Robotics: Science and Systems 2026
Sydney, Australia, July 13–17, 2026

OpenFrontier: General Navigation with Visual-Language Grounded Frontiers

Esteban Padilla-Cerdio[*,1], Boyang Sun[*,1], Marc Pollefeys[1,2], Hermann Blum[3]
[1]ETH Zurich [2]Microsoft Spatial AI Lab [3]University of Bonn
[*]Equal Contribution
https://boysun045.github.io/OpenFrontier-Project/

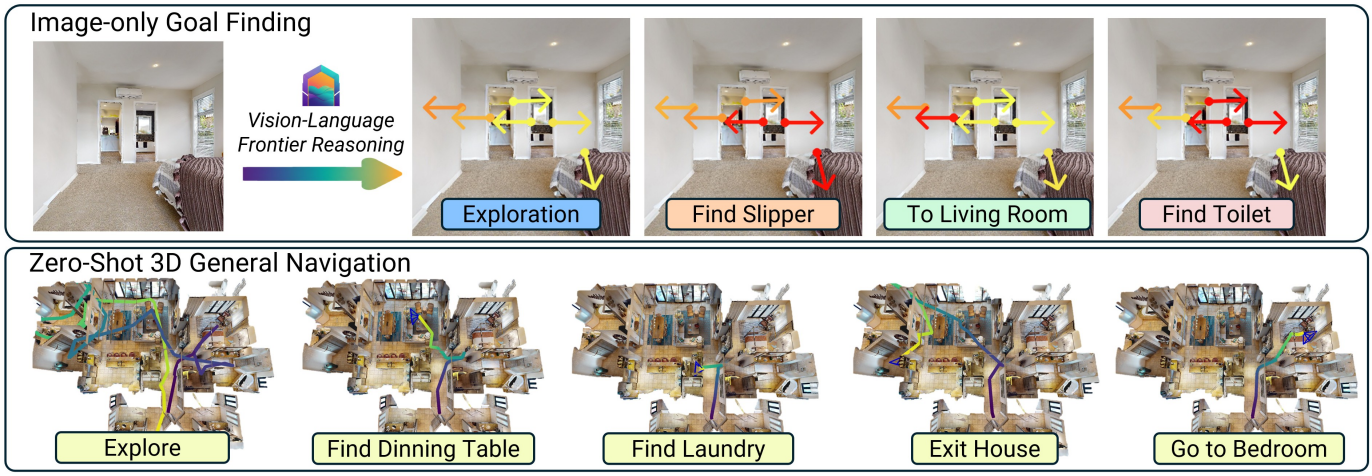

Fig. 1: We present **OpenFrontier**. **Top:** OpenFrontier detects and semantically evaluates visual frontiers directly in the image space, enabling language-conditioned goal reasoning without dense 3D reconstruction. Arrow color encodes goal relevance, from red (high) to yellow (low). **Bottom:** Visual frontiers from individual observations are sequentially grounded into the 3D metric space, driving long-horizon, natural-language–conditioned navigation across diverse goals in a fully zero-shot manner, without task-specific fine-tuning.

***Abstract*—Open-world navigation requires robots to make decisions in complex everyday environments while adapting to flexible task requirements. Conventional navigation approaches often rely on dense 3D reconstruction and hand-crafted goal metrics, which limits their generalization across tasks and environments. Recent advances in vision–language navigation (VLN) and vision–language–action (VLA) models enable end-to-end policies conditioned on natural language, but typically require interactive training, large-scale data collection, or task-specific fine-tuning with a mobile agent. We formulate navigation as a sparse subgoal identification and reaching problem and observe that providing visual anchoring targets for high-level semantic priors enables highly efficient goal-conditioned navigation. Based on this insight, we select visual frontiers as semantic anchors and propose *OpenFrontier*, a navigation framework that requires no task-specific training or fine-tuning and seamlessly integrates diverse vision–language prior models. OpenFrontier enables efficient navigation with a lightweight system design, without dense 3D semantic mapping, task-specific policy training, or model fine-tuning. We evaluate OpenFrontier across multiple navigation benchmarks and demonstrate strong zero-shot performance, as well as effective real-world deployment on a mobile robot.**

I. Introduction

Navigation in open-world environments requires robots to reason over both high-level semantics and low-level geometry while operating under partial observability. Classical object-goal navigation methods typically rely on dense 3D reconstruction followed by object detection and localization in a global map [54, 2, 19], which demands accurate mapping and struggles with cluttered scenes or small, ambiguous objects. Previous learning-based approaches attempt to jointly solve high-level decision making and low-level control through reinforcement learning [6, 7, 52, 37, 53], but are commonly limited to closed-set object categories and show poor generalization beyond training distributions [14]. More recently, large language models (LLMs) and vision-language models (VLMs) have been explored as sources of semantic priors for navigation, either via end-to-end vision-language-action training [42, 55, 8] or by directly prompting models for action cues [26, 15]. However, such approaches often require large-scale interactive training, suffer from real-time constraints, or face fundamental difficulties in grounding high-level semantic reasoning into metric navigation decisions.

While stronger semantic priors, larger datasets, and more versatile policies continue to advance language-conditioned navigation, an equally important challenge lies in identifying an *effective interface* that grounds semantic reasoning into physically meaningful navigation decisions. In this work, we revisit the general robot navigation task and propose **OpenFrontier**, a zero-shot navigation framework that uses visual navigation frontiers as sparse, interpretable, and physically grounded semantic anchors. Frontiers correspond to navigable regions in metric space and naturally encode exploration while remaining directly actionable, making them an ideal substrate for language-conditioned reasoning. OpenFrontier presents detected frontiers to a VLM using a set-of-marks formulation

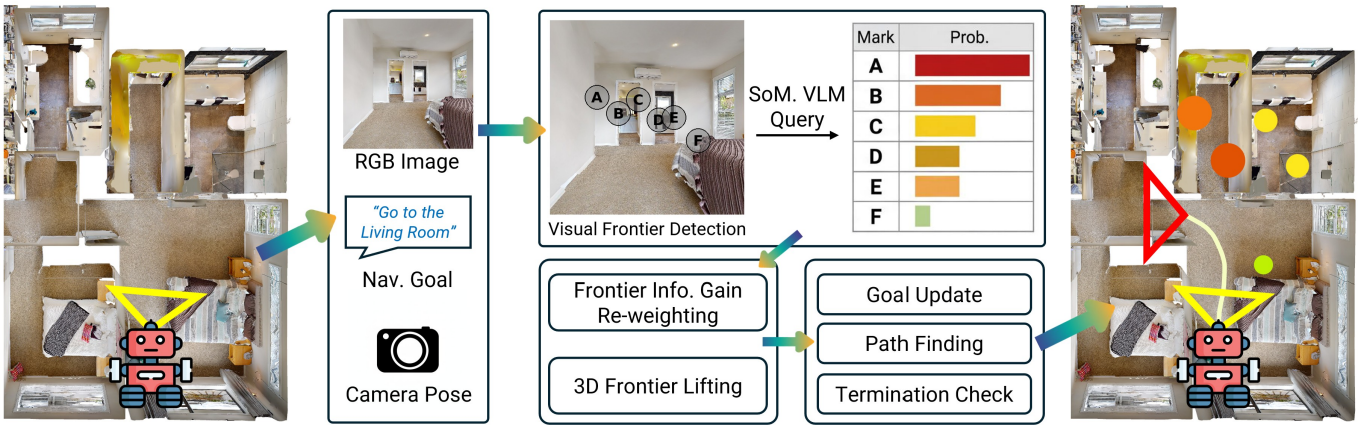

Fig. 2: **System Overview**. Given a posed RGB observation and a natural-language navigation goal, OpenFrontier detects visual frontiers in the image and directly queries a vision–language model to evaluate their relevance (probability to the target) using in-image context. The resulting frontiers are then lifted into the 3D metric space with the updated information gain as goal-conditioned candidates and globally managed to update navigation targets, perform path planning, and determine termination.

[48] over the individual RGB observation, enabling the model to assign semantic relevance and priority to each candidate. It maintains a global updating mechanism to keep track of sparse frontiers. The robot then navigates by iteratively clearing frontiers according to these grounded priors, balancing exploration and exploitation while continuously assessing goal satisfaction.

OpenFrontier requires no dense semantic mapping, no task-specific policy training, and no fine-tuning, is agnostic to the choice of VLM for reasoning, and transfers seamlessly to unseen environments, open-set goals and real-world scenarios. We evaluate OpenFrontier across multiple indoor navigation benchmarks, including both closed-set and open-set object-goal navigation tasks, and demonstrate strong zero-shot generalization compared to prior methods. Despite its simplicity, OpenFrontier shows competitive performance against recent baselines while requiring the minimum mapping or fine-tuning effort. Finally, we deploy our system on a mobile legged robot and demonstrate robust real-world navigation in a large indoor environment. As a summary, our main contributions are:

- We propose **OpenFrontier**, a navigation framework that uses *visual navigation frontiers* as an interface for grounding vision–language priors into actionable navigation goals for language-conditioned object-goal navigation.
- We introduce a visual-target reasoning formulation that evaluates candidate frontiers using vision–language models and integrates semantic relevance with exploration-driven information gain, without requiring dense 3D semantic maps or task-specific policy training.
- We demonstrate strong zero-shot performance across multiple navigation benchmarks and validate the approach in real-world robotic deployment.

II. Related Work

A. Map-based Exploration and Navigation

Building explicit maps for navigation is one of the most classical paradigms in mobile robotics [35, 1]. A wide range of map representations have been investigated. Among them, frontier-based exploration has long been a core approach, where frontiers represent the boundary between known and unknown space and serve as natural candidates for exploration-driven motion [45]. Classical methods extract frontiers from volumetric occupancy maps and select navigation goals based on hand-crafted heuristics such as distance or information gain [20, 36, 10, 17]. While effective, these approaches depend on reliable mapping and typically operate purely on geometric criteria.

Recent works extend frontier-based navigation by incorporating learning-based perception and semantic reasoning. FrontierNet [34] demonstrates that frontiers can be detected and evaluated directly from RGB images without constructing dense maps, enabling lightweight exploration driven by learned visual cues. Vision-Language Frontier Maps (VLFM) [50] integrate vision-language priors into frontier-based navigation by constructing dense semantic frontier maps, allowing semantic relevance to guide frontier selection.

Semantic-enabled frontier representations are particularly natural for object-goal navigation. Object-goal navigation has been widely studied in the context of embodied AI, where agents are tasked with navigating to objects specified by category labels or semantic descriptions. Several benchmarks have been proposed, including HM3D ObjectNav [43, 44] and OVON [51], with OVON explicitly targeting open-vocabulary settings, thereby increasing the complexity of required semantic representations. GOAT-Bench [21] further evaluates open-world object navigation with diverse object categories and environments.

Early approaches rely on dense 3D reconstruction and semantic mapping pipelines, combining object detection with explicit spatial representations to guide navigation [54, 6]. BeliefMapNav [59] constructs voxel-based belief maps to represent object likelihoods in 3D space, enabling zero-shot navigation at the cost of maintaining dense semantic state. Similarly, GOAT [5] maintains a dense semantic map from which global and local policies derive actions. Another line of work uses map representations as inputs to reinforcement learning models to learn spatial information and navigation policies [40, 7, 6, 29].

Another line of work focuses on zero-shot navigation, par-

ticularly enabled by recent advances in large-scale foundation models. CoWs [13] guides frontier-based exploration using CLIP-based semantic similarity. OpenFMNav [22] leverages vision-language foundation models to support open-set object-goal navigation, while History-Augmented VLM [16] incorporates memory mechanisms to improve zero-shot exploration. ForesightNav [31] constructs CLIP-based semantic maps and trains a model to predict vision-language features for unexplored regions to guide navigation decisions. IPPON [28] builds volumetric occupancy maps augmented with language embeddings during exploration and uses them for open-vocabulary goal reasoning. UniGoal [49] proposes a universal framework for zero-shot goal-oriented navigation by combining dense mapping, scene graph reasoning, and both LLMs and VLMs. Despite the different approaches and design choices, these methods often require persistent semantic maps, scene graphs, or learned belief representations, and some further rely on interactive learning to acquire navigation knowledge.

B. Vision-Language Navigation and Action Models

The emergence of large language models (LLMs) and vision-language models (VLMs) has led to significant progress in vision-language navigation (VLN) and vision-language-action (VLA) systems [57]. Many approaches train end-to-end policies conditioned on language instructions using reinforcement learning or imitation learning, often requiring large-scale interactive data collection and task-specific fine-tuning [56, 53, 38]. While effective in constrained settings, these methods typically struggle to generalize beyond training distributions and incur high computational cost at inference time.

Several recent works explore general navigation by directly leveraging pretrained foundation models. InstructNav [25] relies on large-scale, closed-source vision-language and language models to infer navigation actions directly from visual observations and instructions. NAVILA [8] fine-tunes vision-language models using datasets extracted from real-world videos and maps them to robot action spaces through a hierarchical structure. StreamingVLN [38] designs a sliding-window key–value cache management architecture that enables navigation through multi-round dialogue. Uni-NaVid [55] further fine-tunes vision-language navigation models on specific navigation benchmarks to achieve strong performance.

These approaches generally require large-scale datasets for fine-tuning and often rely on frequent model queries, such as direct action inference at every step. Such designs can be computationally expensive and difficult to ground in metric navigation space. Moreover, despite the significant training resources involved, many VLN systems primarily focus on target reaching or instruction following, which does not fully exploit the rich spatial and semantic information inherently captured by large-scale vision-language models.

III. Method

OpenFrontier adopts a minimal but effective principle: *detect and evaluate navigation target directly in the 2D image domain whenever possible*. Following this principle, both goal identification and goal evaluation are carried out directly on RGB observations $\mathbf{I}_t \in \mathbb{R}^{H\times W\times 3}$, together with the associated camera extrinsic parameters $\mathbf{T}_t \in SE(3)$ and the camera intrinsic matrix $\mathbf{K} \in \mathbb{R}^{3\times 3}$ at time step t. OpenFrontier outputs a goal pose, which can be further translated into control actions for the robot. Given a natural-language navigation goal, OpenFrontier processes the current observation $\mathbf{I}_t$ to identify a set of *visual frontiers*. Each frontier is first represented as $\mathbf{F} = \{\mathbf{X}, g\}$, where $\mathbf{X} = \{\mathbf{p}, \mathbf{q}\}$ denotes the 3D position and orientation of the frontier in the world frame, and g denotes its information gain. Importantly, no dense 3D semantic reconstruction is required for goal identification. The selected frontier is grounded in 3D space and used to generate a goal pose, which is then passed to a low-level motion planner for execution. The system iteratively updates frontier information gain and replans as new observations arrive, until the navigation goal is reached. An overview of the full system of OpenFrontier is shown in Fig. 2.

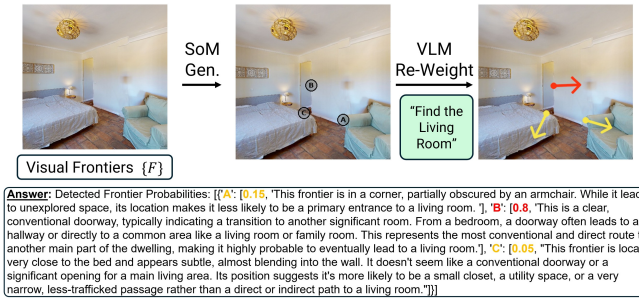

Fig. 3: The detected visual frontiers are jointly queried with the corresponding RGB image using a set-of-marks prompting strategy. Each frontier is marked in the image, enabling the VLM to evaluate its relevance to the given navigation instruction within the local visual context. The resulting relevance probabilities are used to re-weight its exploration-driven information gain from FrontierNet, effectively integrating task-specific semantic priors with exploration.

A. Visual-Frontier Identification

A large body of robot navigation follows the convention of using geometry frontiers [45, 27, 20, 50, 58] extracted from volumetric maps as goal candidates for robot motion. Frontiers provide an effective and natural representation for encouraging exploration toward uncertain regions of the environment. Inspired by FrontierNet [34], we observe that frontiers can be detected and evaluated directly from a single 2D image, which we refer to as *visual frontiers*, without relying on dense 3D mapping or separating detection from evaluation. We use *visual frontier* and *frontier cluster* interchangeably in the remainder of the paper.

Following FrontierNet, we apply a similar pipeline to a keyframe RGB observation $\mathbf{I}_t$ and, after post-processing, returns a set of frontier clusters $\mathbf{F}_i$, $i \in \mathbb{N}$. Each frontier cluster is represented in the image domain and then back-projected into 3D space to obtain its pose. The frontier identification pipeline also provides an information gain estimate $\hat{g}_i$, which captures a pure exploration prior by predicting the volume of unknown space associated with the frontier [34, 3, 10]. To incorporate task-specific semantic priors, we directly ground a VLM onto the frontier clusters in the image $\mathbf{I}_t$. Given a natural-language goal $\mathbf{L}$, the VLM assigns each frontier cluster a probability

$p_i \in [0, 1]$ indicating the likelihood that navigating toward that frontier will lead to accomplishing the goal. The final utility of each frontier is computed as:

$$g_i = p_i \cdot \hat{g}_i, \tag{1}$$

$$p_i = \mathrm{VLM}(\mathbf{I}_t, \mathbf{F}_i, \mathbf{L}), \tag{2}$$

where $\hat{g}_i$ represents exploration-driven information gain and p_i injects task-conditioned semantic guidance, which is obtained via the set-of-marks VLM query described below. This formulation naturally balances general exploration with goal-directed exploitation.

Visual frontiers are detected and clustered based on visual cues in the input image, resulting in a set of frontier clusters localized by their centroid pixel coordinates in the image plane. These 2D locations provide a natural and explicit mechanism for guiding the attention of a vision-language model. To this end, we adopt a set-of-marks querying strategy that encourages the VLM to evaluate the correspondence between each frontier cluster and the language-conditioned goal using the surrounding image context. As illustrated in Fig. 3, we overlay a visual marker [48, 32] at the 2D centroid of each frontier cluster on the RGB observation. The marked image, together with the natural-language goal, is jointly fed into the VLM using an additional prompt that instructs the model to output a probability score for each marker. An example prompt template is provided in the Appendix. This querying strategy enables the VLM to jointly reason about all candidate frontiers within a single forward pass, while leveraging fine-grained 2D visual context directly tied to actionable navigation locations. Moreover, by anchoring reasoning in the image plane, this formulation avoids requiring the VLM to perform explicit 3D spatial reasoning, where current models are known to be less reliable [12].

B. Frontier Management

Long-horizon navigation requires maintaining and updating goals at a global level. We maintain a set of active frontier goals and manage their selection and update throughout the navigation process. Similar to classical exploration strategies, the utility of each frontier is periodically updated based on the current robot state. Specifically, for each frontier i, we compute a global utility

$$u_i = \frac{g_i}{\|\mathbf{p}_r - \mathbf{p}_i\|}, \tag{3}$$

where $\mathbf{p}_r$ denotes the current position of the robot. This formulation favors frontiers that are both semantically relevant and spatially efficient to reach. The frontier with the highest utility is selected as the next navigation goal and passed to a low-level planner. The choice of low-level planner is flexible. When no global map is available, a map-free PointGoal navigation policy [39, 46, 41, 47, 23] can be used. Alternatively, if a 3D volumetric representation is maintained, a map-based collision-free planner can be applied without modifying the frontier management logic.

Algorithm 1 Global Target Management

Require: Global frontier set $\mathcal{F}$, cleared set $\mathcal{C}$, proposals $\tilde{\mathcal{F}}_t$
Require: Robot position $\mathbf{p}_r$, PointNav π, goal $\mathbf{L}$
Require: Thresholds $\tau_{\text{merge}}, r_{\text{clear}}, r_{\text{near}}, r_{\text{goal}}$
Require: Progress parameters $(T_{\text{stall}}, \epsilon_{\text{stall}})$
$\texttt{hasTarget} \leftarrow 0$, $\mathbf{x}^* \leftarrow \mathbf{0}$
$\texttt{stall} \leftarrow 0$, $d_{\text{prev}} \leftarrow 10^9$
while true **do**
  $\mathcal{F} \leftarrow \textsc{MergeUpdate}(\mathcal{F}, \tilde{\mathcal{F}}_t, \tau_{\text{merge}})$
  $\mathcal{F} \leftarrow \textsc{Prune}(\mathcal{F}, \mathcal{C}, r_{\text{clear}})$
  $\mathcal{F} \leftarrow \textsc{InsertViewpointIfAny}(\mathcal{F}, \mathbf{L}, r_{\text{near}})$
  $(\texttt{ok}, \mathbf{F}^\star) \leftarrow \textsc{SelectBest}(\mathcal{F}, \mathbf{p}_r)$
  if $\texttt{ok} = 0$ **then**
    continue
  end if
  $(\mathbf{p}_r, d) \leftarrow \textsc{StepTo}(\pi, \mathbf{F}^\star)$
  $(\texttt{drop}, \texttt{stall}, d_{\text{prev}}) \leftarrow \textsc{HandleProgress}(d, d_{\text{prev}}, \texttt{stall}, T_{\text{stall}}, \epsilon_{\text{stall}})$
  if $\texttt{drop} = 1$ **then**
    $\mathcal{C} \leftarrow \mathcal{C} \cup \{\textsc{Pos}(\mathbf{F}^\star)\}$
    $\mathcal{F} \leftarrow \mathcal{F} \setminus \{\mathbf{F}^\star\}$
    continue
  end if
  if $d < r_{\text{near}}$ **then**
    $\mathcal{C} \leftarrow \mathcal{C} \cup \{\textsc{Pos}(\mathbf{F}^\star)\}$
    $\mathcal{F} \leftarrow \mathcal{F} \setminus \{\mathbf{F}^\star\}$
  end if
  $(\texttt{hasTarget}, \mathbf{x}^*) \leftarrow \textsc{VerifyAndSetTarget}(\mathbf{F}^\star, \mathbf{L}, d, r_{\text{near}})$
  if $\texttt{hasTarget} = 1$ **then**
    $(\mathbf{p}_r, \_) \leftarrow \textsc{StepToPos}(\pi, \mathbf{x}^*)$
    if $\|\mathbf{p}_r - \mathbf{x}^*\| < r_{\text{goal}}$ **then**
      break
    end if
  end if
end while

While the robot navigates through the environment, the RGB observations are processed by an open-vocabulary segmentation model [4] to generate segmentation masks corresponding to the target. When a valid mask is detected, depth and camera pose are used to estimate the 3D centroid of the object. A new frontier with high (effectively infinite) utility is then instantiated at a fixed offset from the object, oriented to face it directly. This viewpoint frontier is inserted into the active frontier set and prioritized by the frontier manager, encouraging the robot to move to a configuration that provides direct visual access to the object.

After the robot reaches the viewpoint frontier, the corresponding RGB observation is queried by the same vision–language model to verify the presence of the target. If the object is not confirmed, the viewpoint frontier is removed and the object hypothesis is discarded. If the object is confirmed, the estimated centroid is designated as the final target position, and the robot is instructed to approach it as closely as possi-

Method	Zero-shot	Spatial Representation	HM3D Val		MP3D Val		OVON Val Unseen	
			SR (%)↑	SPL (%)↑	SR (%)↑	SPL (%)↑	SR (%)↑	SPL (%)↑
History-Aug. VLM [16]	✓	2D semantic & textual history	46.0	24.8	–	–	–	–
OpenFMNav [22]	✓	dense 2D semantic	52.5	24.1	37.2	15.7	–	–
InstructNav [25]	✓	dense 2D value map	58.0	20.9	–	–	–	–
BeliefMapNav [59]	✓	3D voxel belief map	61.4	30.6	37.3	17.6	–	–
VLFM [50]	✓	dense 2D value map	52.5	30.4	36.4	17.5	35.2	19.6
DAgRL+OD [51]	×	none	–	–	–	–	37.1	19.9
UniGoal [49]	✓	3D scene graph	54.5	25.1	**41.0**	16.4	–	–
Uni-NaVid [55]	×	none	73.7	**37.1**	–	–	**39.5**	19.8
OpenFrontier (Ours)	✓	sparse visual frontiers	**77.3**	35.6	40.7	**17.8**	39.0	**20.1**

TABLE I: **Performance Comparison** on Habitat ObjectNav benchmarks. *Zero-shot* indicates no fine-tuning on navigation tasks. *Spatial Representation* denotes the primary global representation each method uses for goal reasoning and planning.

ble. Algorithm 1 summarizes the global frontier management process from goal selection to termination. For simplicity, the algorithm presents a compact formulation; additional implementation details for the helper functions are provided in the Appendix.

C. Flexible Framework Design and Open-Context Query

OpenFrontier is structured around a clear separation between visual-frontier perception/reasoning and global goal management. This separation is enabled by the use of navigation frontiers as the interface between the two components. We construct frontiers as a lightweight and interpretable abstraction that bridges visual-frontier reasoning with global decision making. As a result, each component of the pipeline can be modified or replaced independently, providing a high degree of flexibility in system design.

Importantly, OpenFrontier does not assume any pre-training, fine-tuning, or in-context learning of the vision-language model. The VLM is queried in an open-context manner and can be replaced directly without modifying the rest of the system. Likewise, the framework does not require dense 3D reconstruction or semantic mapping. However, when available, simple geometric information can be incorporated to improve robustness. In our implementation, an optional lightweight volumetric map built from monocular depth priors is used only for basic filtering: occupied space is used to discard unreachable or unsafe frontiers, while free space is used to suppress frontiers whose exploration information gain has been exhausted during navigation. These geometric cues operate asynchronously and are not required for the reasoning steps, but consistently improve navigation performance in practice.

IV. Experiment and Results

A. Experimental Setup

We evaluate OpenFrontier along three dimensions: (1) performance on object-goal navigation tasks, (2) flexibility and generalization across environments, VLMs, and task settings, and (3) robustness to the sim-to-real gap. To measure the overall quantitative performance, we benchmark OpenFrontier on three object-goal navigation datasets: HM3D ObjectNav, MP3D ObjectNav [43, 44], and OVON [51]. All benchmarks

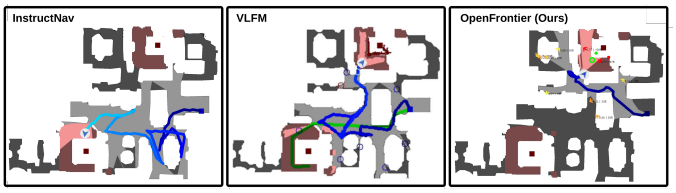

Fig. 4: **Navigation Results** across two representative baseline methods and OpenFrontier (HM3D: `5cdEh9F2hJL`, goal: bed). The red square and shaded region indicate the ground-truth target location and its success region. OpenFrontier shows stronger decision-making ability, e.g. at multi-choice intersections, navigating directly toward the target region while avoiding redundant exploration of irrelevant areas.

follow the Habitat Navigation Challenge formulation. HM3D and MP3D consider closed-vocabulary object goals, while OVON extends the task to open-vocabulary settings with more diverse language descriptions.

We implement OpenFrontier within the Habitat framework and evaluate navigation performance using standard metrics: Success Rate (SR, %) and Success weighted by Path Length (SPL, %). We compare against a set of baseline methods spanning dense 3D mapping approaches, belief-based planners, and recent vision-language navigation methods. For object-goal benchmark experiments, we use Gemini-2.5-flash as the VLM model. Low-level navigation is executed using a DD-PPO point-goal policy [39] with pretrained weights from VLFM [50], which matches the Habitat action space. Visual frontier detection is performed using FrontierNet with pretrained weights from [34]. We use the same set of system parameters across all benchmarks. Empirically, we run frontier detection and reasoning every six steps. All experiments are conducted on a machine equipped with a single RTX 4090 GPU (24GB). Details about parameter settings can be found in the Appendix.

B. ObjectNav Benchmark Evaluation

OpenFrontier achieves competitive performance across all three benchmarks compared to state-of-the-art baselines. Quantitative results are summarized in Table I. As the system only processes keyframes and operates on a sparse set of frontiers, it avoids high-frequency inference such as step-by-step mapping updates used by History-Augmented VLM, or direct action inference as in Uni-NaVid. Under a sin-

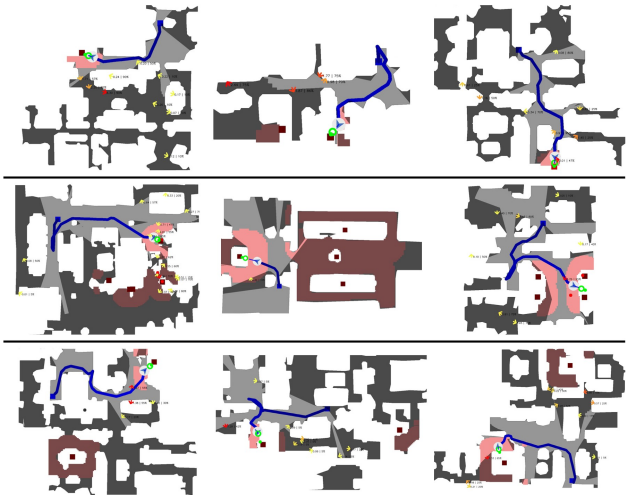

Fig. 5: **Additional Navigation Examples**. **Top:** OVON scenes with goals (left to right) *refrigerator*, *picture*, and *dishwasher*. **Middle:** MP3D scenes with goals *stool*, *table*, and *cushion*. **Bottom:** HM3D scenes with goals *sofa*, *toilet*, and *bed*. All experiments are conducted using the same system configuration and parameters across datasets.

gle, uniform configuration across all scenes, OpenFrontier outperforms most baselines on all three datasets. Compared to the strongest competing methods, OpenFrontier is 1.5% lower in SPL on HM3D validation and 0.5% lower in SR on OVON than Uni-NaVid, and 0.3% lower in SR than UniGoal. Notably, Uni-NaVid is a VLN model fine-tuned on these benchmarks and their action spaces, while UniGoal relies on dense mapping, scene graph construction, and the combined use of both a LLM and a VLM model. Despite these additional assumptions, UniGoal performs more than 20% lower in SR than OpenFrontier on HM3D. Overall, OpenFrontier adopts a minimal design while maintaining stable and competitive performance across benchmarks. Additionally, we provide a failure case decomposition in Fig. 10, with more detailed analysis in the Appendix.

To further inspect qualitative behavior beyond aggregate numbers, we re-ran two representative baselines, InstructNav and VLFM, under the same evaluation setup on the entire HM3D validation set. InstructNav uses both advanced VLM and LLM models, while VLFM adopts a frontier-based strategy combined with dense mapping and feature integration. As shown in Fig. 4, OpenFrontier demonstrates higher navigation efficiency in complex scenes with multiple branching corridors, consistent with the quantitative results. Fig. 5 presents additional qualitative examples across all three evaluation datasets. In these scenarios, OpenFrontier demonstrates effective spatial and semantic reasoning, allowing the agent to bias exploration toward goal-relevant regions and reach the target efficiently. Additional qualitative results are provided in the Appendix.

Vision-language models are used in OpenFrontier for two purposes: frontier utility reweighting and goal verification. Goal verification is triggered only when a candidate target is detected by SAM3 [4]. Neither component requires task-specific training or fine-tuning, allowing the VLM to be easily replaced with alternative models. To evaluate this flexibility,

Method (VLM)	**SR (%)↑**	**SPL (%)↑**
UniGoal (LLaVA 1.6)	54.5	25.1
History-Aug. VLM (LLaVA 1.6)	46.0	24.8
InstructNav (Gemini-2.5-flash)	52.0	22.0
OpenFrontier (LLaVA 1.6)	74.2	35.3
OpenFrontier (InternVL3_5-8B)	74.5	34.5
OpenFrontier (Gemma-3-4b-it)	76.9	33.7
OpenFrontier (Gemini-2.5-flash)	**77.3**	**35.6**

TABLE II: **VLM Sensitivity and Cross-Method Comparison** on HM3D. The swapped VLM here refers to the frontier-reweighting reasoning model; the InstructNav (Gemini-2.5-flash) entry is a re-run with the substituted VLM and therefore differs from the original number in Table I. Within OpenFrontier, VLM-induced variance is small ($\leq$ 3% SR), showing that our framework offers a stable and universal interface for leveraging VLMs. At a matched VLM, OpenFrontier substantially outperforms baselines, indicating that the gain is driven by the visual-frontier design rather than by VLM strength.

we run the full HM3D evaluation using several other VLMs. As shown in Table II, replacing Gemini-2.5-flash [9] with other publicly available or open-source models, including the older LLaVA 1.6 [24], results in only a marginal performance decrease ($\leq$ 3% SR) while maintaining strong compatibility with our framework. To further isolate the contribution of the system design from that of the VLM, the same table also reports a cross-method comparison in which baselines and OpenFrontier are evaluated using the same VLM backbone. At a matched VLM, OpenFrontier substantially outperforms UniGoal, History-Aug. VLM, and InstructNav, indicating that our gains are not attributable to VLM strength alone but to the visual-frontier representation and the system design around it. We further evaluate a broader set of VLMs on real-world images to compare their image-space reasoning ability with frontier selection capability. The qualitative comparison in Fig. 7 illustrates the differences in reasoning behavior across models. Despite differences in the exact probability values generated by different VLMs, all tested models provide reasonable probability estimates. These results suggest that for OpenFrontier stronger VLMs naturally bring improved performance with just a simple switch, however it is also generally robust to the choice of different VLM.

Beyond category-level object navigation, we also evaluate OpenFrontier under more complex language conditions that include spatial or appearance context, which the quantitative protocol does not easily capture. Fig. 6 shows an example comparing the goals *"Plant"* and *"Plant in Bathroom"*. The resulting trajectories and frontier probability assignments differ accordingly, with OpenFrontier prioritizing bathroom-related frontiers under the spatially conditioned query. Additional examples are provided in the Appendix. These results demonstrate that OpenFrontier is able to exploit contextual reasoning from VLMs and successfully ground it onto frontiers to support goal specifications with more complex context.

C. Real-World Deployments

Finally, we evaluate OpenFrontier in real-world settings. The results in Fig. 7 already demonstrate the visual frontier identification module on real indoor images with human-specified language targets, verifying its generalization beyond simulation. We further deploy OpenFrontier on a legged robot

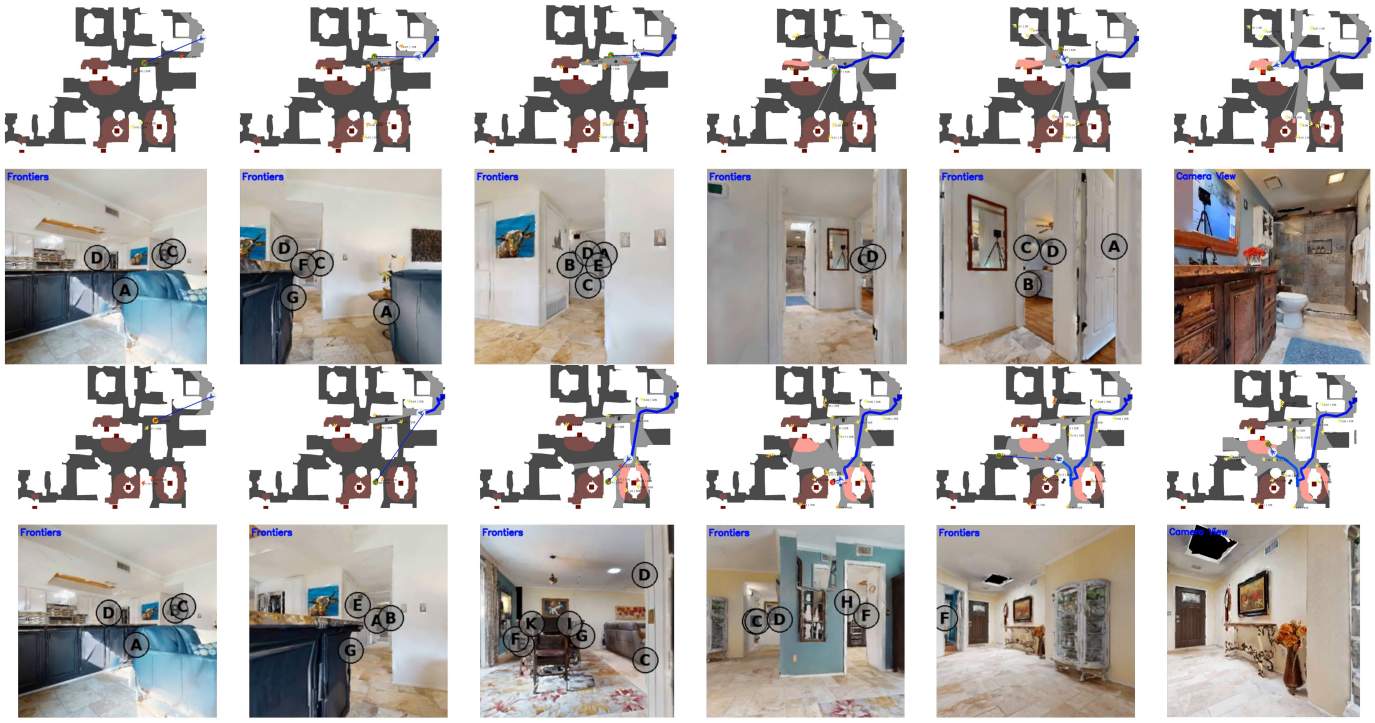

Fig. 6: **OpenFrontier Navigation Example with Different Goal Contexts** (HM3D: `5cdEh9F2hJL`). Top: target is *"plant in the bathroom."* Bottom: target is *"plant."* The robot is initialized at the same starting location in both runs. From left to right, we show selected frames along the navigation trajectory together with the corresponding image observations overlaid with detected frontiers. The rightmost image is the final observation that triggers termination once the target enters the field of view. Despite observing similar frontier locations, OpenFrontier assigns different relevance probabilities depending on the goal context. As a result, the top trajectory prioritizes regions likely associated with the bathroom, while the bottom trajectory moves toward the living room, which also commonly contains plants.

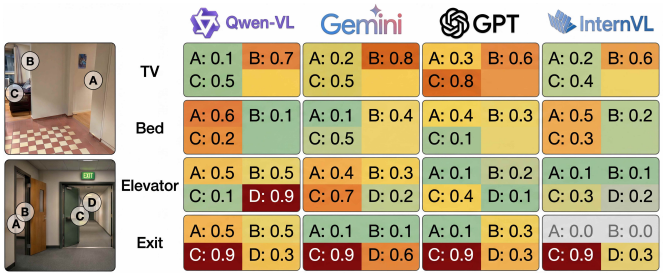

Fig. 7: **Visual Frontier Reasoning Examples.** Four different VLMs are evaluated for frontier probability estimation (without normalization) using the set-of-marks querying strategy (left to right: Qwen3-VL, Gemini-2.5, GPT-4o, InternVL-3.5). All models operate on the same real-world image with the same prompt after frontier detection.

(Boston Dynamics Spot) and conduct navigation tasks in a large-scale indoor environment. The system is integrated within ROS, with RGB observations provided by the camera mounted on the robot arm end-effector. The camera pose is obtained from the robot's onboard vision–inertial odometry. The robot is started at different initial locations and given different language inputs for the target, without any prior knowledge of the scene layout. Fig. 8 shows an example in which the robot autonomously navigates to a fire extinguisher in a large indoor environment without any human intervention. We further conduct a small-scale quantitative evaluation in the indoor environment, covering 5 different target objects with 10 navigation trials in total, and obtain a 70% success rate.

V. Insights

OpenFrontier occupies an intermediate design point between recent advanced navigation approaches that either rely on rich 3D map representations or require training VLN/VLA models for navigation settings. Despite its zero-shot setup, OpenFrontier demonstrates stable and competitive performance across benchmarks and transfers effectively from simulation to real-world deployment. In this section, we discuss several insights that help explain these observations.

A. Why the Design Is Effective

A key observation from OpenFrontier is that *a lightweight navigation pipeline can be highly effective when appropriate representations and system interfaces are used*. We attribute the effectiveness of OpenFrontier to three aspects of the visual-frontier abstraction.

(1) Detection and reasoning operate in a shared image space. Because visual frontiers are detected and evaluated on the same RGB observation, the VLM sees the full visual context behind each candidate without any lossy intermediate representation, and is asked to perform a task it is much closer to its training setting: 2D visual reasoning rather than 3D spatial reasoning over multi-view inputs or abstract scene-level representations. This yields three direct consequences. *(a) Direct Adaptation.* No fine-tuning is needed for the VLMs to give high-quality reasoning results. *(b) Robustness to the VLM backbone.* Swapping the reweighting VLM across four backbones, including the older LLaVA 1.6, induces only $\leq 3\%$

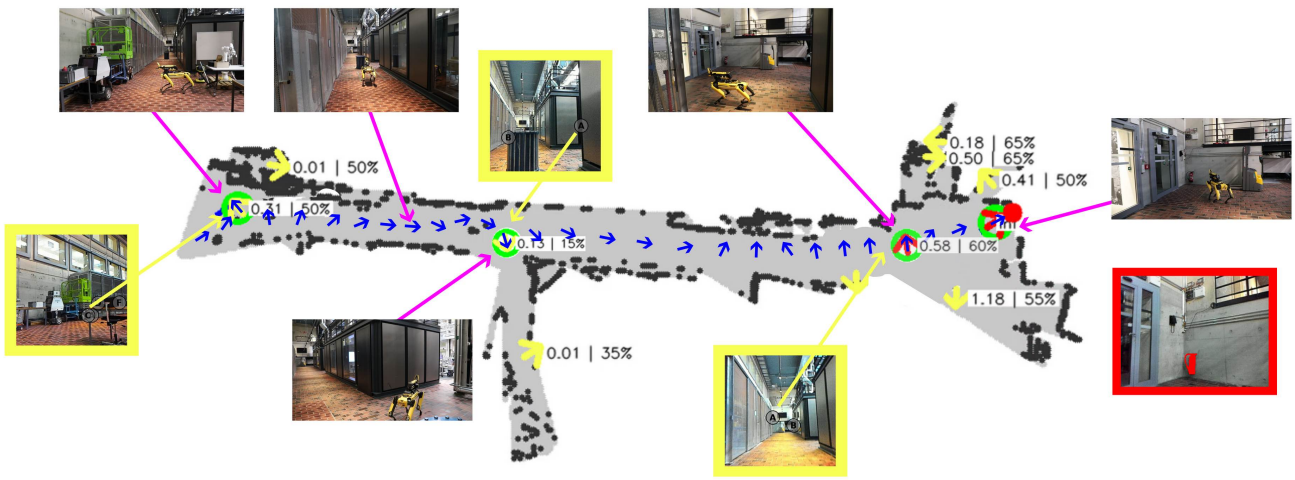

Fig. 8: **Real-world Deployment**. Example of a deployment of OpenFrontier queried to find a fire extinguisher. The blue arrows illustrate the path of the robot through the environment. The yellow boxes mark the VLM prompt images for different keyframes, connected to some key 3D frontiers marked during navigation on the map. The red box shows the detection of the target object that signals successful task completion.

SR variation (Table II). Combined with the absence of any fine-tuning or in-context learning, this gives OpenFrontier a truly zero-shot operating mode in which most of the inductive load is carried by the representation rather than by the VLM. The same image-space framing also explains our advantage at matched VLM backbones (Table II). *(c) Natural extensibility.* The query interface is straightforward to extend with additional context. As a proof-of-concept, a history-augmented variant simply appends a small set of recent keyframes to the VLM query (Fig. 9), so the model can refine a frontier's relevance in light of what has already been observed, while frontier detection, global management, and the rest of the pipeline remain unchanged.

(2) The utility merges complementary exploration and exploitation priors. The frontier utility $g_i = p_i\,\hat{g}_i$ combines a geometric exploration prior $\hat{g}_i$ from FrontierNet (the expected unobserved volume revealed by visiting the frontier) with a semantic exploitation prior p_i from the VLM (high-level scene and object knowledge conditioned on the goal). These priors are genuinely complementary: FrontierNet has no notion of task semantics, while FrontierNet outperforms current VLMs on estimating unobserved 3D volume. Combining them in a single utility lets each compensate for the other's weakness and naturally balances coverage of unexplored regions with goal-directed bias.

(3) The system has a sparse, transferable, and hierarchically aligned interface. Detected in 2D, visual frontiers are then grounded in 3D, where they directly serve as actionable navigation targets. Their sparsity makes them robust to noisy sensor input, and inexpensive to maintain and update over long horizons, providing a form of global memory without dense semantic state. The same abstraction also aligns naturally with the standard high-level / low-level decomposition of robot navigation: frontiers carry the high-level semantic decisions while a low-level controller handles execution, letting each component operate at the level of abstraction where it is most effective. Future work may explore tighter but principled integration between these two levels, enabling mutual feedback while preserving the benefits of modularity.

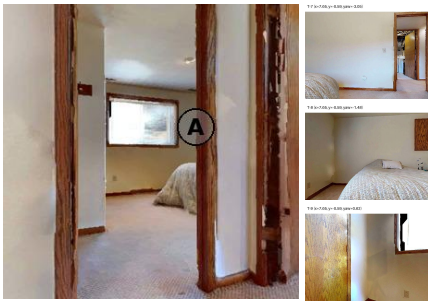

Fig. 9: **History-Augmented Frontier Reasoning**. Left: current observation with set-of-marks frontier annotations. Right: three prior keyframes from the same room, passed as additional context. Given television as the navigation target, without history, the VLM assigns p=0.65 to Frontier A based on category-level priors about bedrooms. With history, the VLM recognizes that the bedroom has already been observed without a television and lowers the probability to p=0.10.

B. When and Why It Fails

Despite strong benchmark performance, OpenFrontier remains susceptible to several failure modes. As Fig. 10 shows, failure types differ across benchmarks but follow a consistent pattern. One major challenge lies in termination criteria, which is non-trivial in open-world navigation settings. Some failures are influenced by limitations of the Habitat benchmark itself. However, our analysis indicates that these cases also expose a broader limitation of OpenFrontier, namely its limited ability to recover from failure once progress stalls. Even after a target is detected and navigation reduces to point-goal execution, the robot may move in incorrect directions or become trapped in local minima. In such cases, the system currently lacks a mechanism to quickly recognize failure and trigger re-reasoning or replanning at the semantic level. Enabling reliable failure detection and recovery is expected to significantly improve robustness and overall performance, and remains an important direction for future work.

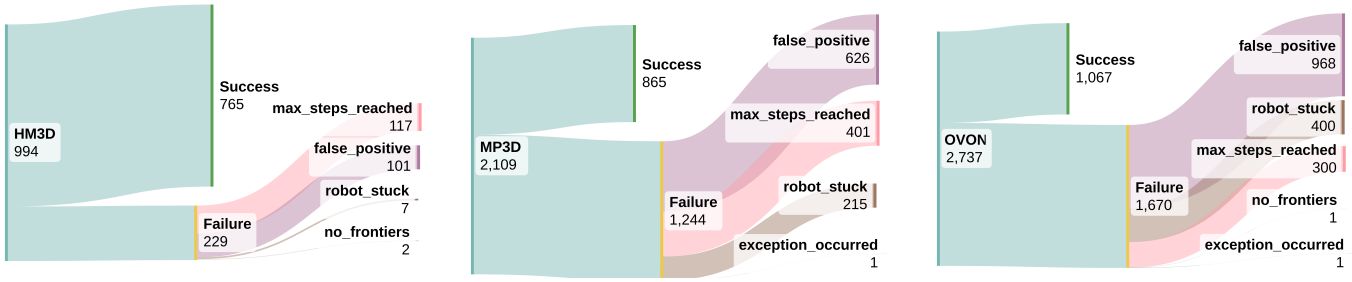

Fig. 10: **Failure Case Analysis.** (Left to right: HM3D, MP3D, and OVON) Across all three benchmarks, the two most common failure modes are false-positive target detections and termination due to exhausting the step budget.

VI. Conclusion

We presented OpenFrontier, a zero-shot framework that enables general language-conditioned navigation by grounding vision-language priors onto visual navigation frontiers. By treating navigation as a discrete subgoal selection problem and performing perception and semantic reasoning primarily in the 2D image domain, OpenFrontier avoids dense semantic mapping, task-specific policy training, and fine-tuning. Frontiers serve as sparse and physically grounded anchors that bridge high-level semantic reasoning with metric navigation, enabling efficient exploration and goal-directed behavior in both closed-set and open-set environments. We demonstrated that this design achieves competitive performance across multiple object-goal navigation benchmarks and transfers robustly to real-world deployment on a mobile legged robot. Our results suggest that effective grounding and system-level abstraction, rather than increasingly complex models or training pipelines, play a central role in scalable open-world navigation. We believe OpenFrontier provides a practical and flexible foundation for integrating future vision-language models into robotic navigation systems without requiring costly retraining or dense environment representations.

Acknowledgement

This work was partially supported by the Lamarr Institute and by the Robotics Institute Germany.

References

[1] Dhruv Batra, Aaron Gokaslan, Aniruddha Kembhavi, Oleksandr Maksymets, Roozbeh Mottaghi, Manolis Savva, Alexander Toshev, and Erik Wijmans. Objectnav revisited: On evaluation of embodied agents navigating to objects. *arXiv preprint arXiv:2006.13171*, 2020.

[2] Kenneth Blomqvist, Michel Breyer, Andrei Cramariuc, Julian Förster, Margarita Grinvald, Florian Tschopp, Jen Jen Chung, Lionel Ott, Juan Nieto, and Roland Siegwart. Go fetch: Mobile manipulation in unstructured environments. *arXiv preprint arXiv:2004.00899*, 2020.

[3] Chao Cao, Hongbiao Zhu, Howie Choset, and Ji Zhang. Tare: A hierarchical framework for efficiently exploring complex 3d environments. In *Robotics: Science and Systems*, volume 5, 2021.

[4] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.

[5] Matthew Chang, Théophile Gervet, Mukul Khanna, Sriram Yenamandra, Dhruv Shah, So Yeon Min, Kavit Shah, Chris Paxton, Saurabh Gupta, Dhruv Batra, et al. Goat: Go to any thing. 2024.

[6] Devendra Singh Chaplot, Dhiraj Prakashchand Gandhi, Abhinav Gupta, and Russ R Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. *Advances in Neural Information Processing Systems*, 33: 4247–4258, 2020.

[7] Devendra Singh Chaplot, Murtaza Dalal, Saurabh Gupta, Jitendra Malik, and Russ R Salakhutdinov. Seal: Self-supervised embodied active learning using exploration and 3d consistency. *Advances in neural information processing systems*, 34:13086–13098, 2021.

[8] An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Xueyan Zou, Jan Kautz, Erdem Biyik, Hongxu Yin, Sifei Liu, and Xiaolong Wang. Navila: Legged robot vision-language-action model for navigation. In *RSS*, 2025.

[9] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[10] Anna Dai, Sotiris Papatheodorou, Nils Funk, Dimos Tzoumanikas, and Stefan Leutenegger. Fast frontier-based information-driven autonomous exploration with an mav. In *ICRA*, 2020.

[11] Gemma Team et al. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

[12] Zhiyuan Feng, Zhaolu Kang, Qijie Wang, Zhiying Du, Jiongrui Yan, Shubin Shi, Chengbo Yuan, Huizhi Liang, Yu Deng, Qixiu Li, et al. Seeing across views: Benchmarking spatial reasoning of vision-language models in robotic scenes. *arXiv preprint arXiv:2510.19400*, 2025.

[13] Samir Yitzhak Gadre, Mitchell Wortsman, Gabriel Ilharco, Ludwig Schmidt, and Shuran Song. Cows on pasture: Baselines and benchmarks for language-driven zero-shot object navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23171–23181, 2023.

[14] Theophile Gervet, Soumith Chintala, Dhruv Batra, Ji-

tendra Malik, and Devendra Singh Chaplot. Navigating to objects in the real world. *Science Robotics*, 8(79): eadf6991, 2023.

[15] Dylan Goetting, Himanshu Gaurav Singh, and Antonio Loquercio. End-to-end navigation with vision language models: Transforming spatial reasoning into question-answering. *arXiv preprint arXiv:2411.05755*, 2024.

[16] Mobin Habibpour and Fatemeh Afghah. History-augmented vision-language models for frontier-based zero-shot object navigation, 2025.

[17] Cherie Ho, Seungchan Kim, Brady Moon, Aditya Parandekar, Narek Harutyunyan, Chen Wang, Katia Sycara, Graeme Best, and Sebastian Scherer. Mapex: Indoor structure exploration with probabilistic information gain from global map predictions. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13074–13080. IEEE, 2025.

[18] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[19] N. Hughes, Y. Chang, S. Hu, R. Talak, R. Abdulhai, J. Strader, and L. Carlone. Foundations of spatial perception for robotics: Hierarchical representations and real-time systems. *The International Journal of Robotics Research*, 2024. doi: 10.1177/02783649241229725. URL https://doi.org/10.1177/02783649241229725.

[20] Matan Keidar and Gal A Kaminka. Efficient frontier detection for robot exploration. *The International Journal of Robotics Research*, 33(2):215–236, 2014.

[21] Mukul Khanna, Ram Ramrakhya, Gunjan Chhablani, Sriram Yenamandra, Theophile Gervet, Matthew Chang, Zsolt Kira, Devendra Singh Chaplot, Dhruv Batra, and Roozbeh Mottaghi. Goat-bench: A benchmark for multi-modal lifelong navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16373–16383, 2024.

[22] Yuxuan Kuang, Hai Lin, and Meng Jiang. Openfmnav: Towards open-set zero-shot object navigation via vision-language foundation models, 2024.

[23] Richard Kuhlmann, Jakob Wolfram, Boyang Sun, Jiaxu Xing, Davide Scaramuzza, Marc Pollefeys, and Cesar Cadena. Sight over site: Perception-aware reinforcement learning for efficient robotic inspection. *ArXiv*, 2025.

[24] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

[25] Yuxing Long, Wenzhe Cai, Hongcheng Wang, Guanqi Zhan, and Hao Dong. Instructnav: Zero-shot system for generic instruction navigation in unexplored environment, 2024.

[26] Andrew Melnik, Gora Chand Nandi, et al. Cognitive planning for object goal navigation using generative ai models. *arXiv preprint arXiv:2404.00318*, 2024.

[27] J A Placed, J Strader, H Carrillo, N Atanasov, V Indelman, L Carlone, and J A Castellanos. A Survey on Active Simultaneous Localization and Mapping: State of the Art and New Frontiers. 2023.

[28] Kaixian Qu, Jie Tan, Tingnan Zhang, Fei Xia, Cesar Cadena, and Marco Hutter. Ippon: Common sense guided informative path planning for object goal navigation. *arXiv preprint arXiv:2410.19697*, 2024.

[29] Santhosh Kumar Ramakrishnan, Devendra Singh Chaplot, Ziad Al-Halah, Jitendra Malik, and Kristen Grauman. Poni: Potential functions for objectgoal navigation with interaction-free learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18890–18900, 2022.

[30] Victor Reijgwart, Cesar Cadena, Roland Siegwart, and Lionel Ott. Efficient volumetric mapping of multi-scale environments using wavelet-based compression. 2023-07.

[31] Hardik Shah, Jiaxu Xing, Nico Messikommer, Boyang Sun, Marc Pollefeys, and Davide Scaramuzza. Foresightnav: Learning scene imagination for efficient exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2025.

[32] Rutav Shah, Albert Yu, Yifeng Zhu, Yuke Zhu, and Roberto Martín-Martín. Bumble: Unifying reasoning and acting with vision-language models for building-wide mobile manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13337–13345. IEEE, 2025.

[33] Ioan A. Şucan, Mark Moll, and Lydia E. Kavraki. The Open Motion Planning Library. *IEEE Robotics & Automation Magazine*, 19(4):72–82, December 2012. doi: 10.1109/MRA.2012.2205651. https://ompl.kavrakilab.org.

[34] Boyang Sun, Hanzhi Chen, Stefan Leutenegger, Cesar Cadena, Marc Pollefeys, and Hermann Blum. Frontiernet: Learning visual cues to explore. *IEEE Robotics and Automation Letters*, 10(7):6576–6583, 2025. doi: 10.1109/LRA.2025.3569122.

[35] Jingwen Sun, Jing Wu, Ze Ji, and Yu-Kun Lai. A survey of object goal navigation. *IEEE Transactions on Automation Science and Engineering*, 22:2292–2308, 2025. doi: 10.1109/TASE.2024.3378010.

[36] Yuezhan Tao, Yuwei Wu, Beiming Li, Fernando Cladera, Alex Zhou, Dinesh Thakur, and Vijay Kumar. Seer: Safe efficient exploration for aerial robots using learning to predict information gain. In *ICRA*, 2023.

[37] Hongze Wang, Boyang Sun, Jiaxu Xing, Fan Yang, Marco Hutter, Dhruv Shah, Davide Scaramuzza, and Marc Pollefeys. What matters in rl-based methods for object-goal navigation? an empirical study and a unified framework. *arXiv preprint arXiv:2510.01830*, 2025.

[38] Meng Wei, Chenyang Wan, Xiqian Yu, Tai Wang, Yuqiang Yang, Xiaohan Mao, Chenming Zhu, Wenzhe Cai, Hanqing Wang, Yilun Chen, et al. Streamvln: Streaming vision-and-language navigation via slowfast context modeling. *arXiv preprint arXiv:2507.05240*,

2025.

[39] Erik Wijmans, Abhishek Kadian, Ari Morcos, Stefan Lee, Irfan Essa, Devi Parikh, Manolis Savva, and Dhruv Batra. Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=H1gX8C4YPr.

[40] Wei Xie, Haobo Jiang, Yun Zhu, Jianjun Qian, and Jin Xie. Naviformer: A spatio-temporal context-aware transformer for object navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 14708–14716, 2025.

[41] Zhefan Xu, Xinming Han, Haoyu Shen, Hanyu Jin, and Kenji Shimada. Navrl: Learning safe flight in dynamic environments. *IEEE Robotics and Automation Letters*, 10 (4):3668–3675, 2025. doi: 10.1109/LRA.2025.3546069.

[42] Xinda Xue, Junjun Hu, Minghua Luo, Xie Shichao, Jintao Chen, Zixun Xie, Quan Kuichen, Guo Wei, Mu Xu, and Zedong Chu. Omninav: A unified framework for prospective exploration and visual-language navigation. *arXiv preprint arXiv:2509.25687*, 2025.

[43] Karmesh Yadav, Santhosh Kumar Ramakrishnan, John Turner, Aaron Gokaslan, Oleksandr Maksymets, Rishabh Jain, Ram Ramrakhya, Angel X Chang, Alexander Clegg, Manolis Savva, Eric Undersander, Devendra Singh Chaplot, and Dhruv Batra. Habitat challenge 2022. https://aihabitat.org/challenge/2022/, 2022.

[44] Karmesh Yadav, Jacob Krantz, Ram Ramrakhya, Santhosh Kumar Ramakrishnan, Jimmy Yang, Austin Wang, John Turner, Aaron Gokaslan, Vincent-Pierre Berges, Roozbeh Mootaghi, Oleksandr Maksymets, Angel X Chang, Manolis Savva, Alexander Clegg, Devendra Singh Chaplot, and Dhruv Batra. Habitat challenge 2023. https://aihabitat.org/challenge/2023/, 2023.

[45] Brian Yamauchi. A frontier-based approach for autonomous exploration. In *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97.'Towards New Computational Principles for Robotics and Automation'*, pages 146–151. IEEE, 1997.

[46] Fan Yang, Chen Wang, Cesar Cadena, and Marco Hutter. iPlanner: Imperative Path Planning. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.064.

[47] Fan Yang, Per Frivik, David Hoeller, Chen Wang, Cesar Cadena, and Marco Hutter. Spatially-enhanced recurrent memory for long-range mapless navigation via end-to-end reinforcement learning. *The International Journal of Robotics Research*, page 02783649251401926, 2025.

[48] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.

[49] Hang Yin, Xiuwei Xu, Linqing Zhao, Ziwei Wang, Jie Zhou, and Jiwen Lu. Unigoal: Towards universal zero-shot goal-oriented navigation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19057–19066, 2025.

[50] Naoki Yokoyama, Sehoon Ha, Dhruv Batra, Jiuguang Wang, and Bernadette Bucher. Vlfm: Vision-language frontier maps for zero-shot semantic navigation. In *International Conference on Robotics and Automation (ICRA)*, 2024.

[51] Naoki Yokoyama, Ram Ramrakhya, Abhishek Das, Dhruv Batra, and Sehoon Ha. Hm3d-ovon: A dataset and benchmark for open-vocabulary object goal navigation, 2024.

[52] Bangguo Yu, Hamidreza Kasaei, and Ming Cao. Frontier semantic exploration for visual target navigation. *arXiv preprint arXiv:2304.05506*, 2023.

[53] Kuo-Hao Zeng, Zichen Zhang, Kiana Ehsani, Rose Hendrix, Jordi Salvador, Alvaro Herrasti, Ross Girshick, Aniruddha Kembhavi, and Luca Weihs. Poliformer: Scaling on-policy rl with transformers results in masterful navigators. In *Conference on Robot Learning*, pages 408–432. PMLR, 2025.

[54] Jiazhao Zhang, Liu Dai, Fanpeng Meng, Qingnan Fan, Xuelin Chen, Kai Xu, and He Wang. 3d-aware object goal navigation via simultaneous exploration and identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6672–6682, 2023.

[55] Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zhongyuan Wang, Zhizheng Zhang, and He Wang. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. *arXiv preprint arXiv:2412.06224*, 2024.

[56] Jiazhao Zhang, Kunyu Wang, Rongtao Xu, Gengze Zhou, Yicong Hong, Xiaomeng Fang, Qi Wu, Zhizheng Zhang, and He Wang. Navid: Video-based vlm plans the next step for vision-and-language navigation. *Robotics: Science and Systems*, 2024.

[57] Yue Zhang, Ziqiao Ma, Jialu Li, Yanyuan Qiao, Zun Wang, Joyce Chai, Qi Wu, Mohit Bansal, and Parisa Kordjamshidi. Vision-and-language navigation today and tomorrow: A survey in the era of foundation models. *arXiv preprint arXiv:2407.07035*, 2024.

[58] Boyu Zhou, Yichen Zhang, Xinyi Chen, and Shaojie Shen. Fuel: Fast uav exploration using incremental frontier structure and hierarchical planning. *IEEE Robotics and Automation Letters*, 6(2):779–786, 2021.

[59] Zibo Zhou, Yue Hu, Lingkai Zhang, Zonglin Li, and Siheng Chen. Beliefmapnav: 3d voxel-based belief map for zero-shot object navigation, 2025.

[60] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.