SemNav: A Model-Based Planner for Zero-Shot Object Goal Navigation Using Vision-Foundation Models

Arnab Debnath, Gregory J. Stein, Jana Košecká
George Mason University
Fairfax, VA
adebnath@gmu.edu, gjstein@gmu.edu, kosecka@gmu.edu

Abstract

Object goal navigation is a fundamental task in embodied AI, where an agent is instructed to locate a target object in an unexplored environment. Traditional learning-based methods rely heavily on large-scale annotated data or require extensive interaction with the environment in a reinforcement learning setting, often failing to generalize to novel environments and limiting scalability. To overcome these challenges, we explore a zero-shot setting where the agent operates without task-specific training, enabling more scalable and adaptable solution. Recent advances in Vision Foundation Models (VFMs) offer powerful capabilities for visual understanding and reasoning, making them ideal for agents to comprehend scenes, identify relevant regions, and infer the likely locations of objects. In this work, we present a zero-shot object goal navigation framework that integrates the perceptual strength of VFMs with a model-based planner that is capable of long-horizon decision making through frontier exploration. We evaluate our approach on the HM3D dataset using the Habitat simulator and demonstrate that our method achieves state-of-the-art performance in terms of success weighted by path length for zero-shot object goal navigation.

1. Introduction

Object-goal navigation is a core challenge in embodied AI, where an agent is tasked with locating and navigating to an instance of a specified object within an unfamiliar environment. Unlike map-based or point-goal navigation, this task requires both semantic understanding and active perception, as the agent must interpret visual cues to identify the target object without prior knowledge of its location. This capability is essential for a wide range of real-world applications, including service robots in domestic environments, assistive robots for the elderly, and search-and-rescue operations in disaster-stricken areas.

Over the years, various learning-based methods have been proposed for object-goal navigation. Supervised approaches [13] rely on large-scale annotated datasets, while reinforcement learning (RL) techniques [2, 11, 19] learn policies through trial-and-error guided by reward shaping. However, these methods are limited to closed sets of object categories and are tightly coupled to their training environments. Extending them to new objects or environments often requires costly retraining, hindering scalability and real-world deployment where generalization to unseen categories and dynamic layouts is essential.

Recent advances in VFMs have greatly enhanced the generalization capabilities of perception systems. Open-vocabulary detectors [6, 9] and segmentation models [3, 4] can recognize diverse object categories without a fixed training set, while vision-language models (VLM) and large language models (LLM) provide strong commonsense reasoning about objects and their spatial contexts. VLFM [20] leverages BLIP-2 [5] to compute the cosine similarity between the current observation and a target image, guiding the agent toward visually similar regions. ESC [24] utilizes GLIP [6] for detecting candidate objects and rooms, and employs an LLM to reason about the likelihood of the target object being present in a specific room or near certain objects. While these approaches show promise, their reliance on image-text matching or reasoning explicitly with discovered objects can hinder exploration when the target object is far away, occluded, or when the detection is poor. In this work, we explore the advancements in VFMs that enables us to use more general prompts which can encourage the agent to move through a hallway or enter a nearby room when the object is not currently visible.

Our approach adopts a frontier-based exploration strategy, treating frontiers as long-term subgoals. We leverage VLM to assign the likelihood of finding the target object to these frontiers, guiding the agent's search. Many existing frontier-based methods are inherently myopic: they score each frontier using a heuristic and greedily select the one with the highest score, often leading to suboptimal and inefficient behavior in complex environments. To address this,

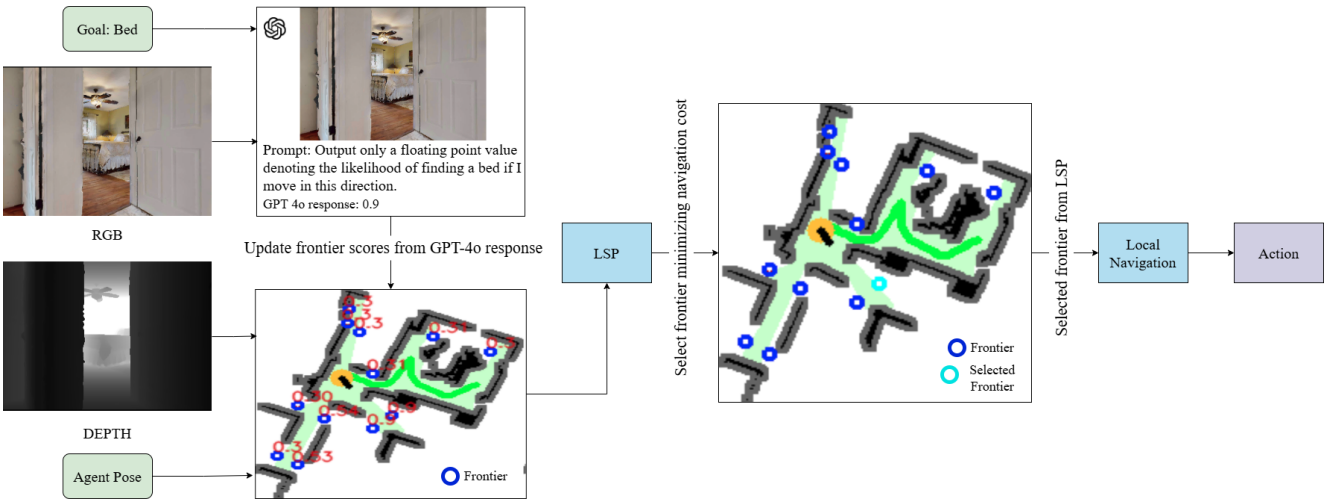


Figure 1. Overview of SemNav

we build upon the model-based frontier planning strategy, Learning Over Subgoals Planner (LSP) [15], which evaluates the expected navigation cost of selecting a frontier given the current belief about the environment. This allows the agent to reason about the long-term consequences of its actions and supports more efficient long-horizon planning.

Our contributions are summarized below:

- We propose a semantic frontier-based exploration framework, SemNav, that leverages VLM to assign cost to frontiers for zero-shot object-goal navigation in unseen environments.
- We integrate a non-greedy, model-based frontier planning strategy (LSP) to reason about long-term exploration costs, enabling more efficient and informed decision-making compared to traditional greedy frontier selection methods.

2. Related Work

Zero-shot object-goal navigation leverages VLM and LLMs to enable open-vocabulary object understanding and semantic reasoning without task-specific training. Recent works explore various strategies, including transfer learning [1, 10], frontier scoring with VLM and LLMs [20, 21], structured scene reasoning [24], and topological exploration guided by LLM [18].

3. Approach

An agent is spawned at a random location within an unknown environment and receives a target object category (e.g., couch). At each time step, the agent receives visual observations as egocentric RGB-D images and utilizes a GPS+Compass sensor. The agent can take several actions at each step: `MOVE_FORWARD`, `TURN_LEFT`, `TURN_RIGHT`, or `STOP`. The `MOVE_FORWARD` action moves the agent forward by $0.25m$, and the `TURN` actions rotate the agent by $30°$ in the specified direction. An episode is considered successful if the agent takes a `STOP` action within 1m of any instance of the target object category within 500 timesteps.

Overview At each timestep t, the robot receives an RGB-D observation. The depth image is used to update a partial map of the environment, identifying both navigable areas and obstacles. Frontiers are extracted as potential exploration targets. To guide exploration, a VFM model (GPT-4o) is queried using the current RGB image and the target object. The model estimates the likelihood of finding the target object in the direction of the current observation. This score is associated with the frontier. A long-horizon global planner LSP[15], selects the most promising frontier that minimizes the expected cost of navigation. A local navigation module then computes a low-level action to move the agent toward the selected frontier. Once the target object is detected, the local navigation module guides the robot to its location and issues a `STOP` action (Fig. 1).

3.1. Mapping module

We maintain an egocentric, partial map of the environment that is updated incrementally as the agent explores. At each timestep, the depth image is converted into a 3D point cloud, from which free space and obstacles are identified using a height threshold. Frontiers are extracted as boundaries between free and unknown space. For each frontier, the center point along its boundary is selected as the navigation target.

In addition to the geometric map, we also construct a two-channel value map, inspired by VLFM [20], consisting of a semantic value, V and a confidence score map, C. The semantic value map stores the estimated likelihood of finding the target object from the VLM (Sec. 3.2) in each explored cell. Instead of averaging

VLM scores across timesteps, we update the value map using a confidence score map that reflects the agent's visual certainty. Confidence is highest (1) along the optical axis and decays to 0 at the edges of the field of view (FOV), ensuring that semantic scores are updated more reliably in well-observed regions. The map cells corresponding to the current FOV are updated using the equations: $v_{i,j}^{new} = ((c_{i,j}^{curr} v_{i,j}^{curr}) + (c_{i,j}^{prev} v_{i,j}^{prev})/(c_{i,j}^{curr} + c_{i,j}^{prev}); c_{i,j}^{new} = ((c_{i,j}^{curr})^2 + (c_{i,j}^{prev})^2)/(c_{i,j}^{curr} + c_{i,j}^{prev})$

3.2. Assigning VLM-probabilities to frontiers

To guide exploration, we leverage GPT-4o to estimate the likelihood of encountering the target object if the agent moves in the direction of its current observation. At each timestep t, the RGB image I_t is fed into GPT-4o along with the prompt: "Output only a floating point value denoting the likelihood of finding a <target object> if I move in this direction." Unlike prior approaches that rely on image captioning models (e.g., BLIP-2) [20] or spatial co-occurrence heuristics based on object-to-object relationships [21, 24], GPT-4o provides a richer understanding of scene semantics. Its integrated vision-language reasoning capabilities allow it to infer plausible object locations by incorporating knowledge about indoor layouts, typical room-object associations, and commonsense spatial relationships. This makes it particularly effective in situations where the target object is not directly visible—such as when it is located through a doorway, or further down a hallway—where purely visual similarity metrics may fail (Fig. 2). Some responses from GPT-4o are included in the Appendix Sec. 7.

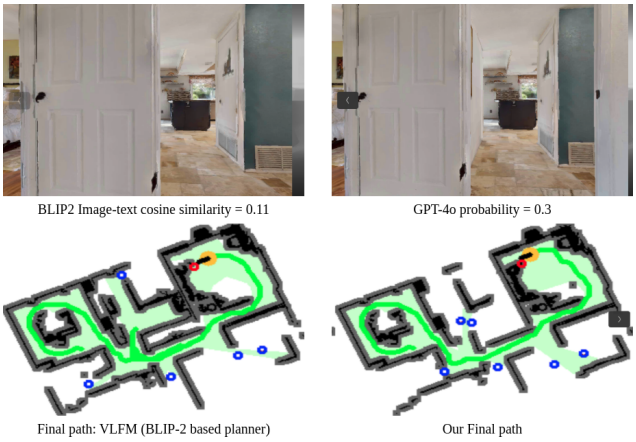


Figure 2. BLIP-2 assigns a lower score for the couch from the same location, leading the agent to explore the left room first, while our GPT-4o-based approach assigns a higher score and guides the agent directly through the hallway to find the couch.

3.3. Global planning with LSP

Once the semantic scores for the frontiers are computed, we must select the most promising frontier. To do so, we adopt the LSP [15], a model-based planning abstraction for efficient navigation. This approach has been shown to be effective for long-horizon reasoning in both exploration [8] and point-goal navigation [7] tasks within simulated environments.

Unlike myopic methods that select the frontier with the highest immediate score, LSP evaluates the long-term utility of each frontier by estimating the expected cost of navigation as:

$$Q(\{m_t, q_t, \mathcal{A}_t\}, a_t) = D(m_t, q_t, a_t) + P_S R_S + (1 - P_S)(R_E + \max_{a \in \mathcal{A}_t \setminus \{a_t\}} Q(\{m_t, q_{a_t}, \mathcal{A}_t \setminus \{a_t\}\}, a) \quad (1)$$

Here, m_t is the current partial map, q_t represents the agent's location, and $\mathcal{A}_t$ is the set of candidate frontiers. $Q(\{m_t, q_t, \mathcal{A}_t\}, a_t)$ estimates the expected cost of successfully locating the target object by first navigating to frontier a_t. $D(m_t, q_t, a_t)$ is the distance from the agent to the frontier. P_S is the probability of successfully finding the target object through frontier a_t, which we get from the VLM. R_S and R_E denote the success and exploration costs respectively. Specifically, R_S represents the expected distance to the object beyond a frontier if found, while R_E captures the additional exploration and return cost if the object is not found. Since VFMs cannot reliably predict distances in unexplored regions, we assume $R_S = 3m$ and $R_E = 6m$ in our experiments, assuming unsuccessful exploration costs roughly twice as much as success. Empirically, we observed that varying these constants has minimal impact on performance. We calculate the expected cost for all frontiers and select the one with the minimum expected cost.

3.4. Local navigation and object detection

The agent relies on a local navigation module to reach either the frontier or the target object location once it is visually detected by YOLOv7 [16]. Upon detection, we use Mobile-SAM to generate a segmentation mask of the object. The segmentation mask is then combined with the corresponding depth image to estimate the closest point on the object surface relative to the robot's current position, which serves as the navigation goal. For reaching any target location (frontier or goal), we utilize the pretrained local point-goal navigation policy from [20], which was trained on the HM3D training split. This policy takes as input the current depth image and the relative position of the goal point, and produces discrete action commands to navigate towards it.

4. Experiments

Dataset We evaluate our approach using the Habitat [14] simulator with the HM3D-val dataset [12], which comprises 2000 episodes across 20 diverse indoor scenes and includes navigation tasks for 6 object categories (bed, couch, chair, potted plant, toilet, tv).

Evaluation Metrics We report performance using two standard metrics: Success Rate (SR) and Success weighted by Path Length (SPL). SR measures the percentage of episodes in which the agent successfully navigates to an instance of the target object. SPL evaluates path efficiency by comparing the length of the agent's trajectory to the shortest path from the starting location to the nearest instance of the target object.

Baselines We compare our approach against the following baselines:

- **ZSON** [10]: Transfers a trained policy on image-goal navigation to semantic object-goals.
- **ESC** [24]: Uses LLM to reason about room-to-target object and discovered objects-to-target object relationship.
- **VLFM** [20]: It uses BLIP-2 [5] to compute cosine similarity between the current observation and the target object and uses the similarity score to prioritize frontiers.
- **VoroNav** [18]: Introduces a voronoi graph-based topological map, selecting subgoals using LLM-inferred probabilities with efficiency and exploration rewards.
- **L3MVN** [21]: Uses masked language model to assess the relevance between frontiers with nearby objects to target
- **TriHelper** [23]: Builds on L3MVN by incorporating three manual intervention helpers—collision, exploration, and detection—to boost navigation performance.
- **MFNP** [22]: Supports exploration through staircases.

5. Result

Approach	HM3D	
	SPL ↑	SR↑
ZSON [10]	12.6	25.5
ESC [24]	22.3	39.2
VLFM [20]	30.4	52.5
VoroNav [18]	26	42
L3MVN [21]	23.1	50.4
TriHelper [23]	25.3	56.5
MFNP [22]	26.7	**58.3**
SemNav-Greedy	34.3	54.6
SemNav	**35.9**	54.9

Table 1. Obj-nav results on HM3D validation set. SemNav-Greedy uses our frontier scoring mechanism with VLM and greedily selects the frontier with the highest probability score. SemNav uses LSP along with VLM-based scoring.

Our method, SemNav, outperforms all baselines in terms of SPL, achieving the most efficient navigation paths (Tab. 1). Compared to VLFM, which also leverages VLM to score frontiers, SemNav achieves 5.5% and 2.4% improvement in SPL and SR, respectively. These results show that our way of prompting the VLM for semantic reasoning, combined with long-horizon planning through LSP, leads to more effective exploration than relying solely on image-text cosine similarity as done in VLFM (Fig. 2). We observe a 1.6% SPL drop with greedy frontier selection. (SemNav-Greedy vs SemNav in Table 1). Fig. 3 compares VLFM, SemNav, and SemNav-Greedy, showing that our approach achieves the highest SPL (detailed discussion in Appendix Sec. 9). Specific examples illustrating SemNav's behavior—such as efficiently exiting rooms when the target object is absent compared to VLFM is provided in the Appendix Sec. 8.

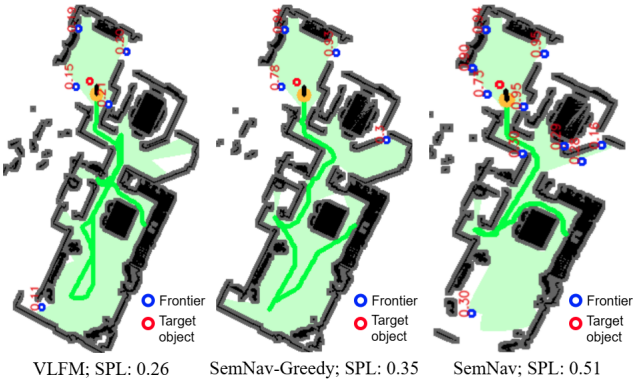


Figure 3. Final navigation paths for the target object bed.

While our method significantly outperforms TriHelper and MFNP in SPL, it achieves slightly lower SR—by 1.6% and 3.4% respectively. These methods specifically address certain failure cases: TriHelper manually intervenes when the robot becomes stuck by relocating it to a more open area and double-checks object detections using a VLM, while MFNP further enables exploration through staircases. In contrast, our approach prioritizes improved semantic exploration through better frontier reasoning and does not currently handle stair navigation.

We also observed that around 25% of the failed episodes are due to a simulator issue, where the agent successfully reaches an instance of the target object category but the simulator still registers the episode as a failure. This phenomenon has also been noted in previous works [17, 23]. Representative examples illustrating this issue are provided in the Appendix (Sec. 10).

6. Conclusion

We proposed SemNav, a frontier-based zero-shot object-goal navigation framework that combines the semantic understanding of VLMs with model-based long-horizon planning. By leveraging VLM-driven frontier scoring and the LSP planner, our approach achieves state-of-the-art SPL performance on the HM3D dataset. Our results highlight the importance of general semantic reasoning and model-based planning for open-world embodied AI tasks.

References

[1] Ziad Al-Halah, Santhosh Kumar Ramakrishnan, and Kristen Grauman. Zero experience required: Plug & play modular transfer learning for semantic visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17031–17041, 2022. 2

[2] 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, 2020. 1

[3] Xiaoyi Dong, Jianmin Bao, Yinglin Zheng, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, et al. Maskclip: Masked self-distillation advances contrastive language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10995–11005, 2023. 1

[4] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 1

[5] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 1, 4

[6] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10965–10975, 2022. 1

[7] Yimeng Li, Arnab Debnath, Gregory J. Stein, and Jana Kosecka. Comparison of model free and model-based learning-informed planning for PointGoal navigation. In *CoRL 2022 Workshop on Learning, Perception, and Abstraction for Long-Horizon Planning*, 2022. 3

[8] Yimeng Li, Arnab Debnath, Gregory J Stein, and Jana Košecká. Learning-augmented model-based planning for visual exploration. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5165–5171. IEEE, 2023. 3

[9] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024. 1

[10] Arjun Majumdar, Gunjan Aggarwal, Bhavika Devnani, Judy Hoffman, and Dhruv Batra. Zson: Zero-shot object-goal navigation using multimodal goal embeddings. *Advances in Neural Information Processing Systems*, 35:32340–32352, 2022. 2, 4

[11] Anwesan Pal, Yiding Qiu, and Henrik Christensen. Learning hierarchical relationships for object-goal navigation. In *Conference on Robot Learning*, pages 517–528. PMLR, 2021. 1

[12] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-Matterport 3D dataset (HM3d): 1000 large-scale 3D environments for embodied AI. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. 3

[13] 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*, 2022. 1

[14] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied AI research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9339–9347, 2019. 3

[15] Gregory J Stein, Christopher Bradley, and Nicholas Roy. Learning over subgoals for efficient navigation of structured, unknown environments. In *Conference on Robot Learning*, pages 213–222. PMLR, 2018. 2, 3

[16] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7464–7475, 2023. 3

[17] Justin Wasserman, Girish Chowdhary, Abhinav Gupta, and Unnat Jain. Exploitation-guided exploration for semantic embodied navigation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2901–2908. IEEE, 2024. 4

[18] Pengying Wu, Yao Mu, Bingxian Wu, Yi Hou, Ji Ma, Shanghang Zhang, and Chang Liu. Voronav: voronoi-based zero-shot object navigation with large language model. In *Proceedings of the 41st International Conference on Machine Learning*, pages 53737–53775, 2024. 2, 4

[19] Karmesh Yadav, Arjun Majumdar, Ram Ramrakhya, Naoki Yokoyama, Alexei Baevski, Zsolt Kira, Oleksandr Maksymets, and Dhruv Batra. Ovrl-v2: A simple state-of-art baseline for imagenav and objectnav. *arXiv preprint arXiv:2303.07798*, 2023. 1

[20] Naoki Harrison Yokoyama, Sehoon Ha, Dhruv Batra, Jiuguang Wang, and Bernadette Bucher. Vlfm: Vision-language frontier maps for zero-shot semantic navigation. In *2nd Workshop on Language and Robot Learning: Language as Grounding*, 2023. 1, 2, 3, 4

[21] Bangguo Yu, Hamidreza Kasaei, and Ming Cao. L3mvn: Leveraging large language models for visual target navigation. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3554–3560. IEEE, 2023. 2, 3, 4

[22] Lingfeng Zhang, Hao Wang, Erjia Xiao, Xinyao Zhang, Qiang Zhang, Zixuan Jiang, and Renjing Xu. Multi-floor zero-shot object navigation policy. *arXiv preprint arXiv:2409.10906*, 2024. 4

[23] Lingfeng Zhang, Qiang Zhang, Hao Wang, Erjia Xiao, Zixuan Jiang, Honglei Chen, and Renjing Xu. Trihelper: Zero-shot object navigation with dynamic assistance. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10035–10042. IEEE, 2024. 4

[24] Kaiwen Zhou, Kaizhi Zheng, Connor Pryor, Yilin Shen, Hongxia Jin, Lise Getoor, and Xin Eric Wang. Esc: Exploration with soft commonsense constraints for zero-shot object navigation. In *International Conference on Machine Learning*, pages 42829–42842. PMLR, 2023. 1, 2, 3, 4

SemNav: A Model-Based Planner for Zero-Shot Object Goal Navigation Using Vision-Foundation Models

Supplementary Material

7. Probability outputs from GPT-4o

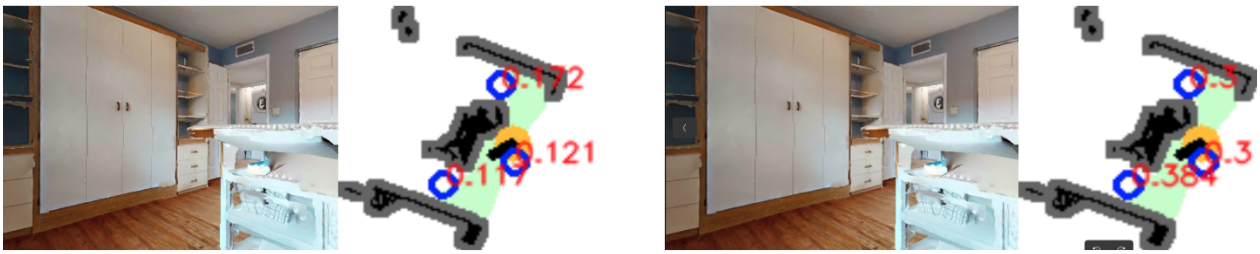
Prompt: Output only a floating point value denoting the likelihood of finding a **toilet** if I move in this direction.

GPT-4o Response: 0.65 | GPT-4o Response: 0.3 | GPT-4o Response: 0.3 | GPT-4o Response: 0.9

Figure 4. Some example images with their corresponding GPT-4o responses.

8. Comparison between SemNav and VLFM

Target Object: Couch



VLFM, Timestep t = 4 | LSPNav, Timestep t = 4

VLFM, Timestep t = 86 | LSPNav, Timestep t = 45

Figure 5. This is an example where we see that SemNav quickly gets out of the room to find a couch than VLFM. Top row: At timestep t=4, both are looking at a doorway which can lead to finding a couch. At that position SemNav had a higher probability score for the frontier closest to the doorway, which is evident from the map. This eventually leads SemNav agent to quickly get out of the room. Bottom row: SemNav gets out of the room at timestep $t = 45$ whereas VLFM took timestep $t = 86$ to get out of the room. The map shows a longer travelled path for VLFM.

9. Comparison among VLFM, SemNav-Greedy and SemNav

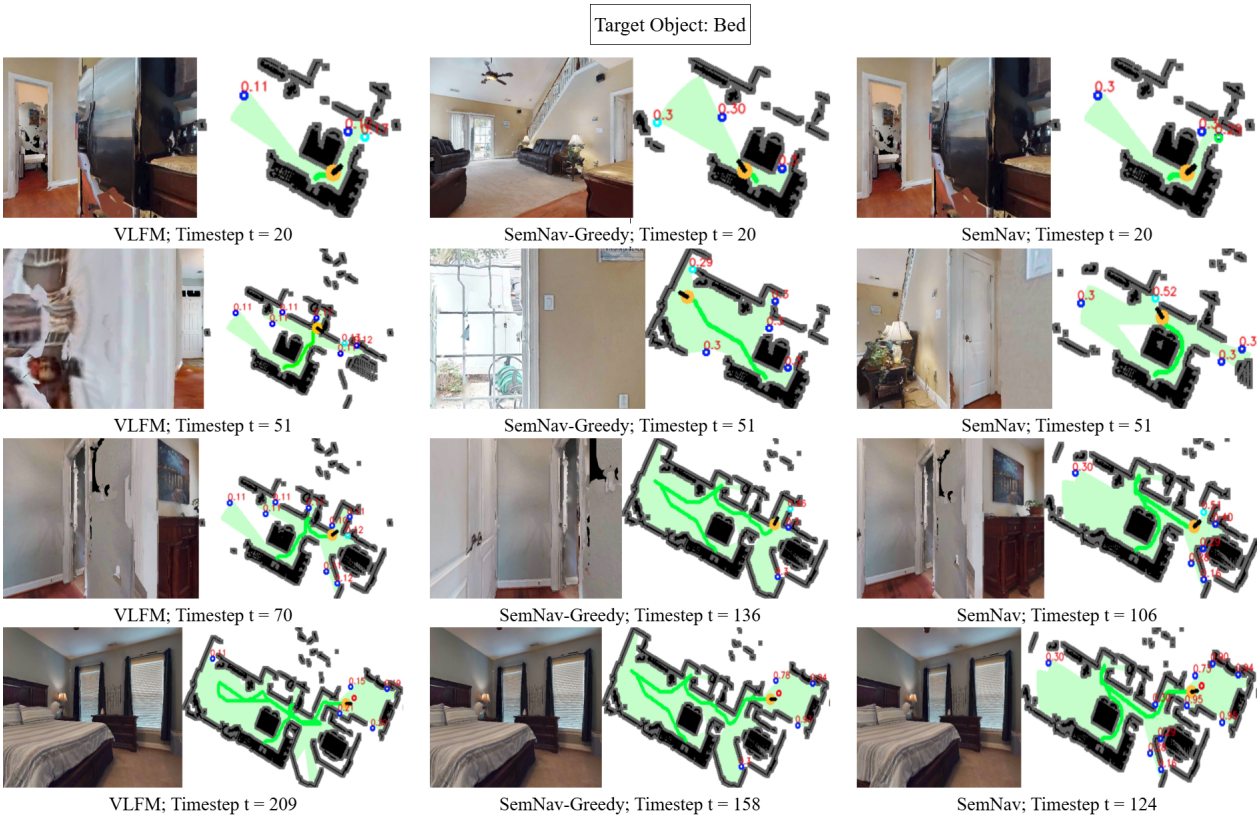


Figure 6. Row 1: At timestep $t = 20$, SemNav (with LSP) selects a frontier with a lower score (0.29 in map) to minimize expected travel cost, choosing closer frontiers that guide it toward the target. In contrast, SemNav-Greedy selects the frontier with the highest score and moves in the opposite direction. Row 2: As a result, SemNav-Greedy is now farthest from the goal. Row 3: All agents observe the door leading to the goal. VLFM receives a poor score from BLIP-2 in that direction and continues exploring other frontiers first. SemNav and SemNav-Greedy, guided by higher GPT-4o scores, proceed directly through the door. Row 4: SemNav reaches the target object faster than all other approaches.

10. Simulator issue

Figure 7. Here, we show a problem with the simulator where the agent successfully reached the target object, but it was still considered failed episodes.