Hierarchical 3D Scene Graph Construction and Belief-based Planning for Semantic Navigation

Bing Wu, Zuyao Chen, and Changwen Chen *

The Hong Kong Polytechnic University
{bing-comp.wu,zuyao.chen}@connect.polyu.hk
changwen.chen@polyu.edu.hk

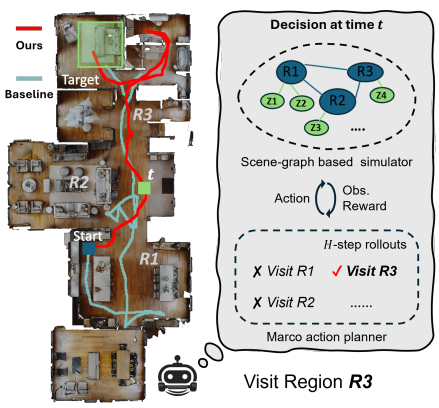
Abstract. Semantic navigation is a fundamental task for embodied agents operating in unseen environments, requiring both semantic understanding and long-term decision-making. Recent foundation models have empowered agents with rich semantic priors for this task. However, without structured global representations, decision-making often falls back on local observations and greedy strategies, resulting in inefficient exploration and myopic behaviors, especially in long-distance navigation. To address these challenges, we propose a zero-shot semantic navigation framework. Our method incrementally maintains an online Hierarchical 3D Scene Graph (HSG) to form a multi-granular semantic topology over objects, zones, and regions, serving as a compact state abstraction for global planning. Building on this memory, we introduce a hierarchical belief-based planning framework that fuses semantic priors with exploration evidence on the HSG, and performs finite-horizon rollouts on an HSG-based simulator to explicitly estimate the long-term expected returns of candidate macro-actions. This enables globally consistent decisions and reduces redundant backtracking. Extensive experiments in high-fidelity simulation environments across multiple tasks and datasets demonstrate that our method outperforms existing state-of-the-art methods, particularly in long-distance scenarios, where our approach improves SR and SPL by an average of 9.4% and 5.0%, respectively.

Keywords: Zero-Shot Semantic Navigation · Hierarchical 3D Scene Graph · Long-Horizon Planning

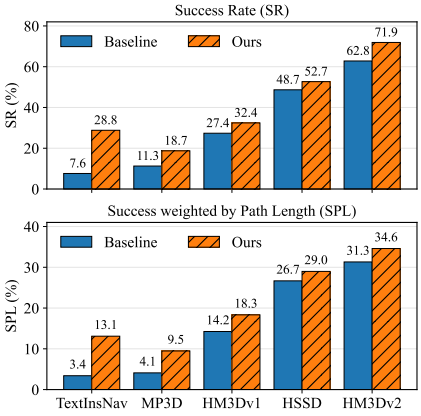
1 Introduction

Semantic navigation is not merely about *where an agent can go* but *where it should go*. In unseen environments, an agent must navigate to a semantic target using only egocentric observations under a limited action budget. The target can be specified as an object category (ObjectNav [6]) or a textual description of a specific instance (TextInstanceNav [23]), which requires joint semantic-spatial perception and long-horizon planning. Recent foundation models have advanced open-vocabulary perception and commonsense reasoning, making zero-shot semantic navigation feasible without task-specific retraining [41].

* Corresponding author.



(a) Planning on the HSG.



(b) Performance on long-distance navigation.

Fig. 1: Hierarchical belief-based planning for long-distance navigation. (a) Unlike the greedy exploration of baselines (cyan path), our agent evaluates long-term expected returns via rollouts on the HSG simulator, enabling globally efficient navigation (red path). (b) Our method outperforms state-of-the-art baselines (ApexNav [37] for ObjectNav, UniGoal [33] for TextInstanceNav) across multiple datasets in long-distance scenarios.

However, despite their powerful perception and commonsense reasoning, foundation models do not directly address the core challenges of spatial memory and long-horizon decision making in semantic navigation [29]. Most existing zero-shot methods lack a global structured semantic representation of the environment and instead rely on local observations with greedy or heuristic search strategies [32, 34, 41], which often results in myopic exploration with redundant revisits and frequent backtracking. Moreover, when foundation model priors misalign with the actual scene, the agent often lacks the explicit observation feedback required to update its belief state. Such errors can accumulate over time, especially in long-distance navigation.

To address these limitations, the agent requires long-horizon planning capabilities, which inherently demand a structured state space with multi-granular semantic abstractions. Recently, Hierarchical 3D Scene Graphs (HSGs) have emerged as a compact and semantically rich environmental representation. However, most existing methods focus on offline or passive reconstruction from complete point clouds [27] or RGB-D sequences [9], assuming full global observability. This offline construction paradigm cannot be directly applied to online exploration tasks constrained by partial observations. Therefore, how to incrementally construct such a hierarchical structure online in unseen environments and integrate it into the decision-making process remains an open challenge.

To bridge this gap, we propose an online hierarchical scene-graph navigation framework that couples *incremental hierarchical 3D scene graph construction* with *belief-based hierarchical planning*. As shown in Fig. 1a, HSG rollouts enable

long-horizon planning beyond greedy exploration. In the perception module, we maintain observed 3D object instances from RGB-D observations and establish spatial adjacency among objects via Generalized Voronoi Diagram (GVD). Leveraging spectral clustering on combined geometric and semantic features, we incrementally construct a bottom-up, multi-granular semantic topology with an Object–Zone–Region hierarchy (Sec. 4). In the decision module, we maintain a hierarchical belief state over the HSG_t, which combines large language model (LLM) semantic priors and frontier information gain evidence. We then construct a scene-graph-based simulator to support finite-horizon lookahead with Partially Observable Monte Carlo Tree Search (POUCT) [22]. By simulating state transitions and observation feedback on the HSG, the planner explicitly evaluates the long-term expected returns of candidate macro-actions to select the optimal action. Finally, a local planner executes the selected macro-action by optimizing the local path and generating low-level commands (Sec. 5).

In short, the contributions of this work can be summarized as follows:

- *Online incremental hierarchical 3D scene graph construction.* We propose a bottom-up approach combining GVD and spectral clustering to abstract local observations into a multi-granular semantic topology, providing a structured global memory.
- *Hierarchical belief-based planning.* We introduce a planning method that fuses LLM priors with observation feedback. By evaluating long-term expected returns via rollouts on the HSG, it effectively overcomes the myopia of greedy strategies.
- *Superior long-distance performance.* Extensive experiments across multiple tasks and datasets demonstrate that our method significantly outperforms state-of-the-art approaches. Most notably, in long-distance episodes, we improve SR and SPL by an average of 9.4% and 5.0%, respectively.

2 Related Work

3D Scene Graph Representation. Compared to traditional geometric maps (*e.g.*, point clouds, voxels) or topological maps, 3D scene graphs have recently emerged as a compact representation that jointly encodes geometry, semantics, and relational structure in 3D environments [2]. Building on this idea, robotic systems such as Kimera [20] and Hydra [9] couple SLAM with semantic parsing to construct 3D scene graphs online, allowing an agent to maintain a sparse, object-centric map of the environment rather than only a dense metric map. More recent works [8, 13] further explore hierarchical and open-vocabulary 3D scene graphs. In hierarchical formulations, nodes are organized into multiple abstraction levels, *e.g.*, building, floor, room, and object, so that the environment is described at different granularities within a unified structure, which in turn supports richer and more flexible scene understanding. Once such a 3D scene graph has been constructed, it can serve as a powerful substrate for complex semantic tasks and long-horizon planning, as demonstrated by systems such as SayPlan [19] and HOVSG [27].

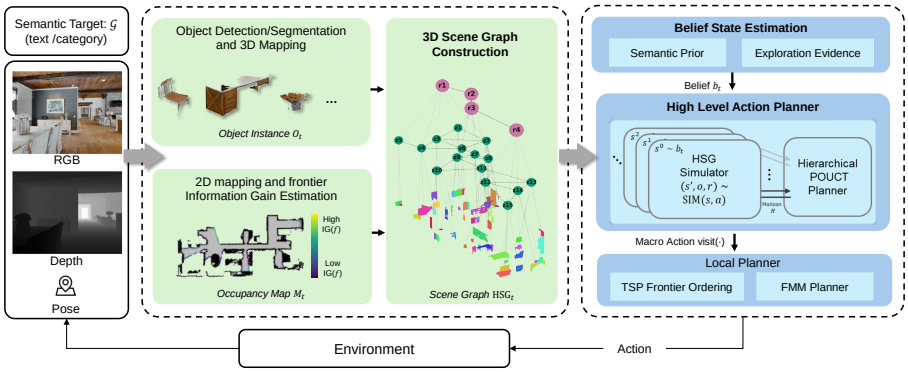


Fig. 2: Framework of our method. Given posed RGB-D observations, the perception module incrementally builds an HSG along with object instances and occupancy map, which the decision module uses to maintain a belief state b_t, select visit macro-actions via hierarchical POUCT, and generate executable actions with local planning.

Despite this progress, many existing hierarchical and open-vocabulary approaches focus on offline reconstruction of 3D scene graphs from complete point clouds [27] or RGB-D sequences [8]. While systems such as Kimera [20] and Hydra [9] support online construction, they primarily focus on SLAM and scene reconstruction rather than using the graph as a planning state for semantic navigation. This motivates our online hierarchical 3D scene graph, which is incrementally constructed from RGB-D observations and explicitly tailored to support semantic-navigation decision making.

Semantic Navigation. Semantic navigation requires robots to reason over *semantic goals* beyond geometric obstacle avoidance. Early Object Navigation (ObjectNav) trains policies at scale in simulation under a closed set of target categories [6], but such policies depend on task-specific data and reward design and often generalize poorly to novel environments and unseen categories [7]. With open-vocabulary vision–language representations (*e.g.*, CLIP) and large language / vision-language models (LLMs/VLMs), recent methods (*e.g.*, ESC [41], OpenFMNav [14], VLFM [34]) move toward zero-shot and open-vocabulary navigation, allowing agents to pursue arbitrary targets without retraining. In zero-shot ObjectNav, a common paradigm is to use an open-vocabulary visual encoder to map the target description and observations into a shared embedding space, and couple it with frontier-based exploration [37] or map-based search strategies [28] for navigation; furthermore, LLMs are often queried for commonsense priors over likely target locations [3, 15].

To provide structural context for these priors, recent methods like SG-Nav [32] and UniGoal [33] introduce 3D scene graphs in navigation tasks, using them as structured prompts for LLMs to score and select candidate frontiers. In contrast, we formulate the hierarchical scene graph as a compact state abstraction,

enabling us to maintain an online belief state and perform explicit long-horizon lookahead via simulator rollouts.

3 Problem Definition

We study the semantic navigation task, where a robot navigates in an unknown 3D indoor environment equipped with an egocentric RGB-D camera and odometry. At each time step, the robot observes the current RGB-D frame and its own pose, and selects an action from a discrete set {`move_forward`, `turn_left`, `turn_right`, `stop`}. An episode is considered successful if the robot reaches a target object $\mathcal{G}$ within at most T steps and issues `stop` when its distance to $\mathcal{G}$ falls within a fixed threshold. A *semantic target* refers to a textual description of the desired object, capturing its category, intrinsic attributes, and relations to nearby objects; when this description reduces to a category label only, the task degenerates to standard ObjectNav [6].

4 Incremental Hierarchical 3D Scene Graph Construction

In this section, we detail the incremental construction of the hierarchical 3D scene graph HSG_t from RGB-D observations. Specifically, while object nodes are actively maintained online, higher-level zone and region nodes are induced in a bottom-up manner leveraging GVD and spectral clustering. Furthermore, each node is enriched with a semantic description generated by a vision-language model (VLM). An overview of the complete pipeline is shown in Fig. 2.

4.1 Hierarchical 3D Scene Graph

We define the hierarchical 3D scene graph at time t as

$$\mathrm{HSG}_t = (V_t, E_t), \tag{1}$$

where

$$V_t = O_t \cup Z_t \cup R_t, \tag{2}$$

denotes the set of Object, Zone, and Region nodes at time t. Each node $v \in V_t$ carries a set of attributes, including its 3D point cloud, observation timestamps, and semantic description. The edge set is given by

$$E_t = E_t^{\mathrm{intra}} \cup E_t^{\mathrm{inter}}, \tag{3}$$

which captures two types of relations. Intra-layer spatial relations E_t^{intra} connect adjacent nodes within the same level, while inter-layer hierarchical relations E_t^{inter} encode cross-level containment through directed object-to-zone and zone-to-region edges. This hierarchical topology facilitates multi-scale planning across object, zone, and region levels.

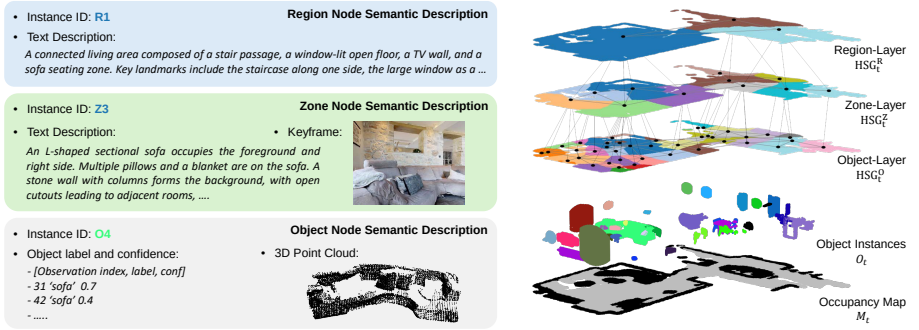


Fig. 3: Hierarchical 3D scene graph overview. Given the occupancy map M_t and object instances O_t, we construct a bottom-up spatial topology comprising object, zone, and region layers. A GVD partitions the map M_t into object-centric cells, which aggregate into larger-scale zone and region partitions based on their hierarchical dependencies.

4.2 Geometric Mapping and Object-Level Representation

At each time step t, the agent uses the current RGB-D observation and pose to incrementally update an occupancy map M_t, which represents the explored free space and obstacles. M_t is then used for generalized Voronoi partitioning and low-level action generation. At the same time, following [8], we apply an open-vocabulary object segmentation model [24] on the RGB image, project the resulting 2D object instances into 3D using depth and pose, and associate them with existing objects to maintain a set of object nodes

$$O_t = \{o_1, \ldots, o_{N_t}\}, \tag{4}$$

where each o_i denotes a 3D object instance in the scene.

4.3 Hierarchical Layer Construction

Based on the object instances O_t and the occupancy grid map M_t at time t, we first perform object-level Voronoi partitioning to obtain spatial adjacency among objects. We then apply spectral clustering on the object graph, combining geometric proximity and semantic similarity, to induce Zone and Region nodes in a bottom-up manner.

Object-level Voronoi Partition. To establish object adjacency relations, we construct a generalized Voronoi diagram [11] on the occupancy map. Specifically, we project the 3D point cloud of each object onto the grid map M_t and obtain a set of occupied cells, which serves as the object's 2D projection. Each explored grid cell is then assigned to its nearest object based on the shortest path distance to all object projections on the map. This partitions the explored map into non-overlapping, object-centric areas. Based on this partition, we build an object-level adjacency graph $\mathrm{HSG}_t^{\mathrm{O}}$, where each object is represented as a node. An

undirected edge is established between two nodes if their corresponding Voronoi regions share a boundary on the grid.

Spectral Clustering to Build Zone and Region Nodes. We leverage a spatial coherence prior that *nearby objects tend to form coherent higher-level semantic groups*. Based on the object adjacency graph $\mathrm{HSG}_t^{\mathrm{O}}$, we cluster objects into higher-level structures by jointly leveraging geometric and semantic cues. Specifically, we construct a weighted affinity matrix $W = [w_{ij}]$ over the adjacency edge set E_t^O, where each edge weight encodes spatial proximity and semantic similarity:

$$w_{ij} = \begin{cases} \exp\left(-\frac{d_{ij}^2}{\sigma_d^2}\right) \exp\left(\frac{\mathrm{sim}(e_i, e_j)}{\tau}\right), & (i,j) \in E_t^O \\ 0, & \text{otherwise} \end{cases}, \tag{5}$$

where d_{ij} denotes the spatial distance between objects o_i and o_j, and e_i represents the semantic embedding encoded from the semantic description of the node. $\mathrm{sim}(\cdot,\cdot)$ is cosine similarity, and σ_d, τ are hyper-parameters. This formulation restricts interactions to spatial neighbors while encouraging semantically coherent grouping.

To extract the hierarchical structure, the clustering is formulated as a graph cut problem, partitioning $\mathrm{HSG}_t^{\mathrm{O}}$ into subgraphs with *dense internal and sparse inter-cluster connections*. This partitioning is approximately solved via spectral clustering [43]. Objects sharing the same cluster label are merged to form a Zone node $z_j \in Z_t$, connected by a hierarchical edge (o_i, z_j). As shown in Fig. 3, the spatial extent of a Zone is derived from the union of its members' Voronoi areas.

Similarly, we apply the same clustering process to the zone layer $\mathrm{HSG}_t^{\mathrm{Z}}$ to construct the region layer $\mathrm{HSG}_t^{\mathrm{R}}$, which provides a coarser-grained semantic abstraction. The numbers of zones and regions are set according to the currently observed area and the number of observed objects.

Incremental Scene Graph Update. We adopt a fixed-step incremental update scheme and update the hierarchical scene graph every N steps. For each layer of HSG_t, we maintain a sparse adjacency list representation with edge weights. Within the window $[t-N, t]$, when objects are added or merged, edge weights are recomputed only for the affected objects and their local neighborhoods. Afterward, bottom-up re-clustering is performed on the updated object graph to update the layers.

4.4 Semantic Description of Scene Graph Nodes

We maintain a semantic description for each node in HSG_t to derive the semantic prior in Sec. 5. For each object node o_i, predicted labels and confidence scores from the open-vocabulary segmentation model are accumulated to obtain stable object semantics, following [37]. For each zone node, a representative keyframe is selected from past RGB observations by maximizing its coverage of the zone area, and a VLM is used to generate a caption as the zone description. For each

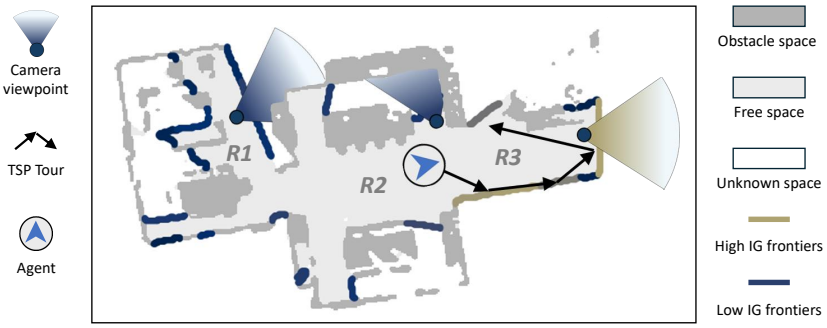


Fig. 4: Exploration evidence and local planning. We compute frontier information gains within partitioned regions and aggregate them as exploration evidence to update the HSG belief state. Upon selecting a macro-action, the local planner generates an efficient exploration tour by modeling the visitation order of high-gain frontiers as a traveling salesman problem.

region node, descriptions of its member zones are aggregated and summarized by an LLM to produce a coarser region semantic representation. Furthermore, these semantic descriptions are utilized to determine the `stop` action. While ObjectNav relies on the object detection results, TextInstanceNav prompts a VLM with the current semantic description and observation images.

5 Hierarchical Semantic Navigation Policy

In this section, we generate the agent's actions from HSG_t and the occupancy map M_t (Sec. 4), using a hierarchical policy with high-level decision making and low-level execution. At each high-level step, we maintain a belief state b_t on HSG_t and instantiate a scene-graph-based simulator SIM. Before executing any real actions, SIM performs forward simulations to estimate the long-horizon return of candidate macro-actions, enabling online selection of the next region or zone to visit. The local planner then converts the selected macro-action into a feasible sequence of frontier visits and discrete actions.

5.1 Belief State Estimation

We estimate a hierarchical belief on HSG_t to model the target existence probability across regions and zones. To avoid overconfidence in explored areas, we introduce virtual nodes r_u and z_u to represent unexplored regions and zones.
Semantic Prior on the HSG. We query an LLM to predict a semantic prior $p_t^R(r)$ for each region r, reflecting the semantic plausibility that the target appears in that region. For the unknown region r_u, we up-weight its prior when all explored regions receive low scores, and down-weight it otherwise. When the agent enters a region r for refinement, we build local semantic descriptions for each $z \in \mathcal{Z}_t(r)$ and obtain the zone-level semantic prior $p_t^Z(z \mid r)$.

Algorithm 1: Hierarchical Scene-Graph Belief Planning

Input: Scene graph belief b_t (Sec. 5.1), Simulator SIM (Sec. 5.2), Horizon H
Output: High-level action a_t

$r^* \leftarrow$ `POUCT-Select`$(\text{SIM}, b_t^R, \mathcal{A}_t^R, H)$. (Sec. 5.3)
if $r^* = r_u$ **then**
  return `visit`(r_u) `// global search`
$z^* \leftarrow$ `POUCT-Select`$(\text{SIM}, b_t^Z(\cdot \mid r^*), \mathcal{A}_t^Z, H)$. (Sec. 5.3)
if $z^* = z_u$ **then**
  return `visit`(r^*) `// region-level search`
else
  return `visit`(z^*) `// zone-level search`

Observation Feedback as Exploration Evidence. We extract frontiers from the occupancy map and compute the information gain $\text{IG}(f)$ for each frontier f, which describes its potential to expand the observable free space [10], as shown in Fig. 4. For region r, exploration evidence is defined as

$$e_t^R(r) = \frac{\sum_{f \in \mathcal{F}(r) \,:\, \text{IG}(f) > \tau_{\mathcal{G}}} \text{IG}(f)}{\sum_{f \in \mathcal{F}(r)} \text{IG}(f) + \varepsilon}, \tag{6}$$

where $\mathcal{F}(r)$ denotes the set of frontiers in region r. This score measures the exploration potential of r. If high IG frontiers dominate, $e_t^R(r) \to 1$; otherwise, $e_t^R(r) \to 0$. We set $\tau_{\mathcal{G}}$ by target scale (smaller for small targets, *e.g.*, plant; larger for large targets, *e.g.*, bed), and use ε for numerical stability.

Belief Construction. At most steps, the agent does not find the target and thus receives a `not_found` feedback. So we avoid an explicit Bayesian belief update. Instead, we treat this negative signal as a decay of the exploration evidence $e_t^R(r)$ and re-estimate the belief at each high-level step. The region level belief is

$$b_t^R(r) = \text{Norm}\left(p_t^R(r) \cdot e_t^R(r)\right). \tag{7}$$

Incorporating observation feedback helps mitigate semantic bias from the LLM. The zone-level belief $b_t^Z(z \mid r^*)$ is constructed analogously.

5.2 HSG Simulator Construction

We build a scene-graph-based simulator SIM from the HSG_t to model state transitions and observation feedback induced by the macro-actions, and use it in Sec. 5.3 to estimate their long-term returns. Formally,

$$(s', o, u) \sim \text{SIM}(s, a; \text{HSG}_t). \tag{8}$$

The state s includes the HSG_t, the agent's current node, and a hypothesis of the target node sampled from the current belief b_t. A macro-action a corresponds

to visiting a region or a zone. To keep planning computationally tractable, the topology of HSG_t is held fixed during each POUCT search. Executing action a transitions the agent to the selected node and yields a binary observation $o \in \{\texttt{found}, \texttt{not\_found}\}$. If $o = \texttt{not\_found}$, the agent updates its position to explore other nodes in subsequent simulation steps. The immediate reward u is

$$u(o, s, a) = R(o) - \lambda\, c_{\mathrm{move}}(s, a), \qquad R(o) = \begin{cases} R_{\mathrm{succ}}, & o = \texttt{found} \\ 0, & o = \texttt{not\_found} \end{cases}, \tag{9}$$

where $c_{\mathrm{move}}(s, a)$ is the shortest-path length on HSG_t from the agent's current node to the action's goal node, and λ weights the movement cost.

5.3 Online Planning with the HSG Simulator

At each high-level decision step t, we first select a macro-action a_t over the region belief b_t^R from the region-level action set $\mathcal{A}_t^R$:

$$a_t = \arg\max_{a \in \mathcal{A}_t^R} Q(b_t^R, a), \qquad \mathcal{A}_t^R = \{\texttt{visit}(r) \mid r \in \mathcal{R}_t\}. \tag{10}$$

To estimate $Q(b_t^R, a)$ online, we use POUCT to perform Monte Carlo Tree Search in the current region belief b_t^R (see [22] for details on POUCT). Specifically, the i-th simulation starts by sampling an initial state instance

$$s^i \sim b_t^R, \tag{11}$$

where s^i denotes the state drawn from the belief, specifying the agent's current node and a target position. We then perform forward simulations to obtain observations o and immediate rewards u while propagating the state, and back up the accumulated returns to update action-value estimates in the search tree, until reaching the planning horizon H or terminating when `found` is observed.

Within the search tree, POUCT uses the standard UCB rule to balance exploration and exploitation [22]. At leaf nodes, it applies a random policy to extend the trajectory and estimate returns. By repeatedly sampling $s \sim b_t^R$ and backing up accumulated returns over simulated trajectories, POUCT estimates the expected action value as

$$Q(b_t^R, a) \approx \mathbb{E}_{s \sim b_t^R}\Big[\sum_{k=0}^{H-1} \gamma^k u_{t+k} \;\Big|\; a_t = a \Big], \tag{12}$$

where the planning horizon H controls the look-ahead depth. A larger H accumulates multi-step movement costs and better captures the long-term return of an action. However, an overly large H increases model bias, since HSG_t is an approximation of the true state and long-horizon simulation accumulates errors, leading to value-estimation drift. Thus, H trades off long-term return evaluation against accumulated model bias.

Table 1: State-of-the-art comparison on the ObjectNav task across four benchmark datasets. "TF" indicates training-free methods. Best results are in bold.

Method	TF	MP3D		HM3Dv1		HM3Dv2		HSSD	
		SR ↑	SPL ↑	SR ↑	SPL ↑	SR ↑	SPL ↑	SR ↑	SPL ↑
ZSON [16]	✗	15.3	4.8	25.5	12.6	-	-	-	-
Uni-NaVid [36]	✗	-	-	73.7	37.1	-	-	-	-
VLingNav [25]	✗	58.9	26.5	79.1	42.9	83.0	40.5	-	-
ESC [41]	✓	28.7	14.2	39.2	22.3	-	-	38.1	22.2
L3MVN [35]	✓	34.9	14.5	50.4	23.1	36.3	15.7	41.2	22.5
VoroNav [28]	✓	-	-	42.0	26.0	-	-	41.0	23.2
VLFM [34]	✓	36.4	17.5	52.5	30.4	63.6	32.5	-	-
SG-Nav [32]	✓	40.2	16.0	54.0	24.9	49.6	25.5	-	-
ImagineNav [39]	✓	-	-	53.0	23.8	-	-	51.0	24.9
UniGoal [33]	✓	41.0	16.4	54.5	25.1	-	-	-	-
BeliefMapNav [42]	✓	37.3	17.6	61.4	30.6	-	-	65.2	32.1
ImagineNav++ [26]	✓	-	-	58.5	26.6	-	-	64.5	27.9
FBN-Nav [38]	✓	42.1	18.1	58.8	31.2	-	-	-	-
ApexNav [37]	✓	39.2	17.8	59.6	33.0	76.2	38.0	61.1	31.5
Ours	✓	**45.9**	**19.6**	**61.6**	**33.9**	**80.1**	**39.5**	**69.9**	**36.4**

After obtaining the region-level selection r^*, we refine the decision hierarchically. If $r^* = r_u$, we execute a global frontier tour for coverage. Otherwise, we construct a zone-level belief $b_t^Z(\cdot \mid r^*)$ on the subgraph of r^* and run POUCT again over the corresponding zone-level action set

$$\mathcal{A}_t^Z(r^*) = \{\texttt{visit}(z) \mid z \in \mathcal{Z}_t(r^*)\}, \tag{13}$$

to select the zone to prioritize within the region, as shown in Alg. 1.

5.4 Local Planner

Given a macro-action $\texttt{visit}(\cdot)$, we collect the set of frontiers associated with the selected region or zone on the occupancy map M_t. Following FUEL [40], we cast the ordering of these frontier visits as an asymmetric traveling salesman problem (ATSP) and solve for a minimum-cost tour to reduce local travel distance. Conditioned on the resulting frontier order, we plan shortest paths segment-by-segment on M_t using Fast Marching Method [21], and discretize the concatenated path into a sequence of executable low-level actions.

6 Experiment

In this section, we present extensive experiments to evaluate our method. We first introduce the experimental setup, then compare our approach with state-

Table 2: Results on Text-instance navigation.

Method	TF	SR ↑	SPL ↑
PSL [23]	✗	16.5	7.5
GOAT [5]	✗	17.0	8.8
UniGoal [33]	✓	20.2	11.4
Ours	✓	**38.1**	**16.0**

Table 3: Ablations on HM3Dv2 ObjectNav.

Method	SR ↑	SPL ↑
w/o Evidence	76.0	37.4
w/o Rollouts (H=0)	76.7	35.9
w/o Zone-level	79.5	38.2
w/ Qwen3-VL-4B	77.2	37.9
Ours	**80.1**	**39.5**

of-the-art baselines and perform ablations on its key components. Finally, we provide qualitative results to further illustrate the benefits of our design.

6.1 Experiment Setup

Dataset: We evaluate our approach on two navigation tasks. For Object Navigation (ObjectNav), we follow SemExp [6] and conduct experiments on Matterport3D (MP3D [4]), HM3D-Semantics-v0.1 (HM3Dv1 [18]), HM3D-Semantics-v0.2 (HM3Dv2 [30]) and Habitat Synthetic Scenes Dataset (HSSD [12]). For Text-Instance Navigation, we use the dataset released in InstanceNav [23] and follow its original experimental settings.

Evaluation Metrics: We adopt *Success Rate* (SR) and *Success weighted by Path Length* (SPL) as the evaluation metrics [1]. SR measures the fraction of episodes in which the agent successfully finds the target. SPL measures path efficiency relative to the shortest path, and is set to 0 for failed episodes.

Implementation Details: We implement our agent in Habitat-Sim [17]. We use Qwen3-VL-8B [31] as both the vision-language model and the language model to generate semantic descriptions of the scene graph and to estimate semantic priors over target objects. For open-vocabulary object detection and segmentation, we adopt YOLOE [24]. All simulation experiments are run on a workstation with an NVIDIA GeForce RTX 4090 GPU and an Intel Core i9-12900K CPU. The map and HSG are updated online, although VLM/LLM inference prevents real-time execution; detailed runtime analysis is provided in the supplementary material.

Baselines: We evaluate our method on ObjectNav and TextInstanceNav against both training-based (as supervised references) and training-free (our primary focus) baselines. While leading training-free methods like VLFM [34], UniGoal [33] and ApexNav [37] leverage foundation model priors, they lack global multi-granularity memory and explicit long-horizon planning, typically defaulting to local greedy frontier search. Additionally, we select ApexNav as a strong training-free baseline. It adaptively switches between semantic-prior guidance and geometry-driven coverage exploration, and serves as a competitive frontier-based reference. We further compare performance across different target distances.

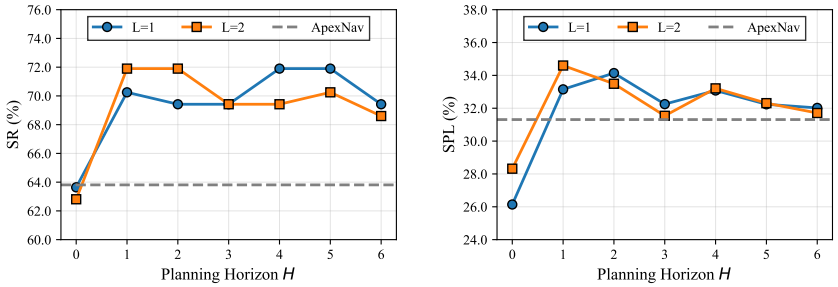


Fig. 5: Level and planning horizon ablation. We evaluate the performance on long-range episodes of the HM3Dv2 ObjectNav dataset.

6.2 Comparison with State-of-the-Art

We evaluate our approach against state-of-the-art baselines on ObjectNav and TextInstanceNav. On ObjectNav (Tab. 1), our method improves average SR and SPL by 5.4% and 2.3% compared to the strong baseline ApexNav across four datasets. Gains are even more pronounced on TextInstanceNav (Tab. 2), where we outperform the current state-of-the-art method by 17.9% in SR and 4.6% in SPL. We attribute this to the rich target attributes and spatial relations in TextInstanceNav queries, which fully exploit our HSG for multi-granularity semantic modeling and explicit long-horizon planning capabilities.

6.3 Long-Distance Navigation Analysis

To validate our contributions to long-horizon decision-making and spatial memory, we specifically evaluate our method on long-distance scenarios. We filter a subset of episodes where the initial shortest-path distance exceeds 10 meters, comparing our approach against ApexNav on ObjectNav and UniGoal on TextInstanceNav. As illustrated in Fig. 1b, our advantages are even more pronounced on these long-distance episodes compared to the full evaluation set, yielding average improvements of 9.4% in SR and 5.0% in SPL. These results demonstrate that our performance gains primarily stem from more effective handling of long-horizon navigation. By leveraging explicit hierarchical memory and evaluating long-term expected returns, the agent significantly mitigates meaningless exploration and redundant backtracking, thereby substantially improving path efficiency. More detailed results are provided in the supplementary material.

6.4 Ablation Study

Component Ablation. In Table 3, we remove exploration evidence, the zone-level hierarchy, long-horizon rollouts, and replace the VLM with Qwen3-VL-4B to isolate the impact of each component.

(i) w/o Evidence. Without exploration evidence, belief estimation relies solely on the LLM prior. Lacking online observation feedback to correct biases, the

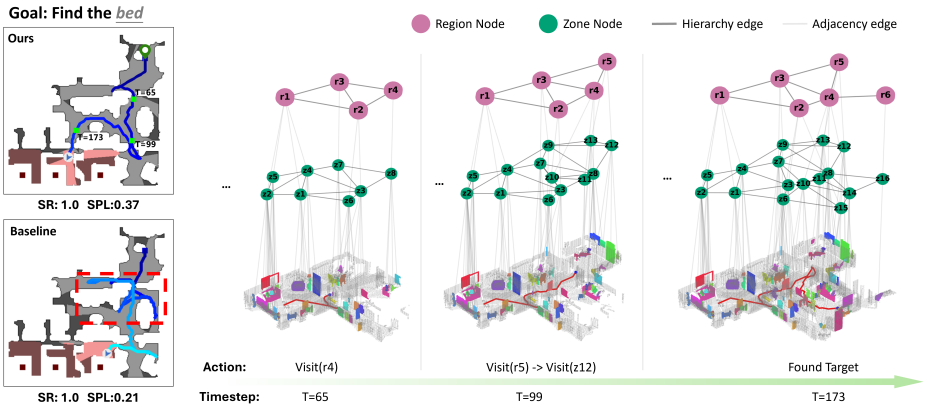


Fig. 6: Visualization of the navigation process in a long-range episode. Left: top-down trajectories of our method and the baseline (ApexNav [37]) with SR/SPL; right: evolution of the HSG at three key timesteps, with the selected visit macro-actions and timesteps annotated.

agent tends to over-search high-prior areas when the prior misaligns with the current scene. (ii) w/o Zone-level. Simplifying the hierarchy to a coarse Object-Region structure restricts the planner's ability to leverage fine-grained spatial differences to filter the candidate space. (iii) w/o Rollouts. Without explicitly evaluating long-term returns ($H = 0$), the agent's greedy decisions lead to myopic exploration. (iv) w/ Qwen3-VL-4B. Using a smaller VLM reduces the reliability of semantic descriptions and priors, which weakens the discriminative power of the belief state and subsequently degrades planning quality.

Level–Horizon Ablation. To investigate the interplay between hierarchy level and planning horizon, we evaluate combinations of level $L \in \{1, 2\}$ and horizon $H \in \{0, 1, 2, 3, 4, 5\}$ on long-range episodes of the HM3Dv2 ObjectNav dataset. $L = 1$ and $L = 2$ denote the Object–Region and Object–Zone–Region hierarchies, respectively. H denotes the planning horizon, i.e., the number of forward-simulation steps executed by the HSG simulator during online planning (Eq. 12). Results are shown in Fig. 5.

The planning horizon H admits an optimal value, and large H can degrade performance. Increasing H allows the planner to explicitly evaluate the long-term benefit of information gathering, improving long-range decision making. However, as H grows, the mismatch between the HSG-based simulator and the real environment accumulates over multi-step simulation, making action-value estimates less reliable and leading to performance degradation.

The optimal horizon depends on the hierarchy level. Since $L = 2$ induces a more detailed HSG-based simulator, the multi-step simulation gap accumulates faster as H increases, making longer planning horizons less reliable. As a result, $L = 2$ prefers a shorter horizon than $L = 1$.

6.5 Qualitative Analysis

Fig. 6 shows a representative long-range ObjectNav episode that illustrates our decision-making behavior. The baseline (ApexNav [37]) exhibits clear revisits in the mid-exploration stage (highlighted by the dashed red box), indicating redundant backtracking. In contrast, our policy plans at the zone/region level and evaluates macro-actions with long-term returns, avoiding redundant revisits.

7 Conclusion

In this paper, we propose an online hierarchical 3D scene graph construction and belief-based planning framework to address the core challenges of inadequate long-horizon decision-making and spatial memory in zero-shot semantic navigation. Operating in unseen environments, we integrate GVD with spectral clustering to incrementally construct a bottom-up, multi-granularity Object-Zone-Region semantic topology. This provides a structured state representation for global planning. Furthermore, we fuse LLM semantic priors with exploration evidence to maintain hierarchical belief states. Through finite-horizon lookahead rollouts within the HSG-based simulator, the agent explicitly evaluates long-term expected returns of macro-actions. Extensive experiments show our approach outperforms SOTA baselines, especially in long-distance navigation tasks.
Limitations and Future Work. Our evaluation is limited to simulation and assumes posed RGB-D observations. In real-world deployment, localization errors may affect object-instance fusion and occupancy mapping. Moreover, although the system operates online, VLM/LLM inference introduces substantial latency and prevents real-time execution. Future work will focus on real-robot evaluation, integration with semantic SLAM, and more efficient semantic inference.

Acknowledgements

This research is supported by Hong Kong Research Grants Council under GRF-15229423 and GRF-15221625.

References

1. Anderson, P., Chang, A., Chaplot, D.S., Dosovitskiy, A., Gupta, S., Koltun, V., Kosecka, J., Malik, J., Mottaghi, R., Savva, M., et al.: On evaluation of embodied navigation agents. arXiv preprint arXiv:1807.06757 (2018)
2. Armeni, I., He, Z.Y., Gwak, J., Zamir, A.R., Fischer, M., Malik, J., Savarese, S.: 3D scene graph: A structure for unified semantics, 3D space, and camera. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5664–5673 (2019)
3. Cao, Y., Zhang, J., Yu, Z., Liu, S., Qin, Z., Zou, Q., Du, B., Xu, K.: CogNav: Cognitive process modeling for object goal navigation with LLMs. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9550–9560 (2025)

4. Chang, A., Dai, A., Funkhouser, T., Halber, M., Nießner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3D: Learning from RGB-D data in indoor environments. In: 2017 International Conference on 3D Vision (3DV). pp. 667–676. IEEE Computer Society (2017)
5. Chang, M., Gervet, T., Khanna, M., Yenamandra, S., Shah, D., Min, S.Y., Shah, K., Paxton, C., Gupta, S., Batra, D., et al.: GOAT: Go to any thing. arXiv preprint arXiv:2311.06430 (2023)
6. Chaplot, D.S., Gandhi, D.P., Gupta, A., Salakhutdinov, R.R.: Object goal navigation using goal-oriented semantic exploration. In: Advances in Neural Information Processing Systems. vol. 33, pp. 4247–4258 (2020)
7. Gadre, S.Y., Wortsman, M., Ilharco, G., Schmidt, L., Song, S.: 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. pp. 23171–23181 (2023)
8. Gu, Q., Kuwajerwala, A., Morin, S., Jatavallabhula, K.M., Sen, B., Agarwal, A., Rivera, C., Paul, W., Ellis, K., Chellappa, R., et al.: ConceptGraphs: Open-vocabulary 3D scene graphs for perception and planning. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 5021–5028. IEEE (2024)
9. Hughes, N., Chang, Y., Hu, S., Talak, R., Abdulhai, R., Strader, J., Carlone, L.: Foundations of spatial perception for robotics: Hierarchical representations and real-time systems. The International Journal of Robotics Research **43**(10), 1457–1505 (2024)
10. Isler, S., Sabzevari, R., Delmerico, J., Scaramuzza, D.: An information gain formulation for active volumetric 3D reconstruction. In: 2016 IEEE international conference on robotics and automation (ICRA). pp. 3477–3484. IEEE (2016)
11. Kalra, N., Ferguson, D., Stentz, A.: Incremental reconstruction of generalized Voronoi diagrams on grids. Robotics and Autonomous Systems **57**(2), 123–128 (2009)
12. Khanna, M., Mao, Y., Jiang, H., Haresh, S., Shacklett, B., Batra, D., Clegg, A., Undersander, E., Chang, A.X., Savva, M.: Habitat synthetic scenes dataset (HSSD-200): An analysis of 3D scene scale and realism tradeoffs for ObjectGoal navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16384–16393 (2024)
13. Koch, S., Vaskevicius, N., Colosi, M., Hermosilla, P., Ropinski, T.: Open3DSG: Open-vocabulary 3D scene graphs from point clouds with queryable objects and open-set relationships. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14183–14193 (2024)
14. Kuang, Y., Lin, H., Jiang, M.: OpenFMNav: Towards open-set zero-shot object navigation via vision-language foundation models. arXiv preprint arXiv:2402.10670 (2024)
15. Long, Y., Cai, W., Wang, H., Zhan, G., Dong, H.: InstructNav: Zero-shot system for generic instruction navigation in unexplored environment. In: Conference on Robot Learning. pp. 2049–2060. PMLR (2025)
16. Majumdar, A., Aggarwal, G., Devnani, B., Hoffman, J., Batra, D.: ZSON: Zero-shot object-goal navigation using multimodal goal embeddings. In: Advances in Neural Information Processing Systems. vol. 35, pp. 32340–32352 (2022)
17. Puig, X., Undersander, E., Szot, A., Cote, M.D., Yang, T.Y., Partsey, R., Desai, R., Clegg, A.W., Hlavac, M., Min, S.Y., et al.: Habitat 3.0: A co-habitat for humans, avatars and robots. arXiv preprint arXiv:2310.13724 (2023)

18. Ramakrishnan, S.K., Gokaslan, A., Wijmans, E., Maksymets, O., Clegg, A., Turner, J., Undersander, E., Galuba, W., Westbury, A., Chang, A.X., et al.: Habitat-Matterport 3D dataset (HM3D): 1000 large-scale 3D environments for embodied AI. arXiv preprint arXiv:2109.08238 (2021)
19. Rana, K., Haviland, J., Garg, S., Abou-Chakra, J., Reid, I., Suenderhauf, N.: SayPlan: Grounding large language models using 3D scene graphs for scalable robot task planning. In: Conference on Robot Learning. pp. 23–72. PMLR (2023)
20. Rosinol, A., Violette, A., Abate, M., Hughes, N., Chang, Y., Shi, J., Gupta, A., Carlone, L.: Kimera: From SLAM to spatial perception with 3D dynamic scene graphs. The International Journal of Robotics Research **40**(12–14), 1510–1546 (2021)
21. Sethian, J.A.: A fast marching level set method for monotonically advancing fronts. proceedings of the National Academy of Sciences **93**(4), 1591–1595 (1996)
22. Silver, D., Veness, J.: Monte-carlo planning in large POMDPs. In: Advances in neural information processing systems. vol. 23 (2010)
23. Sun, X., Liu, L., Zhi, H., Qiu, R., Liang, J.: Prioritized semantic learning for zero-shot instance navigation. In: European Conference on Computer Vision. pp. 161–178. Springer (2024)
24. Wang, A., Liu, L., Chen, H., Lin, Z., Han, J., Ding, G.: YOLOE: Real-time seeing anything. arXiv preprint arXiv:2503.07465 (2025)
25. Wang, S., Luo, Y., Chen, X., Luo, A., Li, D., Liu, C., Chen, S., Zhang, Y., Yu, J.: VLingNav: Embodied navigation with adaptive reasoning and visual-assisted linguistic memory. arXiv preprint arXiv:2601.08665 (2026)
26. Wang, T., Zhao, X., Cai, W., Sun, C.: ImagineNav++: Prompting vision-language models as embodied navigator through scene imagination. arXiv preprint arXiv:2512.17435 (2025)
27. Werby, A., Huang, C., Büchner, M., Valada, A., Burgard, W.: Hierarchical Open-Vocabulary 3D Scene Graphs for Language-Grounded Robot Navigation. In: Proceedings of Robotics: Science and Systems. Delft, Netherlands (July 2024)
28. Wu, P., Mu, Y., Wu, B., Hou, Y., Ma, J., Zhang, S., Liu, C.: VoroNav: Voronoi-based zero-shot object navigation with large language model. In: International Conference on Machine Learning. pp. 53757–53775. PMLR (2024)
29. Xie, J., Zhang, K., Chen, J., Zhu, T., Lou, R., Tian, Y., Xiao, Y., Su, Y.: TravelPlanner: A benchmark for real-world planning with language agents. In: Proceedings of the 41st International Conference on Machine Learning. pp. 54590–54613 (2024)
30. Yadav, K., Ramrakhya, R., Ramakrishnan, S.K., Gervet, T., Turner, J., Gokaslan, A., Maestre, N., Chang, A.X., Batra, D., Savva, M., et al.: Habitat-Matterport 3D semantics dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4927–4936 (2023)
31. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025)
32. Yin, H., Xu, X., Wu, Z., Zhou, J., Lu, J.: SG-Nav: Online 3D scene graph prompting for LLM-based zero-shot object navigation. In: Advances in neural information processing systems. vol. 37, pp. 5285–5307 (2024)
33. Yin, H., Xu, X., Zhao, L., Wang, Z., Zhou, J., Lu, J.: UniGoal: Towards universal zero-shot goal-oriented navigation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 19057–19066 (2025)
34. Yokoyama, N., Ha, S., Batra, D., Wang, J., Bucher, B.: VLFM: Vision-language frontier maps for zero-shot semantic navigation. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 42–48. IEEE (2024)

35. Yu, B., Kasaei, H., Cao, M.: L3MVN: Leveraging large language models for visual target navigation. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3554–3560. IEEE (2023)
36. Zhang, J., Wang, K., Wang, S., Li, M., Liu, H., Wei, S., Wang, Z., Zhang, Z., Wang, H.: Uni-NaVid: A video-based Vision-Language-Action model for unifying embodied navigation tasks. arXiv preprint arXiv:2412.06224 (2024)
37. Zhang, M., Du, Y., Wu, C., Zhou, J., Qi, Z., Ma, J., Zhou, B.: ApexNav: An adaptive exploration strategy for zero-shot object navigation with target-centric semantic fusion. IEEE Robotics and Automation Letters (2025)
38. Zhang, S., Yu, X., Song, X., Wang, Y., Jiang, S.: Function-centric bayesian network for zero-shot object goal navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19535–19545 (2025)
39. Zhao, X., Cai, W., Tang, L., Wang, T.: ImagineNav: Prompting vision-language models as embodied navigator through scene imagination. arXiv preprint arXiv:2410.09874 (2024)
40. Zhou, B., Zhang, Y., Chen, X., Shen, S.: FUEL: Fast UAV exploration using incremental frontier structure and hierarchical planning. IEEE Robotics and Automation Letters **6**(2), 779–786 (2021)
41. Zhou, K., Zheng, K., Pryor, C., Shen, Y., Jin, H., Getoor, L., Wang, X.E.: ESC: Exploration with soft commonsense constraints for zero-shot object navigation. In: International Conference on Machine Learning. pp. 42829–42842. PMLR (2023)
42. Zhou, Z., Hu, Y., Zhang, L., Li, Z., Chen, S.: BeliefMapNav: 3D voxel-based belief map for zero-shot object navigation. arXiv preprint arXiv:2506.06487 (2025)
43. Zivkovic, Z., Bakker, B., Krose, B.: Hierarchical map building and planning based on graph partitioning. In: 2006 IEEE International Conference on Robotics and Automation (ICRA). pp. 803–809 (2006)