# HuLE-Nav: Human-Like Exploration for Zero-Shot Object Navigation via Vision-Language Models

**Peilong Han, Min Zhang, GuoTao Yu, Jianye Hao*, Hongyao Tang, Yan Zheng**
College of Intelligence and Computing, Tianjin University

## Abstract

Enabling robots to navigate as efficiently as humans in unknown environments is an attractive and challenging research goal in the field of embodied intelligence. Following the exploration behaviors of humans, we find that scene semantic understanding, scene spatio-temporal memory, and accumulated knowledge are all key elements to achieve efficient navigation. Inspired by this, we propose a zero-shot object navigation method, HuLE-Nav, which contains two core components: multi-dimensional semantic value maps for human-like exploration memory, human-like exploration processes with multi-dimensional semantic value maps. Specifically, HuLE-Nav first leverages the off-the-shelf Vision-Language Models (VLMs) and real-time observations to dynamically capture the semantic relevance between objects, the scene-level semantics, and spatio-temporal history of exploration paths, and jointly represent them as a multi-dimensional semantic value maps. Then, mimicking the active exploration behavior of humans, we further propose a dynamic exploration and replanning mechanism to flexibly update the long-term goal based on the real-time updated multi-dimensional semantic value maps. Finally, we propose a collision escape strategy based on the powerful reasoning and planning capabilities of VLMs to prevent robots from getting into collisions. The extensive evaluation of HM3D validates HuLE-Nav outperforms the best-performing competitor +7.3% success rate and +27.7% exploration efficiency, respectively.

## 1 Introduction

Understanding how humans navigate efficiently in unseen environments is crucial for developing robots that can mimic human exploration behavior. Typically, efficient human exploration behavior relies on scene semantic understanding, scene spatio-temporal memory, and accumulated knowledge. The first two provide humans with detailed scene semantic maps, and the accumulated knowledge can help humans efficiently decide where to explore next based on the semantic maps. Obviously, such a human decision-making process can naturally be mimicked by using pre-trained foundation models (LLMs and VLMs) with a large amount of commonsense knowledge and strong reasoning and generalization capabilities[1]. Inspired by this, a series of object goal navigation algorithms based on semantic maps and foundation models have been proposed and made substantial progress. For example, SemExp[2] first proposed a semantic maps construction method containing navigable areas, obstacles, and object semantic categories to mimic the map-based exploration process. Following the frontier-based exploration approach FBE[3], L3MVN[4] describes object-type information around map frontiers in textual form and then employs an LLM to reason about more valuable exploration locations. ESC[5] considers both object type and room type semantic information to help the LLM identify which frontier is most likely to contain instances of the target object. LGX[6] also translates visual information into text and uses the LLM for search planning. Further, PixNav[7] uses the foundation models and specifies navigation goals in pixel units to achieve generalized navigation

*Corresponding author: Jianye Hao (jianye.hao@tju.edu.cn)

NeurIPS 2024 Workshop on Behavioral Machine Learning.

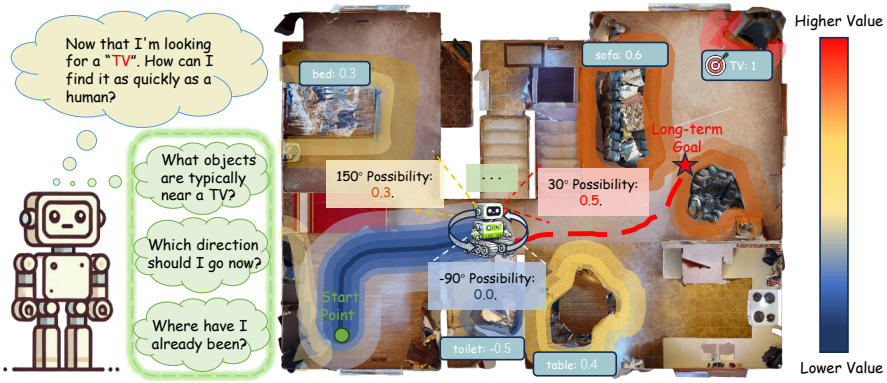


Figure 1: HuLE-Nav addresses three aspects: object-level semantic relevance analysis, scene-level semantic understanding and exploration direction reasoning, and non-repeated exploration of paths.

across object types. VoroNav[8] feeds textual descriptions of paths and images into the Large Language Model (LLM) to prompt the LLM to apply commonsense knowledge to reason about navigation waypoints.

However, these methods mainly consider single spatial semantic clues such as object type, room type, and path, and only convert these semantic clues into textual information for the LLM. Compared with the scene semantics and spatio-temporal memory acquired by humans from the environment, the pre-trained foundation model obtains very limited environmental information, which is very challenging for it to rely on commonsense knowledge to reason about the next exploration location (see Fig.1). To address this problem, we propose the real-time updated multi-dimensional semantic value maps based on VLMs to mimic human-acquired scene semantics and scene spatio-temporal memory as much as possible. Specifically, the proposed multi-dimensional semantic value maps encapsulate object semantics, inter-object semantic relevance, scene-level semantics, and the spatio-temporal history of exploration paths, navigable areas, and obstacles. Based on the multi-dimensional semantic value map, we further propose a dynamic exploration and replanning mechanism to mimic the active exploration behavior of humans and combine it with a collision escape strategy to prevent the robot from getting into collisions. We name our approach HuLE-Nav (Human-Like Exploration for Navigation).

Our contributions can be summarized as follows: (1) for the first time, we highly reproduce the scene semantics and scene spatio-temporal memory necessary for humans to perform navigation decisions through real-time multi-dimensional semantic value maps; (2) we actively apply the scene understanding and action planning capabilities of VLMs, and propose a dynamic exploration and replanning mechanism driven by scene semantic updates and a collision escape strategy to mimic human; and (3) we develop a complete map-based human-like navigation method, which realizes effective integration of different modules and state-of-the-art performance on the Habitat platform.

## 2 HuLE-Nav Approach

In Fig. 2, we illustrate the complete architecture of HuLE-Nav. Specifically, Sec. 2.1 introduces the specific construction method of the multi-dimensional semantic value maps. Sec. 2.2 further explains the human-like navigation process with multi-dimensional semantic value maps.

### 2.1 Multi-dimensional Semantic Value Maps for Human-like Exploration Memory

**Semantic Value Map Overview.** To construct human-like exploration memory, we create a multi-dimensional semantic value map $m_t$ initialized to zero, which is a $K \times M \times M$ matrix. Among them, $M \times M$ represents the map size and $K = C+5$ is the number of channels. $C$ and the first two channels represent the total number of semantic categories, navigation areas, and obstacles, respectively (see App. A.1 for more details). The remaining three channels are specially designed for this study to capture the semantic relevance between objects, the scene-level semantic information, and the

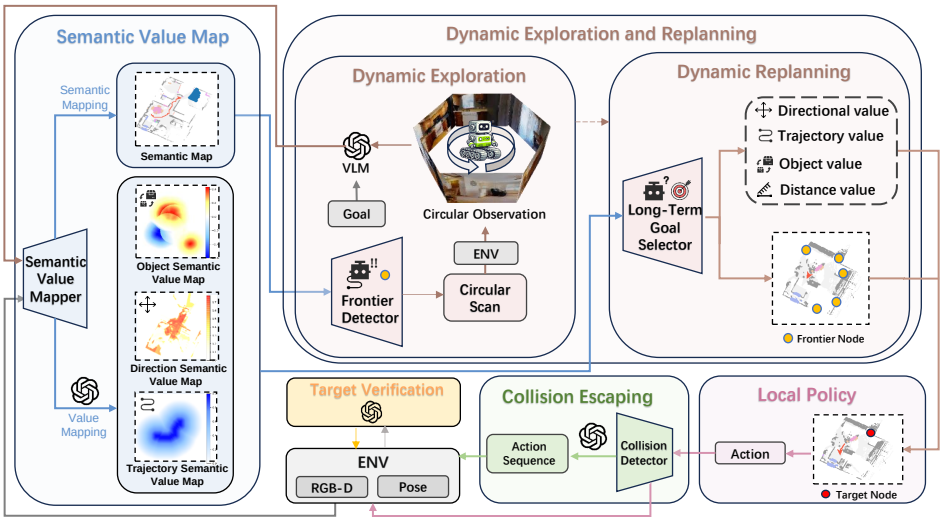


Figure 2: HuLE-Nav includes two main components: Semantic Value Maps and Dynamic Exploration and Replanning, supported by auxiliary functions such as Collision Escaping and Local Policy.

spatio-temporal history of exploration paths, respectively, so as to highly reproduce human-like exploration memory.

**Object Semantic Value Map.** To encourage the agent to search for the target object around more relevant objects quickly, we propose for the first time to radiate the semantic relevance between the target object and other objects to their surrounding areas on the Object Semantic Value Map. Specifically, at task initiation, the VLM assigns semantic relevance values $S_{o_i} \in [-1, 1]$ between each object instance $o_i$ and the target object, with larger positive values indicating a higher likelihood of the two objects co-occurring, and projects these values onto the map as shown in Eq. 1. Then, for each frontier point $p$ within the frontier set $P$ on the map, the value $S_o(p)$ is determined by the object with the highest semantic impact as shown in Eq. 2.

$$S'_o(p, o_i) = S_{o_i} \cdot \left(1 - \frac{d_t(p, o_i)}{r}\right) \cdot \mathbb{I}(d_t(p, o_i) \leq r) \quad \forall p \in P, \tag{1}$$

$$S_o(p) = S'_o(p, o_{i_{\max}}), \quad \text{where} \quad o_{i_{\max}} = \arg\max_{o_i \in \mathcal{O}} |S'_o(p, o_i)| \quad \forall p \in P, \tag{2}$$

where $\mathcal{O}$ represents the set of all objects in the semantic maps, $d_t(p, o_i)$ denotes the minimum distance from point $p$ to the nearest point within the cluster of object $o_i$, and $r$ is the distance threshold, beyond which the value is set to zero.

**Direction Semantic Value Map.** To break the object semantic information bottleneck, we employ VLM to extract scene-level semantic cues to quickly infer the most appropriate exploration direction. Specifically, we maintain a cumulative record of optimal direction choices on the Direction Semantic Value Map. At task initiation and each observation point, the agent performs circular scans to capture six equidistant RGB observations,$\{I_0, \ldots, I_5\}$, along with corresponding pose information. The VLM evaluates these images for the potential presence of the target object $G$ (as shown in Eq. 3), projecting the results onto the corresponding pixels on the map using depth and pose information, with overlapping projections averaged on the same pixel.

$$S_{d_i} = \text{VLM}(I_i, G), \quad i = 0, 1, \ldots, 5 \quad | \quad \sum_{i=0}^{5} S_{d_i} = 1. \tag{3}$$

**Trajectory Semantic Value Map.** To prevent the agent from repeatedly traversing the same paths or getting stuck at the same target point during exploration, we created a Trajectory Semantic Value Map, which assigns lower values $S_t$ around the trajectory $T$, encouraging the agent to explore new

and diverse paths. The semantic value $S_t$ for each frontier point $p$ within the frontier set $P$ on the trajectory semantic value map is given by Eq. 4 and 5.

$$d_t = \min_{t \in T} \|p - t\|, \quad \mathcal{N}(p, r) = \{t \in T \mid \|p - t\| \leq r\} \quad \forall p \in P, \tag{4}$$

$$S_t(p) = -\left(1 - \frac{d_t}{r}\right) \cdot \left(\frac{|\mathcal{N}(p, r)|}{\lambda + |\mathcal{N}(p, r)|}\right) \cdot \mathbb{I}(d_t(p, o_i) \leq r) \quad \forall p \in P, \tag{5}$$

where $d_t$ is the minimum distance from the point $p$ to the nearest trajectory point, and $\mathcal{N}(p, r)$ is the set of trajectory points within a radius $r$ around the point $p$. $\lambda$ is a regularization parameter.

### 2.2 Human-like Exploration Process with Multi-dimensional Semantic Value Maps

**Dynamic Exploration and Replanning.** To encourage the agent to actively look around the environment like a human, capture environmental information as quickly as possible, and adjust the long-term goal flexibly while moving towards the next long-term goal, we propose a dynamic exploration and planning mechanism. Specifically, during the navigation process, the agent's current location $p_l$ becomes an observation point $p_o$ if it has a direct line of sight to a frontier point $p_f$, where $p_f$ is the centroid of a connected region within the set of candidate target points $P$ (denoted as $\mathcal{C}(P)$), with no intervening obstacles $\mathcal{B}$, as shown in Eq. 6. At $p_o$, the agent performs a circular scan, then the agent selects the frontier point $p$ with the highest semantic value $S$ from the frontier map as the new long-term goal. As shown in Eq. 7, the semantic value is calculated by combining a weighted sum of the three dimensions of the semantic map with the normalized distance between the current position and the candidate target points. We defer the details of local policy to App. A.2.

$$p_o = \{p_l \mid \exists p_f \in \mathcal{C}(P), \text{line}(p_l, p_f) \cap \mathcal{B} = \emptyset\}. \tag{6}$$

$$L = \arg\max_{p \in P} \left(S_d(p) + \alpha S_t(p) + \beta S_o(p) - \gamma d_{\text{norm}}(p)\right) \quad \forall p \in P. \tag{7}$$

**Collision Escape Strategy.** The robot getting stuck is a major factor in navigation failure. To address this problem, we propose a VLM-based escape strategy. Specifically, once the long-term goal has not been updated for a long time, our algorithm will activate the escape strategy, allowing the VLM to give an action plan containing 10 actions based on the robot's current position and observations. Based on the action plan, the robot can effectively escape from the current collision.

## 3 Experiments

**Experiment Setting.** For a fair comparison with recent methods, we also evaluate our method on the HM3D[9] dataset in Habitat simulator[10]. We evaluate all approaches using two metrics: success rate (SR) and Success weighted by inverse Path Length (SPL) [11] (see App. A.3 for more details).

Table 1: Comparison with SOTA methods on HM3D.

| Method | Zero-Shot | SR↑ | SPL↑ |
|---|---|---|---|
| Random | ✓ | 0.00 | 0.00 |
| FBE(2023)[3] | ✓ | 23.7 | 12.3 |
| SemExp(2020)[2] | × | 37.9 | 18.8 |
| L3MVN(2023)[4] | ✓ | 50.4 | 23.1 |
| Pixel-Nav(2023)[7] | × | 37.9 | 20.5 |
| ESC(2023)[5] | ✓ | 39.2 | 22.3 |
| VoroNav(2024)[8] | ✓ | 42.0 | 26.0 |
| **HuLE-Nav(Ours)** | ✓ | **54.1** | **33.2** |

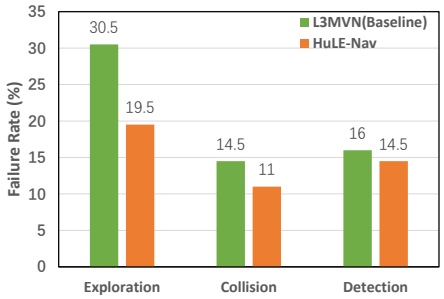

Figure 3: Failure Cases Analysis.

**Experiment Results.** From Tab 1, in terms of SR, HuLE-Nav improves the SR by **128.2%** over FBE[3] and by **7.3%** compared to the second-best L3MVN[4]. In terms of SPL, HuLE-Nav surpasses all other algorithms by a large margin, and its exploration efficiency is **2.7** times that of FBE[3] and **1.28** times that of the second-best VoroNav[8]. To further illustrate the effectiveness of our algorithm

in mimicking human exploration behavior and escaping collisions, we conduct a detailed cause analysis of specific navigation failure cases. From the results in Fig.3, compared with the baseline L3MVN, HuLE-Nav shows lower failure rates in exploration, collision, and detection. In particular, the failure rates due to exploration and collision are reduced by **36.1%** and **24.1%**, respectively.

## 4 Conclusion

In this work, we introduce HuLE-Nav, a novel approach for zero-shot object navigation that mimics human exploration behavior by multi-dimensional semantic value maps, and active exploration and dynamic planning mechanism. Despite its significant performance improvements, HuLE-Nav still has limitations, including underutilization of VLM's full potential and occasional missed detections near targets due to its frontier-based approach. Future work will focus on exploiting the potential of VLMs for navigation tasks and advancing the development of navigation algorithms.

## References


[1] Min Zhang et al. "MFE-ETP: A Comprehensive Evaluation Benchmark for Multi-modal Foundation Models on Embodied Task Planning". In: *arXiv preprint arXiv:2407.05047* (2024).

[2] Devendra Singh Chaplot et al. "Object goal navigation using goal-oriented semantic exploration". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 4247–4258.

[3] Theophile Gervet et al. "Navigating to objects in the real world". In: *Science Robotics* 8.79 (2023), eadf6991.

[4] Bangguo Yu, Hamidreza Kasaei, and Ming Cao. "L3mvn: Leveraging large language models for visual target navigation". In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2023, pp. 3554–3560.

[5] Kaiwen Zhou et al. "Esc: Exploration with soft commonsense constraints for zero-shot object navigation". In: *International Conference on Machine Learning*. PMLR. 2023, pp. 42829–42842.

[6] Vishnu Sashank Dorbala, James F Mullen Jr, and Dinesh Manocha. "Can an embodied agent find your "cat-shaped mug"? llm-based zero-shot object navigation". In: *IEEE Robotics and Automation Letters* (2023).

[7] Wenzhe Cai et al. "Bridging zero-shot object navigation and foundation models through pixel-guided navigation skill". In: *arXiv preprint arXiv:2309.10309* (2023).

[8] Pengying Wu et al. "Voronav: Voronoi-based zero-shot object navigation with large language model". In: *arXiv preprint arXiv:2401.02695* (2024).

[9] Manolis Savva et al. "Habitat: A platform for embodied ai research". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 9339–9347.

[10] Santhosh K Ramakrishnan et al. "Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai". In: *arXiv preprint arXiv:2109.08238* (2021).

[11] Peter Anderson et al. "On evaluation of embodied navigation agents". In: *arXiv preprint arXiv:1807.06757* (2018).

[12] Samir Yitzhak Gadre et al. "Clip on wheels: Zero-shot object navigation as object localization and exploration". In: *arXiv preprint arXiv:2203.10421* 3.4 (2022), p. 7.

[13] Arjun Majumdar et al. "Zson: Zero-shot object-goal navigation using multimodal goal embeddings". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 32340–32352.

[14] Qianfan Zhao et al. "Zero-shot object goal visual navigation". In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2023, pp. 2025–2031.

[15] Samir Yitzhak Gadre et al. "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. 23171–23181.

[16] Yan Zheng et al. "Wuji: Automatic Online Combat Game Testing Using Evolutionary Deep Reinforcement Learning". In: *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. 2019, pp. 772–784. DOI: 10.1109/ASE.2019.00077.

[17] Yuxuan Kuang, Hai Lin, and Meng Jiang. "OpenFMNav: Towards Open-Set Zero-Shot Object Navigation via Vision-Language Foundation Models". In: *arXiv preprint arXiv:2402.10670* (2024).

[18] Junting Chen et al. "How to not train your dragon: Training-free embodied object goal navigation with semantic frontiers". In: *arXiv preprint arXiv:2305.16925* (2023).

[19] Chenguang Huang et al. "Visual language maps for robot navigation". In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2023, pp. 10608–10615.

[20] Roberto Bigazzi et al. "Mapping High-level Semantic Regions in Indoor Environments without Object Recognition". In: *arXiv preprint arXiv:2403.07076* (2024).

[21] Leyuan Sun et al. "Leveraging Large Language Model-based Room-Object Relationships Knowledge for Enhancing Multimodal-Input Object Goal Navigation". In: *arXiv preprint arXiv:2403.14163* (2024).

[22] Chengguang Xu et al. "Vision and Language Navigation in the Real World via Online Visual Language Mapping". In: *arXiv preprint arXiv:2310.10822* (2023).

[23] Matthew Chang et al. "Goat: Go to any thing". In: *arXiv preprint arXiv:2311.06430* (2023).

[24] Dhruv Shah et al. "Navigation with large language models: Semantic guesswork as a heuristic for planning". In: *Conference on Robot Learning*. PMLR. 2023, pp. 2683–2699.

[25] Yuxing Long et al. "InstructNav: Zero-shot System for Generic Instruction Navigation in Unexplored Environment". In: *arXiv preprint arXiv:2406.04882* (2024).

[26] Naoki Yokoyama et al. "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.

[27] Lingfeng Zhang et al. "TriHelper: Zero-Shot Object Navigation with Dynamic Assistance". In: *arXiv preprint arXiv:2403.15223* (2024).

[28] Santhosh Kumar Ramakrishnan et al. "Poni: Potential functions for objectgoal navigation with interaction-free learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 18890–18900.

[29] James A Sethian. "A fast marching level set method for monotonically advancing fronts." In: *proceedings of the National Academy of Sciences* 93.4 (1996), pp. 1591–1595.

[30] Karmesh Yadav et al. "Habitat-matterport 3d semantics dataset". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 4927–4936.

[31] Karmesh Yadav et al. *Habitat Challenge 2022*. `https://aihabitat.org/challenge/2022/`. 2022.

[32] Abhinav Gupta et al. "Robot learning in homes: Improving generalization and reducing dataset bias". In: *Advances in neural information processing systems* 31 (2018).

[33] Jindong Jiang et al. "Rednet: Residual encoder-decoder network for indoor rgb-d semantic segmentation". In: *arXiv preprint arXiv:1806.01054* (2018).

[34] Joel Ye et al. "Auxiliary tasks and exploration enable objectgoal navigation". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 16117–16126.

# A Appendix

## A.1 More Details of Preliminary

**Zero-Shot Object Navigation.** Approaches to Zero-Shot Object Navigation (ZSON) fall into two main categories: map-less methods that use reinforcement or imitation learning[12, 13, 14, 15, 16], and map-based methods that store historical environment information in semantic top-down maps to guide waypoint selection[2, 17, 18]. Map-based methods primarily focus on constructing detailed semantic maps[19, 2, 20, 21, 22]. Recent approaches enhance these maps by integrating frontier-based exploration strategies with large language models for more efficient frontier selection[23, 4, 24]. Furthermore, several methods improve navigation efficiency and adaptability through path-planning algorithms[8], complementary mapping techniques[25, 26], or auxiliary tools[27].

**Episodic Semantic Map.** Semantic Map construct and update a $(K+2) \times M \times M$ map using RGB-D images and poses, where $M$ denotes the dimensions of the map's width and height, and $K+2$ represents the total number of channels in the map. Specifically, $K$ channels represent the semantic channels of the detected objects, 2 channels correspond to an obstacle map and an explored map. Given RGB-D images and the agent's poses at each time step, we can obtain 3D point clouds. The 3D point clouds are projected onto a top-down 2D map by judging the height, resulting in an obstacle map and an explored map, which represent navigable areas and non-navigable obstacle areas, respectively. Simultaneously, the RGB images are used to predict the category masks and filter out specific object categories. These are aligned with the 3D semantic point clouds and ultimately projected onto the corresponding $K$ semantic channels.

**Frontier Map.** We derive the frontier map through a multi-step process that integrates information from both the explored and obstacle maps, adhering to the methodology proposed in [28]. This process entails extracting the explored edge via maximum contour identification from the explored map, followed by edge dilation of the obstacle map. The frontier map is then generated by computing the difference between these processed maps. Subsequently, we employ connected component analysis to identify and cluster frontier cells into coherent chains. The centroids of these frontier connected components serve as potential candidates for long-term goals, effectively balancing exploration and obstacle avoidance.

### A.2 More Details of Method

**Local Policy.** To navigate from the agent's current position to its long-term objective, we utilize the Fast Marching Method (FMM)[29]. The agent then identifies a local goal within a constrained radius of its present location and executes the optimal action to progress towards this proximal target. At each timestep, both the local map and the immediate goal are dynamically updated to incorporate new sensory information. This modular policy approach significantly enhances training efficiency and eliminates the need for explicitly learning obstacle avoidance behaviors.

**Semantic Value Map.** The Semantic Value Map assigns a value to each pixel in the exploration area, quantifying its semantic importance for locating the target object. This value is a parameterized sum of three dimensions: Direction Semantic Value Map, Trajectory Semantic Value Map, and Object Semantic Value Map. The value map is used to evaluate each frontier, with the highest-valued frontier selected for the next exploration step. The Direction Semantic Value Map is iteratively built using depth and pose information to construct a top-down map, where VLM-provided probabilities are projected onto the corresponding map pixels. When probabilities from different directions are projected onto the same pixel, their average is calculated. The Trajectory Semantic Value Map calculates pixel values based on the agent's trajectory path, while the Object Semantic Value Map computes pixel values based on the most influential value from the object list.

### A.3 More Details of Experiment

**Task Definition.** Object navigation tasks challenge agents to locate specific objects within indoor environments, with target categories including beds, chairs, sofas, TVs, plants, and toilets. The agent operates in a discrete action space comprising Stop, MoveForward, TurnLeft, TurnRight, LookUp, and LookDown, with 0.25m movements and 30° rotations. Success is achieved when the agent stops within 0.1m of the target, while failure occurs if the 500-step limit is exceeded (exploration failure), if the agent stops at an incorrect object (detection error), or if it becomes trapped due to insufficient long-term goal updates (collision error). This task evaluates an agent's ability to efficiently navigate, recognize objects, and make decisions in complex indoor spaces.

**Experiment Setup.** The evaluations conducted on HM3D[30] using Habitat Simulator[9] adhere to the parameters established in the Habitat ObjectNav Challenge [31]. The agent is modeled after a LoCoBot[32] with a base radius of 0.18m. It is equipped with an RGB-D camera mounted at a height of 0.88 meters and a pose sensor that provides precise localization. The camera features a 79° Horizontal Field of View (HFoV) and captures frames with dimensions of 480 × 640 pixels. For category prediction across all classes, we employed a finetuned RedNet model[33], following the approach outlined in[34].In the experiment, GPT-4o was used as the VLM. The parameters for the Trajectory Semantic Value Map and Object Semantic Value Map in Eq. 4, 5, and 1 were set with

$r = 30$, while the weights for $\alpha$, $\beta$, and $\gamma$ in Eq. 7. were set to 0.5, 0.3, and 0.1, respectively. And $\lambda$ is set as 10 in Eq. 5.

**Experiment Baselines.** In this work, we compare HuLE-Nav against several baselines:

- **Random Exploration**: A classical baseline that drives the robot to randomly sampled points in unexplored areas.
- **FBE** [3]: This method employs a classical robotics pipeline for mapping and uses a frontier-based exploration strategy to navigate in unfamiliar environments.
- **SemExp** [2]: A semantic map-based method that integrates reinforcement learning to explore and search for the target, relying on pre-trained semantic models.
- **L3MVN** [4]: An LLM-based approach that finetunes a an LLM to conduct frontier-based exploration.
- **Pixel-Nav** [7]: A approach that utilizes foundation models to select navigation pixels from panoramic images and trains a locomotion module to move towards the selected pixels.
- **ESC** [5]: A map-based zero-shot object navigation baseline that combines object and room detection using GLIP and integrates large language models (LLM) with soft commonsense constraints to guide planning.

**Experiment Examples.** We analyzed the success rate for each target type in the experiment and compared it with the baseline L3MVN. As shown in Fig. 4, HuLE-Nav achieves a higher exploration success rate than the baseline across all object categories. We present several examples during the experiment. Tab. 2 shows the pairwise relationship degrees between objects provided by GPT-4o, where one row will be used in the object semantic map during the experiment. Fig. 5, 6, 7, and 8 illustrate some typical examples and processes encountered in various parts of the experiment.

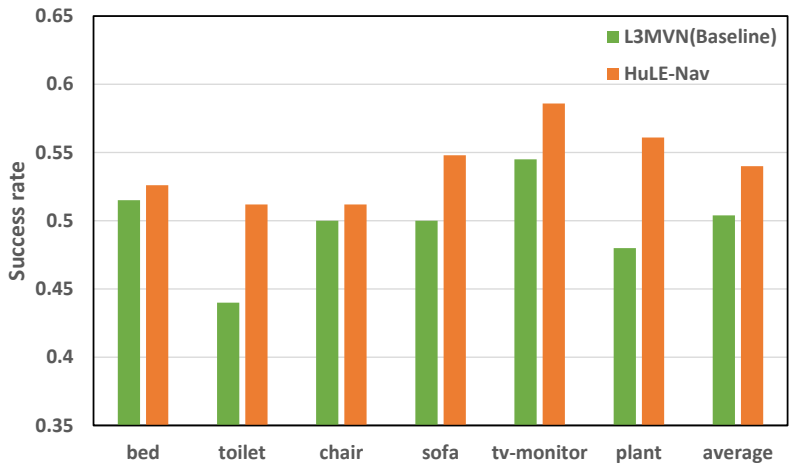


Figure 4: The success rate for each target type in the experiment and compared with L3MVN.

Table 2: The Object Correlation Table provided by GPT-4o indicates that higher values represent stronger relationships between objects, meaning they are more likely to appear together, while lower values suggest they are less likely to co-occur.

| Object | Chair | Sofa | Plant | Bed | Toilet | TV Monitor | Bathtub | Shower | Fireplace | Appliances | Towel | Sink | Chest of Drawers | Table | Stairs |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Chair** | 1 | 0.75 | 0.2 | 0.4 | -0.3 | 0.5 | -0.6 | -0.5 | 0.3 | 0.1 | 0.1 | -0.2 | 0.5 | 0.7 | 0.1 |
| **Sofa** | 0.75 | 1 | 0.3 | 0.5 | -0.4 | 0.6 | -0.5 | -0.5 | 0.4 | 0.2 | 0.2 | -0.2 | 0.6 | 0.8 | 0.2 |
| **Plant** | 0.2 | 0.3 | 1 | 0.1 | -0.2 | 0.2 | -0.2 | -0.3 | 0.2 | 0.1 | 0.3 | 0.2 | 0.2 | 0.3 | 0.1 |
| **Bed** | 0.4 | 0.5 | 0.1 | 1 | -0.6 | 0.3 | -0.3 | -0.4 | 0.2 | 0.1 | 0.1 | -0.5 | 0.6 | 0.5 | 0.1 |
| **Toilet** | -0.3 | -0.4 | -0.2 | -0.6 | 1 | -0.5 | 0.6 | 0.7 | -0.2 | -0.3 | 0.5 | 0.6 | -0.5 | -0.4 | 0.2 |
| **TV Monitor** | 0.5 | 0.6 | 0.2 | 0.3 | -0.5 | 1 | -0.5 | -0.4 | 0.3 | 0.2 | 0.1 | -0.2 | 0.5 | 0.6 | 0.1 |
| **Bathtub** | -0.6 | -0.5 | -0.2 | -0.3 | 0.6 | -0.5 | 1 | 0.8 | -0.2 | -0.3 | 0.4 | 0.5 | -0.5 | -0.4 | 0.1 |
| **Shower** | -0.5 | -0.5 | -0.3 | -0.4 | 0.7 | -0.4 | 0.8 | 1 | -0.3 | -0.4 | 0.5 | 0.6 | -0.6 | -0.5 | 0.1 |
| **Fireplace** | 0.3 | 0.4 | 0.2 | 0.2 | -0.2 | 0.3 | -0.2 | -0.3 | 1 | 0.2 | 0.2 | -0.1 | 0.3 | 0.4 | 0.2 |
| **Appliances** | 0.1 | 0.2 | 0.1 | 0.1 | -0.3 | 0.2 | -0.3 | -0.4 | 0.2 | 1 | 0.2 | 0.3 | 0.2 | 0.2 | 0.3 |
| **Towel** | 0.1 | 0.2 | 0.3 | 0.1 | 0.5 | 0.1 | 0.4 | 0.5 | 0.2 | 0.2 | 1 | 0.5 | 0.1 | 0.2 | 0.1 |
| **Sink** | -0.2 | -0.2 | 0.2 | -0.5 | 0.6 | -0.2 | 0.5 | 0.6 | -0.1 | 0.3 | 0.5 | 1 | -0.4 | -0.3 | 0.2 |
| **Chest of Drawers** | 0.5 | 0.6 | 0.2 | 0.6 | -0.5 | 0.5 | -0.5 | -0.6 | 0.3 | 0.2 | 0.1 | -0.4 | 1 | 0.6 | 0.1 |
| **Table** | 0.7 | 0.8 | 0.3 | 0.5 | -0.4 | 0.6 | -0.4 | -0.5 | 0.4 | 0.2 | 0.2 | -0.3 | 0.6 | 1 | 0.2 |
| **Stairs** | 0.1 | 0.2 | 0.1 | 0.1 | 0.2 | 0.1 | 0.1 | 0.1 | 0.2 | 0.3 | 0.1 | 0.2 | 0.1 | 0.2 | 1 |

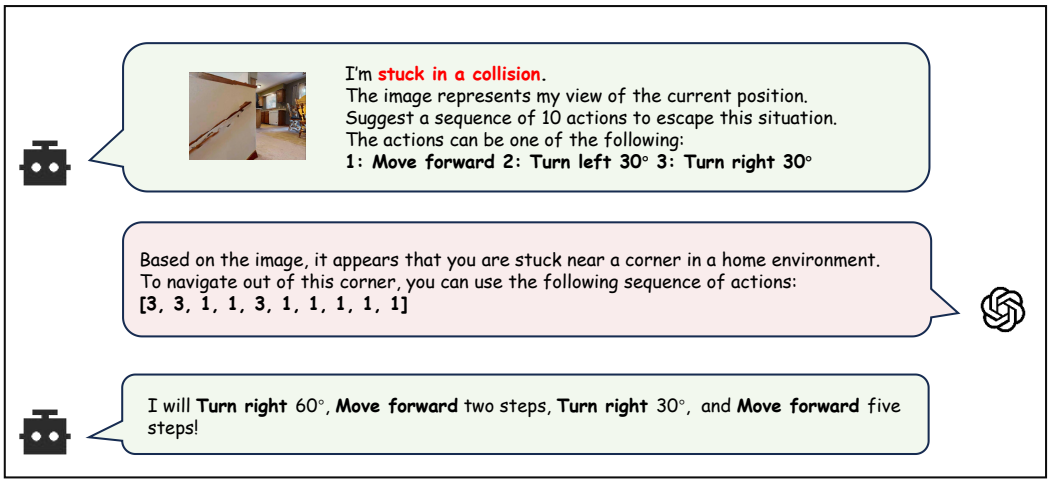


Figure 5: Example of HuLE-Nav collision escape: When the robot encounters a deadlock, GPT-4o generates an action sequence based on the robot's current observations. The robot then executes this sequence to successfully escape the trapped situation.


Figure 6: Example of HuLE-Nav navigation process: The robot's main steps in a task to find a "bed," from initialization to task completion.

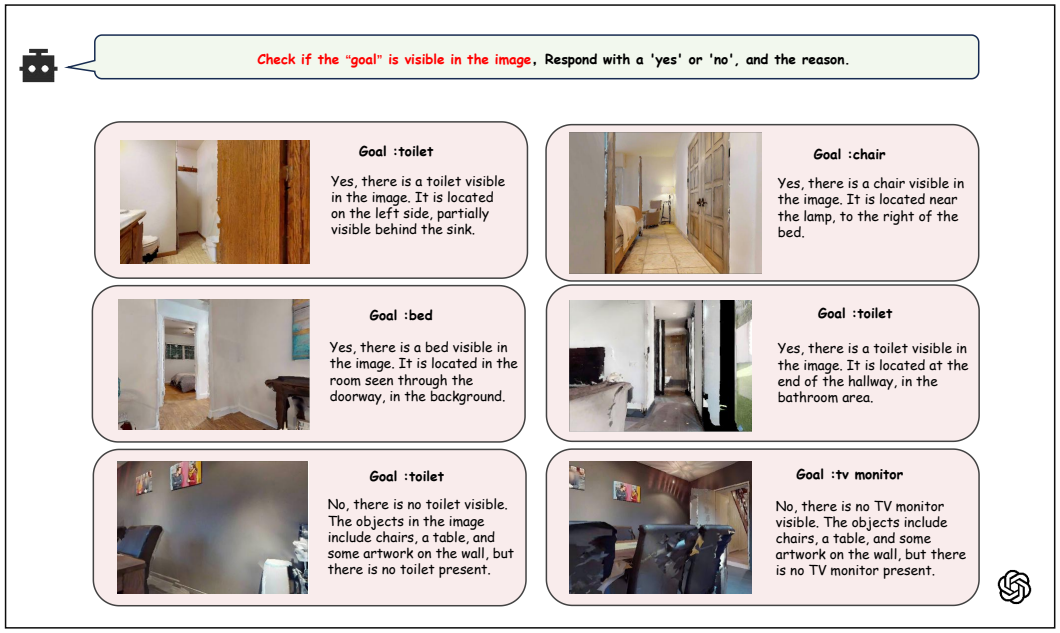


Figure 7: Example of HuLE-Nav target verification: After the target detector identifies the object, GPT-4o is used to verify and confirm the detection.

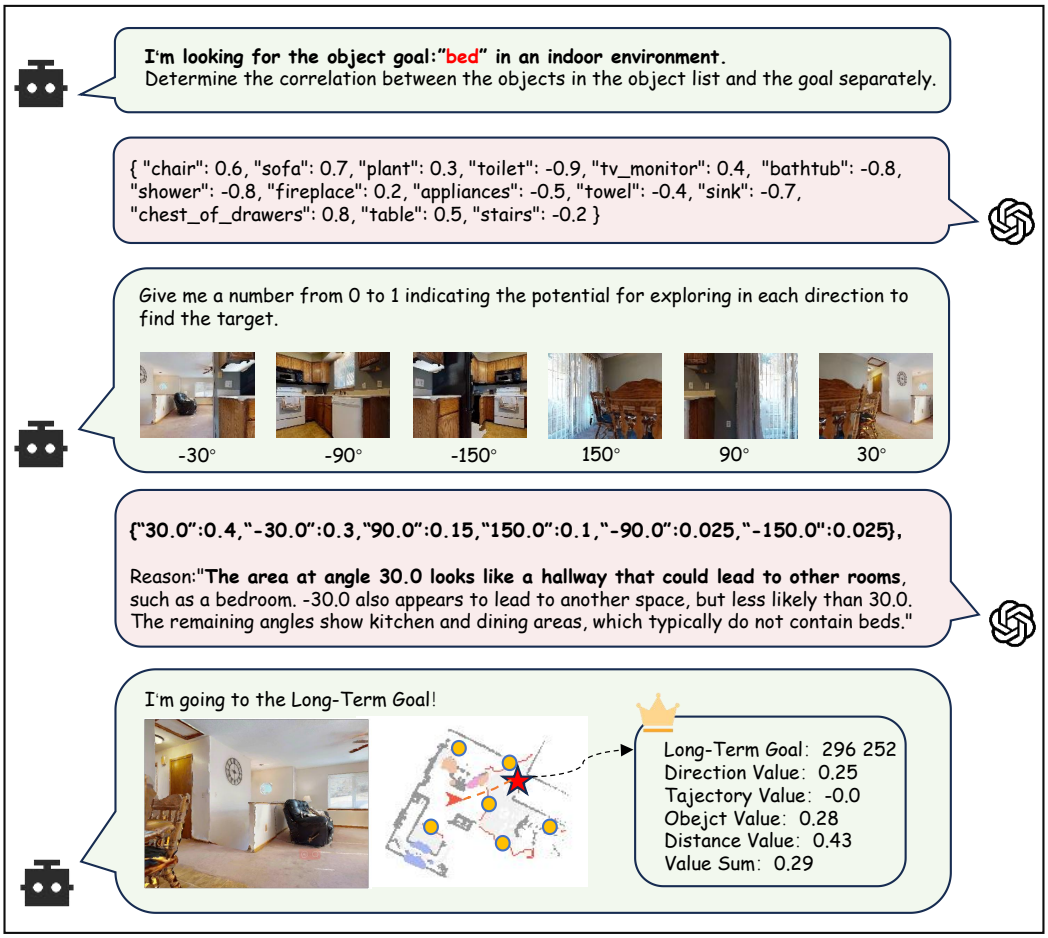


Figure 8: Example of HuLE-Nav circular scan initialization for the semantic value map decision-making based on the updated semantic value map.