulation and real-world settings, the framework demonstrates improved navigation efficiency and strong zero-shot generalization across unseen environments and tasks.

Our work demonstrates mobile manipulation's interactive capabilities can enhance complex navigation tasks. Future research may exploit the flexibility of robotic arms for active exploration and target search in constrained environments.

REFERENCES

[1] D. S. Chaplot, D. P. Gandhi, A. Gupta, and R. R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4247–4258, 2020.

[2] N. Yokoyama, S. Ha, D. Batra, J. Wang, and B. Bucher, "Vlfm: Vision-language frontier maps for zero-shot semantic navigation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 42–48.

[3] H. Yin, X. Xu, L. Zhao, Z. Wang, J. Zhou, and J. Lu, "Unigoal: Towards universal zero-shot goal-oriented navigation," *arXiv preprint arXiv:2503.10630*, 2025.

[4] F. Xia, A. R. Zamir, Z. He, A. Sax, J. Malik, and S. Savarese, "Gibson env: Real-world perception for embodied agents," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9068–9079.

[5] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," *arXiv preprint arXiv:1709.06158*, 2017.

[6] S. K. Ramakrishnan, A. Gokaslan, E. Wijmans *et al.*, "Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai," *arXiv preprint arXiv:2109.08238*, 2021.

[7] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[8] A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan *et al.*, "Deepseek-v3 technical report," *arXiv preprint arXiv:2412.19437*, 2024.

[9] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023.

[10] S. Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, and S. Song, "Cows on pasture: Baselines and benchmarks for language-driven zero-shot object navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 23 171–23 181.

[11] M. Zhang, Y. Du, C. Wu, J. Zhou, Z. Qi, J. Ma, and B. Zhou, "Apexnav: An adaptive exploration strategy for zero-shot object navigation with target-centric semantic fusion," *arXiv preprint arXiv:2504.14478*, 2025.

[12] L. Zheng, R. Mei, M. Wei, R. Liu, H. Ren, G. Pan, and H. Cheng, "Get: Goal-directed exploration and targeting for large-scale unknown environments," *arXiv preprint arXiv:2505.20828*, 2025.

[13] K. Zhou, K. Zheng, C. Pryor, Y. Shen, H. Jin, L. Getoor, and X. E. Wang, "Esc: Exploration with soft commonsense constraints for zero-shot object navigation," in *International Conference on Machine Learning*. PMLR, 2023, pp. 42 829–42 842.

[14] D. Honerkamp, M. Büchner, F. Despinoy, T. Welschehold, and A. Valada, "Language-grounded dynamic scene graphs for interactive object search with mobile manipulation," *IEEE Robotics and Automation Letters*, 2024.

[15] N. Gireesh, D. S. Kiran, S. Banerjee, M. Sridharan, B. Bhowmick, and M. Krishna, "Object goal navigation using data regularized q-learning," in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2022, pp. 1092–1097.

[16] R. Ramrakhya, D. Batra, E. Wijmans, and A. Das, "Pirlnav: Pretraining with imitation and rl finetuning for objectnav," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 896–17 906.

[17] H. Ren, Y. Zeng, Z. Bi, Z. Wan, J. Huang, and H. Cheng, "Prior does matter: Visual navigation via denoising diffusion bridge models," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 12 100–12 110.

[18] J. Zhang, L. Dai, F. Meng, Q. Fan, X. Chen, K. Xu, and H. Wang, "3d-aware object goal navigation via simultaneous exploration and identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 6672–6682.

[19] H. Yin, X. Xu, Z. Wu, J. Zhou, and J. Lu, "Sg-nav: Online 3d scene graph prompting for llm-based zero-shot object navigation," *Advances in Neural Information Processing Systems*, vol. 37, pp. 5285–5307, 2024.

[20] Y. Zeng, H. Ren, S. Wang, J. Huang, and H. Cheng, "Navidiffusor: Cost-guided diffusion model for visual navigation," *arXiv preprint arXiv:2504.10003*, 2025.

[21] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*. PMLR, 2023, pp. 19 730–19 742.

[22] R. Shah, A. Yu, Y. Zhu, Y. Zhu, and R. Martín-Martín, "Bumble: Unifying reasoning and acting with vision-language models for building-wide mobile manipulation," *arXiv preprint arXiv:2410.06237*, 2024.

[23] K. Rana, J. Haviland, S. Garg, J. Abou-Chakra, I. Reid, and N. Suenderhauf, "Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning," *arXiv preprint arXiv:2307.06135*, 2023.

[24] S. Yenamandra, A. Ramachandran, K. Yadav *et al.*, "Homerobot: Open vocabulary mobile manipulation," 2023. [Online]. Available: https://github.com/facebookresearch/home-robot

[25] P. Liu, Y. Orru, J. Vakil, C. Paxton, N. M. M. Shafiullah, and L. Pinto, "Ok-robot: What really matters in integrating open-knowledge models for robotics," *arXiv preprint arXiv:2401.12202*, 2024.

[26] Z. Yan, S. Li, Z. Wang, L. Wu, H. Wang, J. Zhu, L. Chen, and J. Liu, "Dynamic open-vocabulary 3d scene graphs for long-term language-guided mobile manipulation," *IEEE Robotics and Automation Letters*, 2025.

[27] D. Qiu, W. Ma, Z. Pan, H. Xiong, and J. Liang, "Open-vocabulary mobile manipulation in unseen dynamic environments with 3d semantic maps," *arXiv preprint arXiv:2406.18115*, 2024.

[28] P. Zhi, Z. Zhang, Y. Zhao, M. Han, Z. Zhang, Z. Li, Z. Jiao, B. Jia, and S. Huang, "Closed-loop open-vocabulary mobile manipulation with gpt-4v," *arXiv preprint arXiv:2404.10220*, 2024.

[29] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software (TOMS)*, vol. 22, no. 4, pp. 469–483, 1996.

[30] T. Sorgente, S. Biasotti, and M. Spagnuolo, "Polyhedron kernel computation using a geometric approach," *Computers & Graphics*, vol. 105, pp. 94–104, 2022.

[31] C. Li, F. Xia, R. Martín-Martín *et al.*, "igibson 2.0: Object-centric simulation for robot learning of everyday household tasks," in *Proceedings of the 5th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, A. Faust, D. Hsu, and G. Neumann, Eds., vol. 164. PMLR, 08–11 Nov 2022, pp. 455–465. [Online]. Available: https://proceedings.mlr.press/v164/li22b.html

[32] P. Anderson, A. Chang, D. S. Chaplot, A. Dosovitskiy, S. Gupta, V. Koltun, J. Kosecka, J. Malik, R. Mottaghi, M. Savva *et al.*, "On evaluation of embodied navigation agents," *arXiv preprint arXiv:1807.06757*, 2018.

[33] P. He, J. Gao, and W. Chen, "Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing," *arXiv preprint arXiv:2111.09543*, 2021.

[34] W. Xu and F. Zhang, "Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3317–3324, 2021.

[35] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.