Article | Received 30 June 2026; Revised 9 October 2026; Accepted 29 October 2026; Published xx November 2026
https://doi.org/10.55092/rl20260028

HospNav: failure-aware open-vocabulary hospital service robot navigation in annotated maps

Weifan Zhang, Yunxi Tang, Yunhui Liu and Xiangyu Chu*

Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong SAR, China

* Correspondence author; E-mail: xiangyuchu@cuhk.edu.hk.

Highlights:

- We formulate open-vocabulary navigation on annotated hospital semantic maps with region, object, and explicit absent-target outcomes.
- We introduce a training-free framework that performs no task-specific training or fine-tuning, uses pretrained foundation models, and couples a one-shot semantic prior, persistent grounding, and distance-aware region visitation.
- We formulate prior-aware, policy-relative abstention and evaluate its stopping conditions and search-effort trade-offs through gate-subset replay.

Abstract: Hospital service robots can often access floor plans, room labels, and wayfinding semantics before deployment, changing navigation from unconstrained exploration into structured target search. However, the challenge is to ground open-vocabulary instructions across symbolic map annotations and visual observations, while deciding when a bounded search justifies abstention rather than continued exploration. We present HospNav, a training-free framework for open-vocabulary navigation on annotated semantic maps. HospNav couples a one-shot large language model (LLM) prior over functional regions with distance-aware visitation, multi-source grounding, persistent object memory, and temporal confirmation. Its central contribution is a prior-aware bounded stopping policy that turns non-arrival into an explicit, auditable, policy-relative `FAILED` action. On the evaluated traces, stopping with a coverage gate yields higher present-target success than stopping on no-match evidence alone. Across 378 episodes in one synthetic hospital, the LLM-conditioned policy obtained the highest object-navigation point estimates among the internal priors, with a success rate (SR) of 57% and success weighted by path length (SPL) of 0.367 against 40%/0.177 for CLIP and 30%/0.160 for Uniform priors, and returned policy-relative `FAILED` on 67% of absent targets. Counterfactual replay on equal-budget traces reproduced every live stopping decision. These results show how lightweight semantic-map annotations can support failure-aware open-vocabulary hospital navigation through structured target search and explicit, auditable abstention.

Copyright©2026 by the authors. Published by ELSP. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium provided the original work is properly cited.

Keywords: hospital service robot; vision-language navigation; open-vocabulary navigation; semantic map; multi-source grounding

1. Introduction

Hospital service robots are increasingly considered for delivery, guidance, and equipment transport in large medical facilities [1–3]. Unlike generic indoor exploration settings, hospitals often provide substantial spatial semantics before deployment through floor plans, room labels, signage systems, and wayfinding databases. Using these resources, the robot searches for language-specified targets within a known, semantically annotated map rather than exploring an unknown environment. This setup is considered as known-map open-vocabulary navigation, in which an instruction may point to a labeled region, an object category grounded through observations, or an absent target for which no admissible grounding exists.

Vision-Language Navigation (VLN), Object Navigation (ObjectNav), and recent open-vocabulary navigation methods have made substantial progress in grounding natural-language goals in visual environments [4–8]. However, many of these methods initially assume that the environment is unknown and rely on online exploration to construct task-specific scene representations. This assumption is out of step with the reality of hospital service scenarios, as semantic map information is typically available in hospital settings, and search behavior should remain interpretable and bounded within access-sensitive spaces. Recent semantic search systems have shown that spatial priors, semantic maps, and the common sense of a large language model (LLM) can improve the performance of object-goal search [9,10]. However, the characterization of known-map open-vocabulary navigation with explicit "target missing" decisions remains insufficient.

Therefore, known-map open-vocabulary navigation is not solely a goal-reaching problem with a language input. The instruction must be grounded against heterogeneous evidence from symbolic region annotations and visual object observations accumulated during search. Hospital functions impose regularities on object placement, so region semantics provide a prior over where to acquire object evidence. The known map also makes bounded non-arrival a meaningful terminal condition. Once the regions selected by a frozen search policy have been covered without an admissible grounding, the robot may return `FAILED` instead of continuing until timeout or attributing the outcome to a navigation-stack failure. This perspective integrates semantic priors, multi-source grounding, and bounded stopping into a unified known-map search formulation.

This paper presents HospNav, a training-free framework for known-map open-vocabulary hospital navigation. The framework operates on a pre-annotated semantic map containing an occupancy grid and manually annotated regions. HospNav represents hospital common sense through a one-shot LLM prior for annotated regions and combines this prior with distance-aware visitation to enable semantic search. A multi-source grounding interface evaluates annotated regions and observed object instances based on acceptance thresholds specific to each source. When no admissible target is confirmed, a prior-aware stopping rule determines whether the system returns the policy-relative `FAILED` decision. Section 5.5 evaluates the impact of its stopping criteria on premature abstention and search effort.

The innovation of this method lies in the decision layer. HospNav makes bounded non-arrival a first-class action and specifies the evidence condition under which the action is taken. The implementation makes that decision observable and auditable within a complete ROS2/Nav2 stack. This study presents four major contributions.

(1) We model open-vocabulary hospital navigation on a known map as a semantic search, distinguishing between three types of results—`REACHED`, policy-relative `FAILED`, and `TIMEOUT` —for queries regarding regions, objects, and missing targets.
(2) We propose HospNav, a training-free framework that combines a one-shot LLM region prior, distance-aware visitation, multi-source grounding, persistent object memory, and temporal target confirmation.
(3) We propose a bounded abstention policy based on prior knowledge and analyze the impact of its termination conditions through gate-subset replay.
(4) We evaluated HospNav through internal and external baselines, statistical inference that accounts for cluster structure, and fixed-trace stopping ablations.

The remainder of this paper is structured as follows. Related work is reviewed in Section 2. The problem statement is formally defined in Section 3. For details on HospNav and its module algorithms, see Section 4. The experimental setup and results are presented in Section 5. Limitations and deployment considerations are discussed in Section 6. Concluding remarks appear in Section 7.

2. Related work

2.1. Vision-language navigation

VLN investigates how embodied agents interpret natural language instructions and take action in a visually grounded environment. The Room-to-Room (R2R) benchmark formalizes instruction following as navigation along photorealistic indoor trajectories [4]. Subsequent setups extend VLN to continuous control, remote object grounding, and multilingual instruction grounding [5, 6, 11]. These benchmarks define "language-conditioned navigation" as the challenge of coordinating the semantics of instructions, visual observations, and embodied movement. ObjectNav and open-vocabulary navigation require agents to locate targets of a specified category or described in natural language within unfamiliar environments, thereby reducing reliance on route-based instructions. HM3D-OVON provides a large-scale Habitat benchmark for open-vocabulary object-goal navigation on photorealistic scans and hundreds of object categories [8]. Recent systems have utilized vision-language models (VLMs), LLMs, and foundation model features to enhance zero-shot semantic navigation and target object search capabilities [12–16]. CoW combines RGB-D mapping, frontier-based exploration (FBE), and open-vocabulary object localization, and does not use pre-annotated semantic maps [7]. In contrast, NavA^3 handles high-level, long-horizon instructions through the Reasoning-VLM global policy and the learned NaviAfford pointing model for local object localization [17]. Like HospNav, it employs a "coarse-to-fine" decomposition method and utilizes pre-built scene representations. NavA^3 reasons over an annotated global 3D view and uses an unannotated local map in conjunction with NaviAfford for waypoint selection and target localization. HospNav, on the other hand, uses annotated semantic regions

and fixed anchors, and explicitly investigates policy-relative abstention behavior. MAG-Nav further addresses complex language descriptions by performing active grounding and memory replay in unknown scenes [18]. Despite these advances, most methods still focus on online exploration, reaching target objects, or trajectory following. In contrast, open-vocabulary search on known maps requires the agent to utilize existing functional region labels and distinguish between arrival, policy-relative abstention, and budget exhaustion.

2.2. Semantic maps for language-guided navigation

Semantic maps provide structured interfaces between language grounding and spatial decision making. VLMaps integrates pre-trained vision-language features with metric 3D reconstruction techniques to achieve language-indexed localization of open-vocabulary navigation targets [19]. ConceptGraphs and HOV-SG are no longer limited to dense feature maps. They organize open-vocabulary scene information into graph or hierarchical scene graph representations, thereby supporting spatial reasoning and language-grounded planning [20,21]. MapNav uses annotated semantic maps as memory representations for VLM-based VLN, replacing historical visual frames with structured, top-down representations of scene semantics [22]. LLM-guided semantic search methods demonstrate that common-sense prior information can effectively guide navigation to target objects. ESC applies soft commonsense constraints during zero-shot object exploration, while osmAG-LLM combines semantic maps with LLM reasoning for open-vocabulary object navigation [10,23]. These studies demonstrate how semantic maps and prior knowledge from LLMs and VLMs support language-guided search. However, these methods are primarily aimed at online map construction, dense open-vocabulary mapping, object-centered search, or memory representations. In pre-annotated maps, the integrated decision-making challenges involving annotated functional regions, observed object instances, and missing targets have not yet been fully studied.

2.3. Stopping and failure decisions

Stopping is a core component of VLN, but it is often implicit. In R2R, REVERIE, and RxR, when an agent determines that it has reached the destination or referent specified by an instruction, it executes a STOP action [4,6,11]. This formulation primarily treats stopping as an arrival decision and assumes that instructions are feasible. Progress-aware VLN models make this decision more explicit by estimating navigation progress or the probability of the target appearing along the trajectory. Self-Monitoring employs an auxiliary progress estimator to coordinate instruction completion with action selection [24]. Mind the Gap analyzes the difference between the success rate (SR) and the "oracle" SR, and recovers missed stop locations by revisiting candidate trajectories [25]. These studies have improved target localization and stopping behavior. However, the termination action still depends on reaching an existing target.

Recent studies have begun to explore infeasible instructions. VLN-NF introduces "false premise" VLN episodes, in which the referenced target does not exist in the specified room, and the agent must output `NOT-FOUND` after collecting evidence [26]. This setup elevates "failure to reach the goal" to a primary decision, rather than treating it as a mere byproduct of evaluation. HospNav follows the same explicit non-arrival reporting principle but examines a different formulation. The decision to return `FAILED`. is grounded in a known hospital semantic map and is coupled to coverage of high-prior

functional regions, rather than being treated only as an in-room exploration outcome. Since this condition depends on the prior, thresholds, visited anchors, sensor evidence, and a finite budget, we consistently use "policy-relative abstention" throughout this paper and do not equate `FAILED` with a proof of global non-existence.

3. Problem formulation

We consider an episode of an open-vocabulary navigation task on a known map in a hospital environment. The robot receives a natural language instruction L and a pre-annotated semantic map

$$\mathscr{M} = (M_{\text{occ}}, \mathscr{R}), \tag{1}$$

where $M_{\text{occ}} \in \{0, 1, -1\}^{H \times W}$ is an occupancy grid representing free, occupied, and unknown cells, and $\mathscr{R} = \{r_i\}_{i=1}^{N}$ is a finite set of semantic regions. Each region is represented as

$$r_i = (\text{id}_i, n_i, \mathscr{A}_i, P_i, a_i), \tag{2}$$

where id_i is a unique identifier, n_i is the canonical name, $\mathscr{A}_i$ is a set of aliases, P_i is the polygonal region boundary, and $a_i \in \text{SE}(2)$ is a reachable navigation anchor.

At time t, the robot has pose $\theta_t \in \text{SE}(2)$ and an RGB-D observation

$$o_t = \langle I_t^{\text{rgb}}, I_t^{\text{depth}}, K, T_{\text{base}}^{\text{cam}} \rangle, \tag{3}$$

where K denotes the camera intrinsics and $T_{\text{base}}^{\text{cam}}$ denotes the fixed base-to-camera extrinsics. The language-guided decision layer maps the instruction, semantic map, observation history, and current pose to either a metric goal or an explicit abstention:

$$\mu(L, \mathscr{M}, o_{\le t}, \theta_t) \to u_t \in \text{SE}(2) \cup \{\texttt{FAILED}\}. \tag{4}$$

When $u_t \in \text{SE}(2)$, it is a goal pose g_t passed to the downstream navigation stack. The agent emits the episode-level status `REACHED` after reaching and accepting its selected metric goal, or `FAILED` when HospNav abstains after its bounded search gates are satisfied. The episode ends as `TIMEOUT` when the 600 s simulated-time budget is exhausted. The evaluator scores a present-target `REACHED` as geometrically successful within a frozen 2.0 m radius of an admissible ground-truth target. Therefore, `FAILED` represents a semantic abstention decision rather than a failure of the navigation stack: it indicates that both the policy's coverage and no-match conditions have been satisfied, not that the target is absent.

This problem formulation distinguishes between the target source and the search outcome. For region-grounded instructions, the target is an annotated semantic region, and the desired goal is the corresponding anchor pose. For object-grounded instructions, the target is an observed object instance, while the desired goal is a reachable pose associated with that instance. For the agent, the "target missing" instruction is not a separate category. The agent receives only this instruction and its regular online inputs. Target missing is determined by the evaluator based on the true state of the frozen scene, and `FAILED` outputs in such experiments will be counted as correct policy-relative rejections. Compound service commands involving objects and destinations require multi-stage task planning, which is beyond the scope of this framework.

4. Method

4.1. System overview

Figure 1 illustrates the three interacting modules in HospNav.

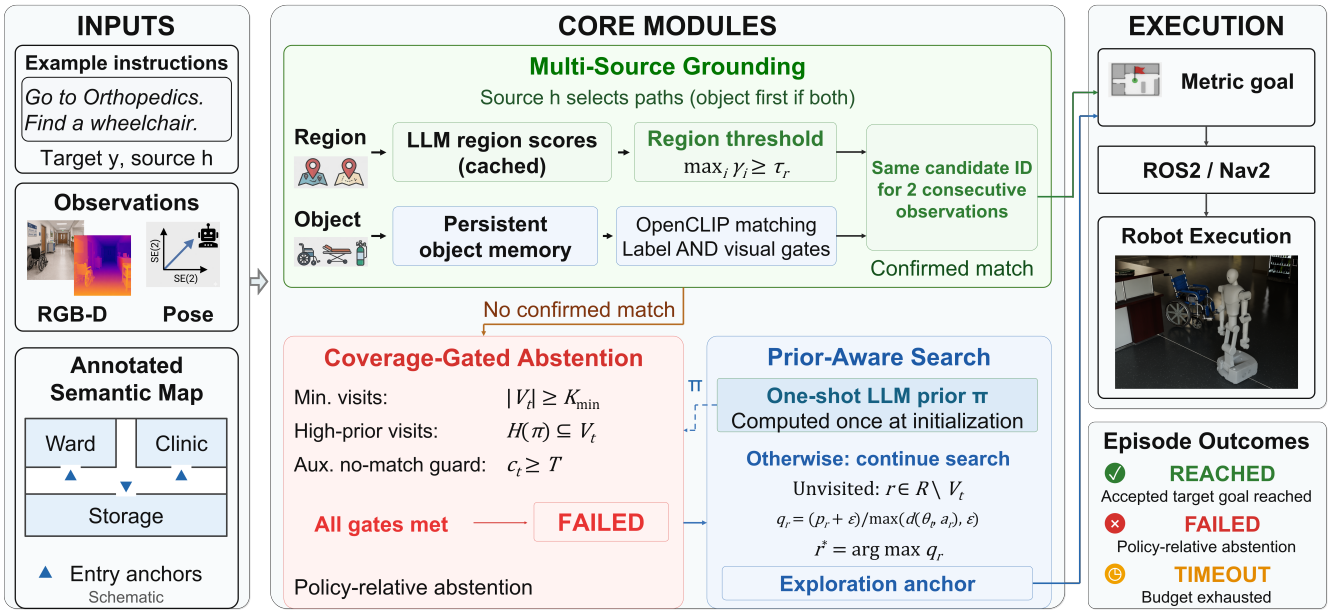


Figure 1. Overview of HospNav. Multi-source grounding confirms region or object matches and provides metric goals for ROS2/Nav2. Without a confirmed match, coverage gates and a no-match guard determine whether to return policy-relative `FAILED` or continue exploration guided by the one-shot region prior and distance weighting.

HospNav, based on a predefined semantic map of a hospital, converts free-form instructions into a series of metric goals. For a given instruction L, the instruction parser identifies the target phrase y and a coarse source hint $h \in \{\texttt{region}, \texttt{object}, \texttt{both}\}$. At each step, the agent-level grounding interface evaluates whether y corresponds to a labeled semantic region or an observed object instance. On a match, the system returns the target pose g_t. Otherwise, the one-shot region prior selects the next labeled region to visit. The process alternates between grounding and prior-guided navigation until the target is identified or the prior-aware abstention rule returns `FAILED`. The downstream ROS2/Nav2 stack is responsible for path planning and tracking.

4.2. Instruction and detector-vocabulary interface

The instruction parser is called once per episode at temperature 0. Its fixed system prompt requires a strict JSON format, containing two semantic fields: `target_text` (a short target noun phrase) and `source_hint` (whose value is one of `region`, `object`, or `both`). This prompt assigns "find/look for/is there" requests to the object path, explicit "go/navigate/head/wait in" requests to the region path, and `both` to truly ambiguous expressions.

Open-vocabulary detectors still require a set of text queries. To avoid both the implicit use of an evaluator's vocabulary and tuning during testing, this interface deterministically constructs an ordered

vocabulary for each episode based on a fixed hospital base list, the original and parsed target phrases, and a frozen alias table, without referencing scene object truth. Appendices Appendix A. and Appendix B. list the complete LLM prompts, detector vocabulary, alias mappings, OpenCLIP templates, and the frozen-replay contract.

4.3. Hospital semantic map representation

This section describes how the map in Equations (1) and (2) is constructed in practice: the polygon P_i, the alias set $\mathscr{A}_i$, and the navigation anchor a_i.

The anchor is located at a reachable entry point, typically near a doorway or the region boundary, rather than at the polygon centroid. At this location, an anchor can serve both as a direct navigation target for region-class instructions and as a convenient inspection point for object-class searches within a room. The anchor orientation is automatically calculated based on the direction from the anchor point to the centroid:

$$\mathrm{yaw}_r = \mathrm{atan2}(c_r^y - a_r^y,\ c_r^x - a_r^x), \tag{5}$$

where c_r represents the centroid of the polygon, defined as the arithmetic mean of its vertices.

Polygons represent the spatial extent of rooms, hallways, and irregular functional areas, while the alias set translates hospital-specific terminology into natural language descriptions. Therefore, the map can still be interpreted on a symbolic level without requiring all users to strictly adhere to the original terminology used in the floor plan annotations.

4.4. Multi-source grounding

Grounding checks for the target specified by y in the current state and returns its geometric goal if found. HospNav employs an agent-level multi-source grounding interface that includes two acceptable sources: annotated regions and observed object instances. The source hint h specifies which sources are queried. When both sources are permitted, object grounding is prioritized over region grounding to prevent prematurely accepting a seemingly reasonable room before the requested object has been observed.

For region grounding, the scorer F_R maps the target phrase and annotated regions to scores

$$F_R(y, \mathscr{R}) = \{\gamma_i \in [0,1]\}_{r_i \in \mathscr{R}}. \tag{6}$$

The region path selects the region with the highest score, resolving ties by proximity to the robot, and accepts this region only if:

$$s^{\mathrm{region}} = \max_i \gamma_i \geq \tau_r. \tag{7}$$

If this condition is met, the anchor in the matching region becomes the target. The main system employs an LLM-based scorer to implement F_R.

For each object instance $o_j \in \mathscr{O}_t$, the object memory records the class label ℓ_j, geometric position, and visual feature $\mathbf{f}_j$. The target phrase and object labels are encoded as unit-normalized OpenCLIP text vectors $\mathbf{e}_y$ and $\mathbf{e}_{\ell_j}$, respectively, following CLIP-style contrastive language-image training and utilizing the DataComp-trained OpenCLIP backbone as implemented in this work [27,28]. Object matching consists of two stages:

(1) Label gating: Filters categories based on text-text similarity, retaining only candidates that satisfy

the condition $\langle \mathbf{e}_y, \mathbf{e}_{\ell_j} \rangle \geq \tau_{\text{label}}$.

(2) Visual matching: Among the filtered objects, candidate objects are ranked based on text-image similarity $s_j^{\text{obj}} = \langle \mathbf{e}_y, \mathbf{f}_j \rangle$, and the highest-ranked candidate is accepted only if $s_j^{\text{obj}} \geq \tau_o$.

The region and object paths generate scores with distinct semantics and scales, so each path applies its own acceptance threshold rather than directly comparing raw scores. A candidate that passes its source-specific threshold must remain selected for two consecutive observations before the agent locks it in. The accumulator uses the persistent candidate ID as the key: an unmatched frame resets it, a different ID resets the count to one, and a lock is created when the count reaches two. If neither of the two admissible grounding paths yields a confirmed match, the agent will continue with prior-guided exploration.

In each independent episode, persistent object memory is cleared. Within an episode, RGB-D detections are projected into world coordinates and merged into existing instances when their spatial, visual, and label consistency satisfies the following condition:

$$A(d, o_j) = \left(a_{\text{pos}}\, a_{\text{vis}}\, a_{\text{lab}}\right)^{1/3} \geq 0.5, \tag{8}$$

where a_{pos}, a_{vis}, and a_{lab} represent scores for centroid proximity, visual feature similarity, and label consistency, respectively. Subsequently, the matched centroids, features, and confidence scores are updated using cumulative averages weighted by the number of observations, and confidence scores decay over time since the last observation, thereby pruning stale instances. Appendix Appendix B. provides the thresholds and decay constants.

4.5. LLM-guided region prior

When grounding fails to identify the target in the current observation, the system selects the next region to approach. The system employs a one-shot LLM-guided region prior. At the start of each episode, the target phrase y and a complete, ordered list containing region IDs, canonical names, and aliases are scored once by Qwen-Plus [29] at temperature 0, returning a numerical value for each region, where 1 on its scale indicates a strong association and 0 indicates no association.

$$\pi = F_{\text{prior}}(y, \mathscr{R}) = \{r_i \mapsto p_i \in [0,1]\}_{r_i \in \mathscr{R}}. \tag{9}$$

A prior is used to estimate the regions where the target is likely to be located semantically. However, it does not independently verify the existence of the target.

The prior π is computed once per episode, rather than at each navigation step [23,30]. This method is applicable to known hospital maps because the set of regions is finite, and within a single episode, the semantic relationships between target phrases and room functions remain largely unchanged. These scores are cached for reuse throughout the episode. Appendix B states the region-line template, the JSON contract, and the replay rules.

4.6. Prior-aware visit planning

Given the region prior π, the visit planner determines the next region anchor to visit using a prior-weighted nearest-distance rule:

$$r^* = \arg\max_{r \in \mathscr{R} \setminus \mathscr{V}_t} \frac{p_r + \varepsilon}{\max(d(\theta_t, a_r), \varepsilon)}, \tag{10}$$

where $\mathcal{V}_t$ is the set of regions visited up to time t, and $d(\theta_t, a_r)$ is the Euclidean distance from the robot's current pose to anchor a_r. The parameter $\varepsilon = 10^{-6}$ prevents division by zero and allows regions with $p_r = 0$ to be visited after all regions with nonzero prior have been explored.

When the robot enters the vicinity of an anchor, the associated region is added to $\mathcal{V}_t$. This method prevents oscillation between adjacent high-prior regions and provides a discrete search history for the failure rule.

HospNav uses a set of known regions and a prior π to establish a bounded abstention condition.

$$\mathcal{H}(\pi) = \{r_i : p_i \geq \rho \cdot \max_j p_j\} \tag{11}$$

Let denote the high-prior region set, where ρ is the high-prior ratio. A robot may return `FAILED` only when both of the following coverage conditions are satisfied:

- Count gate. The robot has visited at least $K_{\min}$ distinct anchor points: $|\mathcal{V}_t| \geq K_{\min}$.
- Identity gate. All high-prior regions have been visited: $\mathcal{H}(\pi) \subseteq \mathcal{V}_t$.

The count gate specifies the minimum number of distinct anchors that must be visited, while the identity gate specifies which high-prior anchors must be included. Under a flat prior, $\mathcal{H}(\pi)$ includes all labeled regions, whereas under a peaked prior it includes regions with scores exceeding a relative threshold. This implementation also requires that there be no accepted matches within T consecutive frames ($c_t \geq T$) as an auxiliary safeguard against transient perception loss. Section 5.5 evaluates the gates individually and in combination. The anchor arrival radius δ_v, $K_{\min}$, ρ, and T use the values from Table 1 and the accompanying text. See Algorithm 1 for the overall process.

Algorithm 1 HospNav Main Loop

Require: Instruction L; semantic map $(M_{\text{occ}}, \mathcal{R})$
$\langle y, h\rangle \leftarrow$ ParseInstruction(L)
$\pi \leftarrow$ RegionPrior$(y, \mathcal{R})$ ▷ one-shot score
$\mathcal{V} \leftarrow \emptyset$; $\mathcal{H} \leftarrow \{r : p_r \geq \rho \max_{r'} p_{r'}\}$
while episode budget remains **do**
  $\langle o_t, \theta\rangle \leftarrow$ GetObservation$()$
  $\mathcal{O}_t \leftarrow$ UpdateObjectMemory(o_t, θ)
  $\mathcal{V} \leftarrow \mathcal{V} \cup \{r : d(\theta, a_r) < \delta_v\}$
  $\langle \texttt{status}, g\rangle \leftarrow$ Ground$(y, h, \mathcal{R}, \mathcal{O}_t)$
  if `status` = `MATCHED` and the match is temporally confirmed **then**
    NavigateTo(g)
    if navigation succeeds **then**
      return `REACHED`
    end if
  end if
  if $|\mathcal{V}| \geq K_{\min}$ and $\mathcal{H} \subseteq \mathcal{V}$ and no-match evidence persists for T frames **then**
    return `FAILED` ▷ coverage gates plus the auxiliary guard
  end if
  $r^* \leftarrow$ SelectNextRegion$(\pi, \mathcal{V}, \theta)$ ▷ Equation (10)
  NavigateTo(a_{r^*})
end while
return `TIMEOUT`

5. Experiments

5.1. Experimental setup

We use Isaac Sim as the simulation platform. Its hospital scene, `hospital_v1`, comprises 17 manually annotated semantic regions, described in Section 4.3. These include clinical rooms, storage areas,

restrooms, patient rooms, emergency exits, and public service areas. The occupancy grid map is generated from the scene's ground-truth geometry. The region polygons and anchors are stored in a YAML file. The resulting map is shown in Figure 2.



Figure 2. Annotated hospital map used in the evaluation. 17 semantic regions and their navigation anchors are overlaid on the occupancy map.

The implementation operates on ROS2 Jazzy and utilizes Nav2 [31]. The perception stack employs YOLO-World (`yolov8x-worldv2`) [32] with the deterministic task-conditioned query interface described above, MobileSAM [33,34] for mask refinement, and depth projection to maintain the object memory $\mathcal{O}_t$. Object features are encoded using the OpenCLIP Vision Transformer (ViT) model ViT-L-14 trained on DataComp-XL [28], and the LLM scorer uses Qwen-Plus at temperature 0. Nav2 executes the metric goals of HospNav using NavfnPlanner and RegulatedPurePursuitController. Key settings remain constant across all ablation experiments, and see Table 1 for details.

We use the same gating parameters for all prior configurations: $K_{\min} = 6$ for minimum search coverage, $\rho = 0.25$ for high-prior region selection, and $T = 50$ frames (5s at 10Hz) to avoid premature failure detection. Increasing ρ favors earlier `FAILED` decisions, while decreasing ρ or increasing $K_{\min}$ or T raises the risk of timeout.

Table 1. Core runtime and evaluation settings used in the experiments described in this report.

Setting	Value	Role
Region grounding	$\tau_r = 0.50$	Accepts LLM region matches
Object grounding	$\tau_{\text{label}} = 0.30,\ \tau_o = 0.20$	Label-gate and text-image matching thresholds
Anchor-arrival radius	$\delta_v = 2.0$ m	Agent parameter: marks a region visited
Evaluator success radius	$\delta_{\text{succ}} = 2.0$ m	Evaluator parameter: geometric success test
FAILED gates	$K_{\min} = 6,\ \rho = 0.25,$	Count and identity coverage gates
Simulation budget	600 s from ROS2 `/clock`	The decision count limit has been disabled (10 Hz is the call frequency of the driver).

5.2. Tasks, outcomes, and baselines

The core split comprises three categories of instructions, with 10 instructions in each category: regions, existing objects, and missing targets. Region instructions target a specific semantic region, object-present instructions target instantiated object classes, and target-missing instructions cover six missing object classes and four missing region labels. Each instruction is repeated from the same three start poses, producing 90 episodes per region-prior configuration and 270 internal-ablation episodes.

Outcome classification and metrics: For region and object experiments where the target exists, a correct `REACHED` requires both the corresponding termination action and geometric success within a $\delta_{\text{succ}} = 2.0$ m radius of any acceptable ground-truth target. All other attempts are classified as geometrically incorrect `REACHED` (false acceptance), premature `FAILED`, `TIMEOUT`, and invalid infrastructure/protocol states. For trials with missing targets, HospNav results are policy-relative `FAILED`, false `REACHED`, or `TIMEOUT`, and results for "object missing" and "region missing" will be reported separately. SR is the fraction of geometrically successful present-target trials, and success weighted by path length (SPL) is $\text{SPL} = \frac{1}{N}\sum_i S_i \cdot \frac{\ell_i^*}{\max(\ell_i,\ell_i^*)}$, where ℓ_i^* terminates at the first reachable point inside the success radius and ℓ_i is the executed path length. For stopping analysis, the termination time and path length are based on the average across all source episodes, rather than solely on the average of successful trials. The policy-relative FAILED rate refers to the proportion of absent-target trials that end in a policy-relative `FAILED`. Its corresponding false acceptance rate and timeout rate are always displayed together to ensure that timeouts are not misinterpreted as successful rejections.

Internal ablations: We compare three region-prior configurations under the same known-map grounding and navigation stack:

(1) Uniform (control): setting $p_r = 1$ for all r yields nearest-unvisited-first selection.

(2) CLIP: a non-LLM semantic-prior baseline. With unit-normalized OpenCLIP text features, $p_r = \big(1 + \max_{z\in\{n_r\}\cup\mathscr{A}_r}\langle \mathbf{e}_y, \mathbf{e}_z\rangle\big)/2$. The maximum preserves a strong alias match, while the affine transformation maps cosine similarity from $[-1,1]$ to $[0,1]$.

(3) LLM: our method, with one-shot scoring by Qwen-Plus.

All three configurations use the same grounding and navigation modules. The region prior changes both the visit order and the regions that must be searched before `FAILED`. Therefore, this experiment compares complete prior-conditioned policies rather than the independent quality of LLM grounding.

External baseline: In addition, we evaluated CoW-FBE (OWL-ViT-B/32), which is our adaptation of CoW based on Isaac/ROS2. To evaluate CoW-FBE under the same Isaac Sim protocol, we integrated the FBE and OWL-ViT object localization components with the same ROS2/Nav2 navigation stack used in HospNav [7]. The agent receives RGB-D data, pose information, and object instructions, but does not receive annotated semantic maps, region labels, LLM prior information, or object-truth files. The same reset procedure, scene, starts, 600 s budget, and primary success radius are used. Because CoW-FBE has neither a region interface nor a `FAILED` action, it is evaluated on the ten present-object and six object-absent core instructions. On absent objects, its outcomes are false `REACHED` or `TIMEOUT`, with timeouts excluded from correct rejections.

5.3. *Confirmatory evaluation protocol*

The confirmatory evaluation comprises 378 validated episodes: 270 core HospNav episodes (Uniform, CLIP, and LLM priors over the 30 core instructions), 48 external episodes (CoW-FBE over the 10 present objects and 6 object-absent targets), and 60 Stopping-OFF episodes (LLM HospNav over the 10 core objects and 10 absent targets). All methods use the same three starts and a 600 s simulated-time budget.

A separate 60-episode diagnostic collection is described in Section 5.6 and its 30 present-object episodes also serve as the original-layout condition in the 90-episode controlled evaluation in Section 5.7.

Statistical analysis: The instruction, not an individual start, is the independent sampling unit. The reported 95% instruction-cluster bootstrap confidence intervals (CIs) are computed from 10,000 resamples with seed 42, retaining the three starts within each sampled cluster. LLM *versus* Uniform and LLM *versus* CLIP are compared using exact paired sign-flip tests on instruction-level mean differences. Fisher's exact test provides a secondary, unpaired analysis. Both tests use Holm adjustment for the two comparisons within each endpoint. We report effect sizes with their interval and p-value estimates. Near-saturated region performance is treated descriptively.

Stopping analysis: In a dedicated LLM run, the policy abstention mechanism was disabled while maintaining the same 600 s budget, and complete frame-by-frame stop evidence was recorded. Each trajectory is replayed by varying each factor around $(K_{\min}, \rho, T) = (6, 0.25, 50)$ one at a time, where $K_{\min} \in \{4,6,8,10,12,14,17\}$, $\rho \in \{0.05, 0.15, 0.25, 0.35, 0.50, 0.75, 0.90\}$, and $T \in \{0, 50, 500, 1000, 1500, 2000\}$ frames, thereby ensuring that each factor is explored within the range where it may influence decision-making. The same traces are additionally replayed under each of the seven non-empty gate subsets at the frozen parameters, selecting the earliest frame at which the active gates are simultaneously true. The replay reports present-object SR, premature-FAILED rate, absent policy-relative FAILED, false `REACHED`, `TIMEOUT`, termination time, and path length.

5.4. *Main results*

All 378 reported episodes are validated equal-budget runs. Table 2 reports the main-evaluation present-target outcomes.

Table 2. Present-target performance and complete outcome counts. SR and SPL intervals are 95% instruction-cluster bootstrap intervals.

Method/subset	*N*	**SR [95% CI]**	**SPL [95% CI]**	**False REACHED**	**FAILED**	**T/O**
Uniform/region	30	100.0 [100.0, 100.0]	.914 [.881, .940]	0	0	0
Uniform/object	30	30.0 [13.3, 50.0]	.160 [.062, .277]	18	2	1
CLIP/region	30	100.0 [100.0, 100.0]	.913 [.880, .940]	0	0	0
CLIP/object	30	40.0 [16.7, 63.3]	.177 [.061, .311]	17	1	0
LLM/region	30	100.0 [100.0, 100.0]	.911 [.878, .938]	0	0	0
LLM/object	30	56.7 [33.3, 80.0]	.367 [.150, .601]	10	3	0
CoW-FBE [7]/object	30	16.7 [0.0, 33.3]	.082 [.000, .169]	18	N/A	7

Region SR saturated at 100% for all three priors and is therefore a descriptive check of the shared region-grounding and Nav2 path rather than evidence that one prior is better. On objects, the LLM-conditioned policy achieved the highest point estimates: 56.7% SR and 0.367 SPL, *versus* 40.0%/0.177 for CLIP and 30.0%/0.160 for Uniform. HospNav also produced a substantial empirical gap over our Isaac/ROS2 CoW-FBE adaptation (56.7% *versus* 16.7% SR) under the same scene, starts, budget, and success radius. Differences in semantic map access and perception make this a system-level comparison, so it does not isolate the contribution of the decision policy. Figure 3 shows representative examples for each result category, pairing the scene as seen by the robot upon arrival with the path it followed.

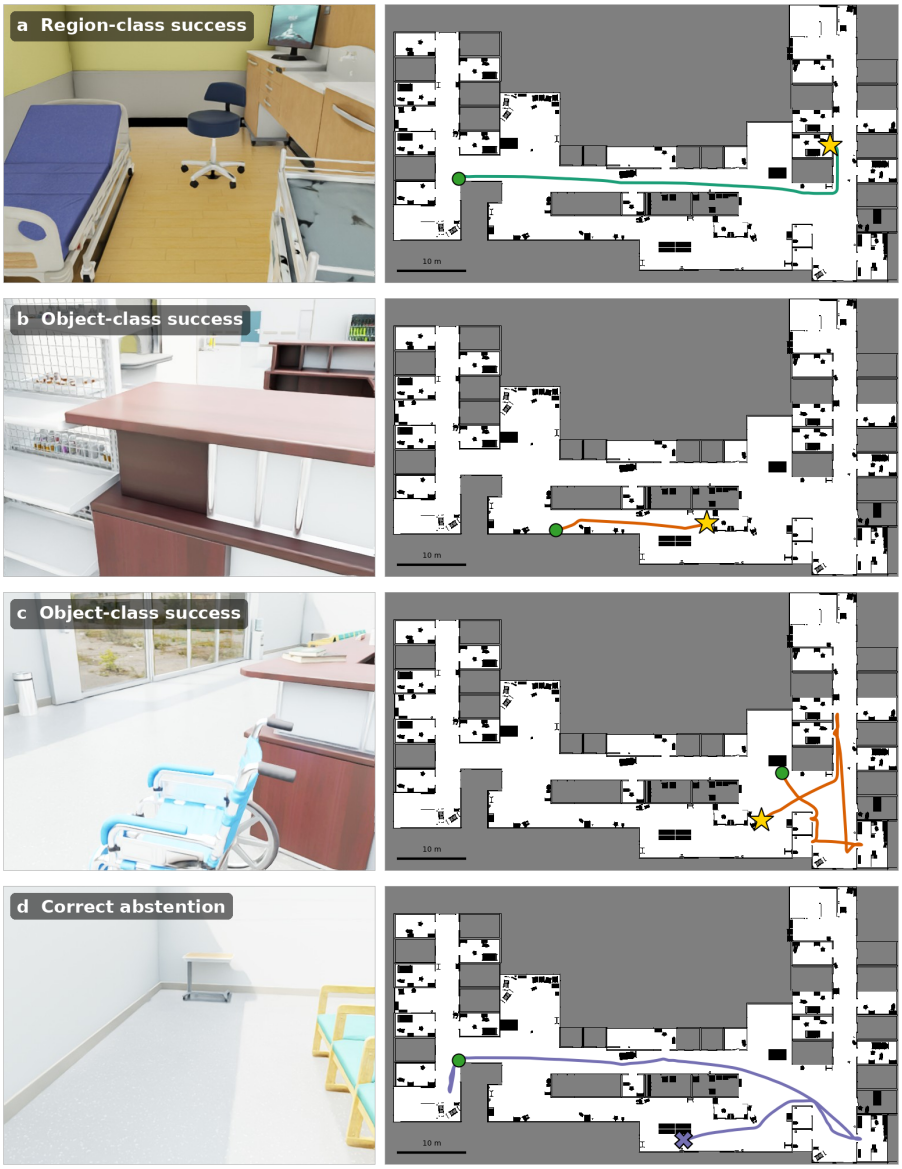


Figure 3. Representative confirmatory episodes from the LLM-prior run, pairing the on-arrival first-person view (left) with the executed trajectory on the annotated map (right). Green marks the start pose, a star a geometrically successful `REACHED`, and a cross the pose at which the policy returned `FAILED`. **(a)** Region-class success; **(b, c)** Object-class successes; **(d)** Correct abstention on an absent target.

Table 3 reports absent-target outcomes, and timeouts are not treated as rejections. The LLM-conditioned policy returned policy-relative `FAILED` in 66.7% of trials, compared to 63.3% for CLIP and 46.7% for Uniform. CoW does not have a `FAILED` operation: its ten timeouts were due to budget exhaustion, not a correct rejection.

Table 3. Aggregate absent-target outcomes. HospNav reports policy-relative `FAILED` outcomes. CoW-FBE timeouts do not count as rejections. The 95% bootstrap intervals, clustered by instructions, are provided in square brackets.

Method	*N*	**FAILED [95% CI]**	**False REACHED [95% CI]**	**TIMEOUT**
Uniform	30	14; 46.7 [20.0, 73.3]	15; 50.0 [26.7, 76.7]	1
absent object	18	8	9	1
absent region	12	6	6	0
CLIP	30	19; 63.3 [40.0, 86.7]	9; 30.0 [6.7, 56.7]	2
absent object	18	14	3	1
absent region	12	5	6	1
LLM	30	20; 66.7 [36.7, 90.0]	10; 33.3 [10.0, 63.3]	0
absent object	18	14	4	0
absent region	12	6	6	0
CoW-FBE/absent object	18	N/A	8; 44.4 [16.7, 72.2]	10

Table 4 presents the paired estimates from the ablation experiments. The object-SR contrasts against both baselines had positive point estimates, and the bootstrap intervals did not include zero. None of the Holm-adjusted comparisons in the table reaches the 0.05 significance level, for either the paired tests or the secondary Fisher tests. The absent-target contrasts have wider uncertainty relative to their point estimates, with the LLM–CLIP interval spanning zero.

Table 4. Instruction-paired inference for LLM *versus* internal priors. Differences are percentage points with 95% paired cluster-bootstrap intervals. The Fisher test is a secondary, unpaired analysis.

Endpoint/comparison	**Difference [95% CI]**	**Exact** p	**Holm** p	**Fisher** p	**Fisher–Holm** p
Object SR: LLM–Uniform	+26.7 [13.3, 43.3]	.0313	.0625	.0673	.1346
Object SR: LLM–CLIP	+16.7 [6.7, 26.7]	.0625	.0625	.3015	.3015
Absent FAILED: LLM–Uniform	+20.0 [3.3, 36.7]	.1250	.2500	.1923	.3846
Absent FAILED: LLM–CLIP	+3.3 [−6.7, 13.3]	1.000	1.000	1.000	1.000

Uniform and CLIP included all 17 regions in the high-prior set in every core episode, whereas LLM included 1–11 regions (mean 5.17, median 4). Therefore, the prior affects both the coverage required before `FAILED` and the order of visits.

5.5. Stopping policy validation and sensitivity

On the same traces, the replay reproduced all 60 actual Stopping-ON decisions made under the default parameters, including the exact terminal indices, and reconstructed each gate value recorded in the

Stopping-OFF traces. Independently executed rollouts agree less often (35/60 exact decision indices), which reflects run-to-run variation in Isaac/Nav2 and perception rather than replay fidelity.

Since the stopping rule determines only *when* to terminate and never influences goal selection or navigation, truncating a recorded trajectory at a different frame constitutes a precise conditional counterfactual for that trajectory, rather than a causal estimate for a newly executed trajectory. After applying frozen gating to 60 Stopping-OFF traces, the average termination time and path length decreased from 241.3 s and 133.0 m to 83.2 s and 76.7 m, respectively. The 21 absent-target traces that timed out with stopping disabled would instead terminate as `FAILED`. The trade-off is that 3/30 of the "target present" trajectories would abstain prematurely, while the success rate in the target-present trials drops from 19/30 to 17/30. Appendix D provides complete breakdown data.

Table 5 shows that count-only stopping matches the full rule on present-target outcomes, terminates earlier (61.5 *vs.* 83.2 s), and has a slightly higher `FAILED` rate when the target is absent. These results do not establish any additional performance benefit from the identity gate. Both variants retain the same LLM-conditioned, distance-aware visit planner. Therefore, the "count-only" stopping strategy still follows a prior-guided search trajectory, which may result in visiting high-prior regions before the count threshold is reached. Stopping solely based on "no match" evidence leads to frequent premature abstention. These results support requiring visits before abstention under the evaluated visit policy.

Table 5. Under fixed parameters $(K_{\min}, \rho, T) = (6, 0.25, 50)$, a gate-subset ablation experiment was conducted on the same set of 60 fixed trajectories. Every subset that retains a coverage gate reproduces the full policy on present targets. The "no match" gate alone triggers abstention approximately 5 s after episode onset, making it unusable on its own. The three omitted subsets reproduce the retained rows exactly—count + identity matches the default, identity + debounce matches identity only, and count + debounce matches count only—because the debounce is non-binding. This ablation fixes the trajectories to isolate the stopping-gate structure on LLM-prior traces.

Active gates	**Present SR**	**Prem. FAILED**	**Absent FAILED**	**Time (s)**
Count + identity + debounce (default)	56.7 [33.3,80.0]	10.0 [0.0,30.0]	70.0 [43.3,93.3]	83.2
Identity only	56.7 [33.3,80.0]	10.0 [0.0,30.0]	70.0 [43.3,93.3]	78.9
Count only	56.7 [33.3,80.0]	10.0 [0.0,30.0]	73.3 [46.7,96.7]	61.5
Debounce only	**13.3** [3.3,23.3]	**76.7** [60.0,93.3]	80.0 [50.0,100.0]	6.8

Figure 4 scans each parameter over a wide range that covers the observed activation range. $K_{\min}$ exhibits a monotonic trade-off: increasing it from 4 to 8 raises SR (when the target exists) from 56.7% to 63.3% and lowers premature `FAILED` from 10.0% to 3.3%. Increasing it to 17, the region count, approaches exhaustive search at 158.3 s. Lowering ρ expands the high-prior set, resembling a higher coverage requirement, while raising it to 0.9 gives the earliest termination at 61.5 s. The frozen $T = 50$ is non-binding: the results remained unchanged through $T = 500$, and the first observed change appeared at $T = 1000$. Frozen default values remain unchanged.

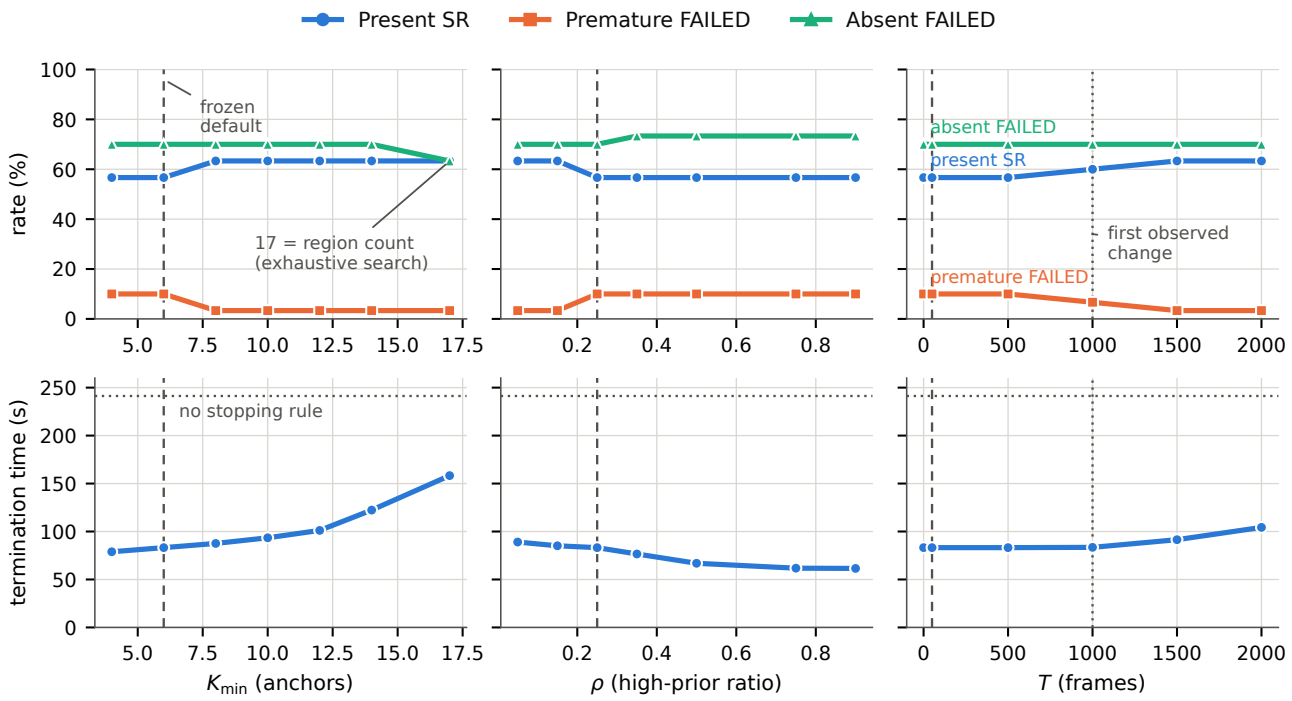


Figure 4. For 60 fixed Stopping-OFF trajectories, counterfactual stopping replays were performed by changing only one parameter at a time, with the scanning range covering the observed activation range. Top row: Result occurrence rates. Bottom row: Mean termination times over all 60 episodes. The dotted line shows actual Stopping-OFF results for reference. The dashed vertical lines mark the frozen default parameters $(K_{\min}, \rho, T) = (6, 0.25, 50)$. In the T column, the dotted line marks the lowest swept value at which any change in the decision occurred: results remained consistent up to and including $T = 500$, with the first observed change occurring at $T = 1000$. Path length tracks termination time throughout the sweep (Appendix D).

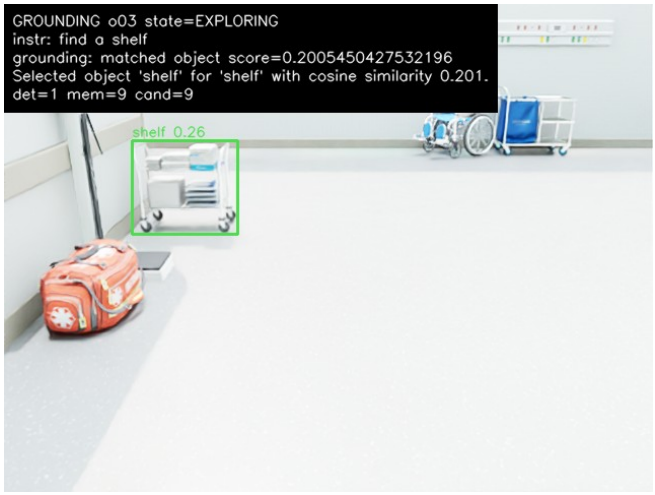
5.6. Failure analysis

Across three prior configurations, 86.5% of the failure cases were attributed to erroneous `REACHED` actions. These failures were divided almost evenly between cross-label matches and geometrically incorrect matches to the requested class. In the corresponding funnel plot, each episode contains an eligible object candidate. Appendix Appendix E. provides count statistics for each method.

Figure 5 shows two representative confirmatory traces, illustrating a false `REACHED` outcome and premature abstention on a present-target task.

An independent diagnostic dataset was created using Isaac Sim 6.0.0.1 to record synchronized RGB-D data and physical instance masks for the LLM-conditioned policy within the original hospital layout. This dataset contains 60 episodes, covering ten present-object and ten absent-target instructions, all starting from the same three initial positions, with fixed policy parameters. The visibility analysis uses the 30 present-object episodes, covering 9,875 processed frames. These diagnostic outcomes are reported separately from the 378-episode confirmatory evaluation.

(a)



(b)



Figure 5. Confirmatory LLM trace snapshots for the two failure modes discussed in the text. Left figure: The "shelf" hypothesis, which was accepted near the threshold, results in an erroneous `REACHED` as its terminal action. Right: the terminal frame of a present-computer episode ending in premature `FAILED`, with a best candidate score of 0.185 below the 0.200 matching threshold. The values shown in the figures are the system's own estimates, not measurements taken by evaluators. **(a)** `o03-s0-t0`: selected shelf hypothesis at score 0.2005; **(b)** `o01-s0-t0`: terminal frame with best score 0.185.

Table 6 traces the entire process of a real target, from visibility to detection, and finally to its inclusion in the grounding candidate pool. A target is considered visible when at least one acceptable physical instance occupies 0.1% of the image. Detection recall uses instance-mask bounding boxes with an intersection over union (IoU) of at least 0.5. Candidate recall follows actual detection-to-memory associations. Recall events are counted once per episode before termination, with detection and candidate recall conditioned on episode-level visibility. Appendix E specifies definitions related to time, association, and statistics.

Table 6. True target visibility and recall rates on an independent, original layout diagnostic dataset. Rates are expressed as percentages, accompanied by 95% bootstrap intervals clustered by instructions. Detection recall and candidate target recall are conditional on target visibility.

Episode-level metric	Count	Rate [95% CI]
Target visible	23/30	76.7 [53.3, 96.7]
Physical-instance detection conditional on visibility	23/23	100.0 [100.0, 100.0]
True-target candidate conditional on visibility	21/23	91.3 [78.9, 100.0]

The 11 failed present-object episodes in the diagnostic collection comprise seven without target observations, one with a true-target detection filtered before candidate formation, two with true-target candidates below the matching threshold, and one with a correct-instance lock but a terminal position outside the success radius. No failure involves a visible but undetected target. One of the two matching failures ends in premature `FAILED`, while the remaining ten failures end in false `REACHED`.

Across 24 visits to target-containing regions whose anchors were the active exploration goals before arrival, a target in the corresponding region was visible in 17 cases (70.8%) by departure or episode termination. Anchor visitation therefore does not ensure target visibility. Appendix E2 defines the visit selection and cumulative observation window.

5.7. Controlled placement and annotation changes

We evaluate the LLM-conditioned policy under controlled placement and annotation changes using the fixed settings in Table 1 and the simulation setup described in Section 5.6. The same ten present-object instructions are evaluated from three starts in each of three conditions, totaling 90 episodes. The original-layout condition reuses the 30 present-object episodes from the diagnostic collection. Under the relocation condition, ten instances across four categories (wheelchairs, water dispensers, trash cans, and drinks machines) were moved to semantically compatible service locations while maintaining their quantity, appearance, and orientation. The missing-annotation condition preserves the original object layout and removes the names and aliases of the operating room, vending area, and reception station, while retaining the geometry and anchor points of the regions. All three conditions use anonymous region identifiers.

Table 7 shows that repositioning objects caused the SR to change from 63.3% to 60.0%, and the SPL to change from 0.351 to 0.371. Removing region names and aliases reduces SR to 50.0% and SPL to 0.218, indicating that region semantics contribute to navigation performance in this setting.

Table 7. The "object target presence" navigation task was conducted under controlled variations in placement and labeling, with 30 episodes performed under each condition.

Condition	SR (%)	SPL
Original layout	63.3 (19/30)	0.351
Relocated objects	60.0 (18/30)	0.371
Missing annotations	50.0 (15/30)	0.218

6. Discussion

6.1. Method and rmpirical interpretation

HospNav is a method and system contribution centered on an executable decision policy. The one-shot prior guides evidence acquisition, persistent multi-source grounding determines target acceptance, and two coverage gates govern bounded abstention. Exact same-trace replay results indicate that this decision can be audited counterfactually, rather than existing solely as an implicit side effect of a timeout.

The LLM-conditioned policy has the highest object-navigation point estimates among the internal priors in the evaluated annotated-map setting. Paired estimates and results of tests corrected for multiplicity are presented together in Table 4. A 66.7% absent-target `FAILED` rate, combined with the policy's "false-`REACHED`" results, reflects its abstention behavior.

6.2. Limitations and directions for future research

This study is limited to a single synthetic hospital scene with a labeled semantic map. This study does not include standard multi-scene ObjectNav benchmarks such as HM3D-OVON [8]. The controlled placement and labeling changes in Section 5.7 capture within-hospital variability. Generalization across environments, categories, or sensors remains untested. When object locations do not conform to the typical functional layout of a hospital, the reliability of region priors may also decrease.

Failure in object search can occur either before the formation of true-target candidates or during the matching and target execution processes. Future work should improve observation and multi-view verification mechanisms and evaluate the complete system in more diverse environments. Finally, `FAILED` is a policy-relative decision based on current observations and the search budget. It does not establish the target's physical absence.

7. Conclusion

HospNav is a training-free framework for open-vocabulary target search on annotated semantic maps. Its core contribution is a prior-aware bounded-stop policy that explicitly returns an auditable, policy-relative `FAILED` outcome. Multi-source grounding, persistent object memory, temporal confirmation, and ROS2/Nav2 execution provide the evidence and control path for this policy. Across the controlled single-hospital evaluation, the LLM-conditioned policy produced the highest object point estimates (57% SR and 0.367 SPL), compared with 40%/0.177 for CLIP, 30%/0.160 for Uniform, and 17%/0.082 for our Isaac/ROS2 CoW-FBE adaptation, and returned `FAILED` on 67% of absent-target trials. Same-trace replay reproduced all 60 corresponding live Stopping-ON decisions, and counterfactual replay on the equal-budget Stopping-OFF traces quantified the efficiency–premature-abstention trade-off. The target-linked diagnostic collection further distinguishes failures of observation, candidate formation, matching, and localization/navigation. However, our evidence is limited to one synthetic hospital. Future work will evaluate HospNav's performance in more diverse environments and enhance object grounding through multi-view verification.

Appendix A. Frozen LLM prompts

The following text is the complete parser system prompt used to populate the frozen publication cache (the ASCII arrows below represent two examples from the source prompt):

```
You are an instruction parser for an indoor robot navigation assistant. The
user gives a natural-language instruction telling the robot where to go or
what to find. Your task is to parse the instruction into strict JSON.

Output format (must be valid JSON, no markdown code fences or extra text):
{
  "target_text": "the key target phrase extracted from the instruction",
  "source_hint": "region" or "object" or "both"
}

source_hint selection rules (apply the FIRST matching rule):
```

```
- "object": the user wants to locate a movable / device-like item, or any
  request phrased with "find a/the X", "look for a/the X", "is there a X",
  "see if there is a X", or "spot a X". This holds even if the X often
  appears in a particular room (e.g. "find a water dispenser" -> object,
  not both; "find a reception desk" -> object, even though receptions are
  rooms). The task is a visual search, not navigation to a place.
- "region": the user explicitly wants to go to or be in a static area,
  room, department, or facility. Phrasings like "go to / take me to /
  navigate to / head to / wait in <place>" or named rooms ("orthopedics",
  "room 305"). These are place-targeted.
- "both": ONLY when the surface phrasing genuinely could mean either,
  e.g. "where is a printer" (could be a printer object, or a print room).
  Prefer "object" or "region" whenever the verb makes the intent clear.

target_text extraction rules:
- Extract the noun or proper noun the user most cares about; keep it short.
- Drop function words / phrases such as "find", "take me to", "where is".
```

The parser user template is as follows:

```
User instruction: {instruction}

Return JSON.
```

The complete region-prior system prompt is as follows:

```
You estimate where an indoor target is likely to be found. Given a target
description and a list of regions (id / name / aliases), output strict
JSON {region_id: score in [0, 1]} with one entry per input id.
1.0 = strongly associated, 0.0 = unrelated. No prose, no code fences.
```

Its user message and repeated region-line template are as follows:

```
Target: {target_text}

Regions:
- {region_id} | name="{canonical_name}" | aliases="{comma-separated aliases}"

Return JSON: {"<id>": 0.x, ...} (one entry per region above)
```

Both calls were made using Qwen-Plus at temperature 0. Appendix B specifies the frozen replay cache.

Region grounding uses the same prompt, region-line rendering, Qwen-Plus model, and replay cache as the region-prior scorer. In the LLM-prior configuration, it reuses the per-episode prior scores. In the Uniform and CLIP configurations, requests with a region or both source hint obtain the same scorer's outputs from the frozen cache.

Appendix B. Implementation details

This appendix records the interface and threshold settings behind the method described in Section 4. All values were calibrated during development and fixed before the confirmatory outcomes were inspected. The reported object categories informed that calibration, so the values are not presented as optimal or as transferring to other categories. These settings were retained without retuning for the controlled-condition evaluation in Section 5.7. Per-episode seeds derive from a fixed base seed, but residual GPU and

simulator-scheduling nondeterminism remains.

Detector query construction: The per-episode vocabulary is

$$\mathcal{Q}(L) = \text{dedup}(\mathcal{Q}_{\text{base}} \cup \{y_{\text{raw}}, y\} \cup A_{\text{frozen}}(y_{\text{raw}}, y)), \tag{12}$$

where $\mathcal{Q}_{\text{base}}$ is the fixed hospital vocabulary, y_{raw} is the target phrase retained from the original request, y is the parser output, and A_{frozen} expands the target using fixed label mappings and spelling groups. The ordered base vocabulary contains the following 23 queries:

```
chair; computer; monitor; desktop computer; pc; desk;
bedside table; shelf; bookshelf; reception desk; reception counter;
front desk; check-in desk; wheelchair; water dispenser; water cooler;
toilet; urinal; lavatory; trash can; bin; drinks machine; vending machine
```

The label mappings are listed in Table A1. The raw target, parsed target, and mapped canonical target form the expansion seeds. When either label in a label-mapping pair matches the seed, that pair contributes both its alias and its canonical label. A spelling group in Table A2 contributes all its phrases when a seed matches any member after removing non-alphanumeric characters. The final query order is the base list, raw target, parsed target, canonical target, and selected aliases in configuration order. Queries are lowercased, whitespace is normalized, and duplicates are removed while retaining their first occurrence. The construction never queries scene object truth.

Table A1. All 21 frozen detector-label mappings. Each label in the left column maps to the canonical label on the right. The same mappings supply aliases during target-conditioned query expansion.

Detector labels	Canonical label
laptop, monitor, screen, desktop computer, pc, workstation	computer
dining table	desk
bookshelf, bookcase	shelf
bin, garbage bin, waste bin	trash can
vending machine	drinks machine
reception counter, front desk, check-in desk	reception desk
water cooler, water fountain	water dispenser
urinal, lavatory, commode	toilet

Table A2. All 11 frozen task-alias groups used in detector-query expansion. A matching group contributes the target phrase and every additional phrase.

Target phrase	Additional phrases
wheelchair	wheel chair
trash can	trashcan
bedside table	bed-side table
chart rack	medical chart rack
first aid kit	first-aid kit
hospital bed	medical bed
medical cabinet	medicine cabinet
supply cabinet	supply cupboard
supply cart	supply trolley
washbasin	wash basin
xray viewer	x-ray viewer, x ray viewer

Candidate filtering: YOLO-World uses a 0.25 confidence threshold, 0.50 non-maximum-suppression threshold, and at most 12 detections per frame. Candidates must pass fixed box/mask-size checks, contain at least 5% valid depth support with at least 20 pixels between 0.15 and 6.0 m, have depth spread no greater than 2.5 m, and project within 8.0 m of the camera. Mask crops are expanded by 10% before OpenCLIP feature extraction.

Object-memory association and update: In Equation (8), a_{pos} uses the spatial threshold of 1.5 m, a_{vis} uses the normalized visual similarity threshold of 0.7, and a_{lab} uses a soft label consistency score: a score of 1 is assigned when labels match, and 0.2 otherwise. Matched 3D centroids, normalized visual features, and detector confidence scores are all updated using cumulative averages weighted by the number of observations. Confidence is multiplied by 0.95 per 5 s without observation. Instances below 0.1 confidence or older than 30 s are pruned. Target-text features use the following 15 OpenCLIP templates, where `{}` is replaced by the target phrase. The target feature is the unit-normalized mean of unit-normalized prompt embeddings. Candidate labels use their canonical text without this ensemble.

```
a photo of a {}.
a photo of the {}.
an image of a {}.
a clear photo of a {}.
a close-up photo of a {}.
a low resolution photo of a {}.
a bright photo of a {}.
a dark photo of a {}.
a photo of one {}.
a photo of a {} in a hospital.
a photo of a {} in a clinic.
a photo of a {} in a hallway.
a photo of a {} in a room.
a {} in a hospital.
this is a photo of a {}.
```

Region-prior contract: The scorer uses the prompt and region-line template in Appendix A. Its output must be strict JSON with exactly one numeric score for every supplied region ID, and values are clamped to $[0, 1]$.

Frozen replay cache: Each request/response pair from the parser and the prior scorer is recorded once and replayed without accessing the network. An error will be triggered if there is a cache miss or if the response is invalid or incomplete. Each cache key is a SHA-256 digest of a serialized request containing the messages, model, temperature, and maximum-token setting.

Supplementary materials: The supplementary file `R2_4_frozen_inputs.zip` provides the frozen instructions, configurations, and cached LLM requests and outputs used in the confirmatory evaluation.

Appendix C. Per-target present-object outcomes

Table A3 reports the descriptive breakdown underlying the aggregated results for "target object present." Each target uses the same three frozen starts. The table does not compare held-out categories.

Table A3. Per-target present-object outcomes over three starts. Each cell lists success/false REACHED/premature FAILED/TIMEOUT, with column totals of 30 confirmatory episodes per method.

Target	Uniform	CLIP	LLM
chair	1/2/0/0	1/2/0/0	1/2/0/0
computer	0/2/1/0	0/2/1/0	0/1/2/0
desk	0/2/0/1	0/3/0/0	0/3/0/0
shelf	0/3/0/0	0/3/0/0	1/2/0/0
reception desk	2/1/0/0	2/1/0/0	3/0/0/0
wheelchair	1/2/0/0	3/0/0/0	3/0/0/0
water dispenser	3/0/0/0	3/0/0/0	3/0/0/0
toilet	1/1/1/0	1/2/0/0	2/0/1/0
trash can	0/3/0/0	0/3/0/0	1/2/0/0
drinks machine	1/2/0/0	2/1/0/0	3/0/0/0

Appendix D. Stopping sensitivity sweep

Table A4 presents the numerical data underlying Figure 4, including interval estimates and the path length column omitted from the figure.

Table A4. Actual Stopping-OFF results and counterfactual replays varying one parameter at a time on the same 60 trajectories. Rates are percentages with 95% instruction-cluster intervals. Time and path are means over all 60 episodes. Outcomes remain identical through $T = 500$, with the first observed change at $T = 1000$. The debounce is satisfied at decision index 49, whereas the coverage gates first hold together between indices 1015 and 2628, and the smallest consecutive no-match count reached at coverage across the 60 traces is 899. $K_{\min} = 17$ equals the region count and degrades the rule toward exhaustive search.

Condition	Present SR	Prem. FAILED	Absent FAILED	Absent false REACHED	Absent TIMEOUT	Time (s)	Path (m)
Stopping disabled (live source)	63.3 [40.0,83.3]	0.0 [0.0,0.0]	0.0 [0.0,0.0]	30.0 [6.7,56.7]	70.0 [43.3,93.3]	241.3	133.0
Default $(6, .25, 50)$	56.7 [33.3,80.0]	10.0 [0.0,30.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	83.2	76.7
$K_{\min} = 4$	56.7 [33.3,80.0]	10.0 [0.0,30.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	78.9	73.0
$K_{\min} = 8$	63.3 [40.0,83.3]	3.3 [0.0,10.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	87.6	80.5
$K_{\min} = 10$	63.3 [40.0,83.3]	3.3 [0.0,10.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	93.4	85.6
$K_{\min} = 12$	63.3 [40.0,83.3]	3.3 [0.0,10.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	101.1	92.7
$K_{\min} = 14$	63.3 [40.0,83.3]	3.3 [0.0,10.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	122.3	112.6
$K_{\min} = 17$	63.3 [40.0,83.3]	3.3 [0.0,10.0]	63.3 [36.7,86.7]	30.0 [6.7,56.7]	6.7 [0.0,16.7]	158.3	132.4
$\rho = 0.05$	63.3 [40.0,83.3]	3.3 [0.0,10.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	89.0	81.9
$\rho = 0.15$	63.3 [40.0,83.3]	3.3 [0.0,10.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	85.1	78.5
$\rho = 0.35$	56.7 [33.3,80.0]	10.0 [0.0,30.0]	73.3 [46.7,96.7]	26.7 [3.3,53.3]	0.0 [0.0,0.0]	76.6	70.5
$\rho = 0.50$	56.7 [33.3,80.0]	10.0 [0.0,30.0]	73.3 [46.7,96.7]	26.7 [3.3,53.3]	0.0 [0.0,0.0]	66.9	61.5
$\rho = 0.75$	56.7 [33.3,80.0]	10.0 [0.0,30.0]	73.3 [46.7,96.7]	26.7 [3.3,53.3]	0.0 [0.0,0.0]	61.8	56.5
$\rho = 0.90$	56.7 [33.3,80.0]	10.0 [0.0,30.0]	73.3 [46.7,96.7]	26.7 [3.3,53.3]	0.0 [0.0,0.0]	61.5	56.2
$T = 0$	56.7 [33.3,80.0]	10.0 [0.0,30.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	83.2	76.7
$T = 500$	56.7 [33.3,80.0]	10.0 [0.0,30.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	83.2	76.7
$T = 1000$	60.0 [36.7,83.3]	6.7 [0.0,20.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	83.5	76.9
$T = 1500$	63.3 [40.0,83.3]	3.3 [0.0,10.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	91.5	83.7
$T = 2000$	63.3 [40.0,83.3]	3.3 [0.0,10.0]	70.0 [43.3,93.3]	30.0 [6.7,56.7]	0.0 [0.0,0.0]	104.3	95.4

Appendix E. Grounding funnel and failure modes

Table A5 gives the per-method funnel and failure-mode breakdown behind the failure analysis in Section 5.6. The funnel records any candidate at each stage in the confirmatory runs. The following target-linked analysis uses the separate diagnostic collection described in Section 5.6.

Table A5. Episode-Level Grounding Funnel and Failure Modes. Funnel stages are episodes with an object candidate in memory (M), a label/feature-eligible candidate (E), a score above τ_o (A), a confirmed target lock (L), and geometric success (S), out of 30 episodes per method. Failure-mode counts are mutually exclusive and cover all 21/18/13 Uniform/CLIP/LLM failures. The two lock columns distinguish cross-label false locks from same-label geometric false `REACHED` actions.

Method	Episode funnel $M \to E \to A \to L \to S$	Cross-label lock	Same-label geometric	Matched, unconfirmed	Below τ_o
Uniform	30→30→28→27→9	10	8	1	2
CLIP	30→30→29→29→12	9	8	0	1
LLM	30→30→28→27→17	4	6	1	2

Appendix E.1. Target-linked diagnostic definitions

The diagnostic target set consists of frozen and admissible physical instances for each instruction. The visibility requirement is that at least one such instance occupies 0.1% of the RGB image or 308 pixels at a resolution of 640×480. This analysis uses synchronized instance masks and policy frames until the end of the episode.

Each raw or filtered detection result is assigned to the physical instance with the highest intra-frame IoU value for bounding boxes, provided that the IoU value is greater than or equal to 0.5. The bounding box encloses visible masked pixels, and this association does not rely on label consistency. Duplicate detections of the same instance do not increase episode-level recall.

True-target candidates are traced through the actual filtered detection, memory association, and grounding candidate pool. A candidate contributes to recall when it has confirmed target support and no known non-target support. Mixed target/non-target associations are excluded, and updates without an identified physical source are retained as unknown. Visibility, detection, and candidate entry each record whether the event occurred before termination. Detection and candidate recall are conditioned on target-visible episodes.

Intervals use 10,000 instruction-cluster bootstrap resamples with seed 42. Each draw retains all three starts and all visits of a sampled instruction, and each rate is the ratio of the summed numerator to the summed denominator. Sampling results with a denominator of zero will be excluded. An all-positive empirical sample produces a degenerate bootstrap interval and does not eliminate population uncertainty.

Appendix E.2. Observation across region visits

The regional diagnostic includes 24 visits from ten episodes. Each visit requires that the region anchor was the exploration target immediately before arrival and that the region contained a valid target for the instruction. An episode may contribute multiple regions. Target membership follows the frozen geometric

assignment: more than half of the samples on the projected world bounding-box footprint must lie inside the region polygon. “Arrival” refers to the first entry within a range of 2 m from a selected anchor point. The observation window extends from the start of the episode through the first subsequent frame outside the union of the region polygon and the anchor’s 2 m neighborhood, including that exit frame, or through termination if it occurs first. Earlier observations are included and subsequent revisits are excluded. A visit is considered observed when at least one admissible target assigned to that region meets the visibility threshold specified in Appendix E.1. during this time window.

Data availability statement

Frozen evaluation inputs and processed results are provided in the supplementary materials.

Declaration of generative AI and AI-assisted technologies

Generative AI tools were used for language editing and readability improvements. The authors take full responsibility for the manuscript.

Acknowledgments

The authors thank members of the laboratory for helpful discussions and infrastructure support.

Authors’ contribution

Weifan Zhang developed the method and software, conducted the experiments and analysis, and drafted the manuscript. Yunxi Tang revised the manuscript. Yun-Hui Liu and Xiangyu Chu supervised the research and secured resources and funding. All authors reviewed and approved the manuscript.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Bloss R. Mobile hospital robots cure numerous logistic needs. *Ind. Robot* 2011, 38(6):567–571.

[2] Sarrica M, Brondi S, Fortunati L. How many facets does a social robot have? a review of scientific and popular definitions online. *Inf. Technol. People* 2020, 33(1):1–21.

[3] Holland J, Kingston L, McCarthy C, Armstrong E, O’Dwyer P, *et al.* Service robots in the healthcare sector. *Robotics* 2021, 10(1):47.

[4] Anderson P, Wu Q, Teney D, Bruce J, Johnson M, *et al.* Vision-and-language navigation: interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, USA, June 18–23, 2018, pp. 3674–3683.

[5] Krantz J, Wijmans E, Majumdar A, Batra D, Lee S. Beyond the nav-graph: vision-and-language navigation in continuous environments. In *Proceedings of the European Conference on Computer Vision*, Glasgow, UK, August 23–28, 2020, pp. 104–120.

[6] Qi Y, Wu Q, Anderson P, Wang X, Wang WY, *et al.* REVERIE: remote embodied visual referring expression in real indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, USA, June 14–19, 2020, pp. 9982–9991.

[7] Gadre SY, 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*, Vancouver, Canada, June 18–22, 2023, pp. 23171–23181.

[8] Yokoyama N, Ramrakhya R, Das A, Batra D, Ha S. HM3D-OVON: a dataset and benchmark for open-vocabulary object goal navigation. *arXiv* 2024, arXiv:2409.14296.

[9] Ginting MF, Kim SK, Fan DD, Palieri M, Kochenderfer MJ, *et al.* SEEK: semantic reasoning for object goal navigation in real world inspection tasks. *arXiv* 2024, arXiv:2405.09822.

[10] Xie F, Schwertfeger S, Blum H. osmAG-LLM: zero-shot open-vocabulary object navigation via semantic maps and large language models reasoning. *IEEE Robot. Autom. Lett.* 2026, 11(3):2426–2433.

[11] Ku A, Anderson P, Patel R, Ie E, Baldridge J. Room-across-room: multilingual vision-and-language navigation with dense spatiotemporal grounding. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Online, November 16–20, 2020, pp. 4392–4412.

[12] Yokoyama N, Ha S, Batra D, Wang J, Bucher B. VLFM: vision-language frontier maps for zero-shot semantic navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Yokohama, Japan, May 13–17, 2024, pp. 42–48.

[13] Yu B, Kasaei H, Cao M. L3MVN: leveraging large language models for visual target navigation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Detroit, USA, 2023, pp. 3554–3560.

[14] Kuang Y, Lin H, Jiang M. OpenFMNav: towards open-set zero-shot object navigation via vision-language foundation models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, Mexico City, Mexico, June 16–21, 2024, pp. 338–351.

[15] Zhang J, Wang K, Xu R, Zhou G, Hong Y, *et al.* NaVid: video-based VLM plans the next step for vision-and-language navigation. *arXiv* 2024, arXiv:2402.15852.

[16] Yu B, Liu Y, Han L, Kasaei H, Li T, *et al.* Vln-game: vision-language equilibrium search for zero-shot semantic navigation. *IEEE Trans. Robot.* 2026, 42:1824–1839.

[17] Zhang L, Hao X, Tang Y, Fu H, Zheng X, *et al.* *NavA*^{3}: understanding any instruction, navigating anywhere, finding anything. *arXiv* 2025, arXiv:2508.04598.

[18] Zhang W, Li T, Liu Y. MAG-Nav: language-driven object navigation leveraging memory-reserved active grounding. *arXiv* 2025, arXiv:2508.05021.

[19] Huang C, Mees O, Zeng A, Burgard W. Visual language maps for robot navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, London, UK, May 29–June 2, 2023, pp. 10608–10615.

[20] Gu Q, Kuwajerwala A, Morin S, Jatavallabhula KM, Sen B, *et al.* Conceptgraphs: open-vocabulary 3D scene graphs for perception and planning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Yokohama, Japan, May 13–17, 2024, pp. 5021–5028.

[21] Werby A, Huang C, Büchner M, Valada A, Burgard W. Hierarchical open-vocabulary 3D scene graphs for language-grounded robot navigation. *arXiv* 2024, arXiv:2403.17846.

[22] Zhang L, Hao X, Xu Q, Zhang Q, Zhang X, *et al.* MapNav: a novel memory representation via annotated semantic maps for VLM-based vision-and-language navigation. *arXiv* 2025, arXiv:2502.13451.

[23] Zhou K, Zheng K, Pryor C, Shen Y, Jin H, *et al.* ESC: exploration with soft commonsense constraints for zero-shot object navigation. In *Proceedings of the International Conference on Machine Learning*, Honolulu, USA, July 23–29, 2023, pp. 42829–42842.

[24] Ma CY, Lu J, Wu Z, AlRegib G, Kira Z, *et al.* Self-monitoring navigation agent via auxiliary progress estimation. *arXiv* 2019, arXiv:1901.03035.

[25] Zhao C, Qi Y, Wu Q. Mind the gap: improving success rate of vision-and-language navigation by revisiting oracle success routes. *arXiv* 2023, arXiv:2308.03244.

[26] Su HT, Wang TJ, Yeh JF, Sun M, Hsu WH. VLN-NF: feasibility-aware vision-and-language navigation with false-premise instructions. *arXiv* 2026, arXiv:2604.10533.

[27] Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, *et al.* Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, Online, July 18–24, 2021, pp. 8748–8763.

[28] Gadre SY, Ilharco G, Fang A, Hayase J, Smyrnis G, *et al.* DataComp: in search of the next generation of multimodal datasets. In *Proceedings of Advances in Neural Information Processing Systems*, New Orleans, USA, December 10–16, 2023, pp. 27092–27112.

[29] Qwen Team. Qwen2.5: a party of foundation models. 2024. Available: https://qwenlm.github.io/blog/qwen2.5/. (accessed on 9 August 2026).

[30] Yang J, Chen X, Qian S, Madaan N, Iyengar M, *et al.* LLM-Grounder: open-vocabulary 3D visual grounding with large language model as an agent. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Yokohama, Japan, May 13–17, 2024, pp. 7694–7701.

[31] Macenski S, Martín F, White R, Clavero JG. The marathon 2: a navigation system. *arXiv* 2020, arXiv:2003.00368.

[32] Cheng T, Song L, Ge Y, Liu W, Wang X, *et al.* YOLO-World: real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, USA, June 17–21, 2024, pp. 16901–16911.

[33] Zhang C, Han D, Qiao Y, Kim JU, Bae SH, *et al.* Faster segment anything: towards lightweight SAM for mobile applications. *arXiv* 2023, arXiv:2306.14289.

[34] Kirillov A, Mintun E, Ravi N, Mao H, Rolland C, *et al.* Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Paris, France, October 2–6, 2023, pp. 4015–4026.