CVPR #19

Human-like Navigation in a World Built for Humans

Anonymous CVPR submission

Paper ID 19

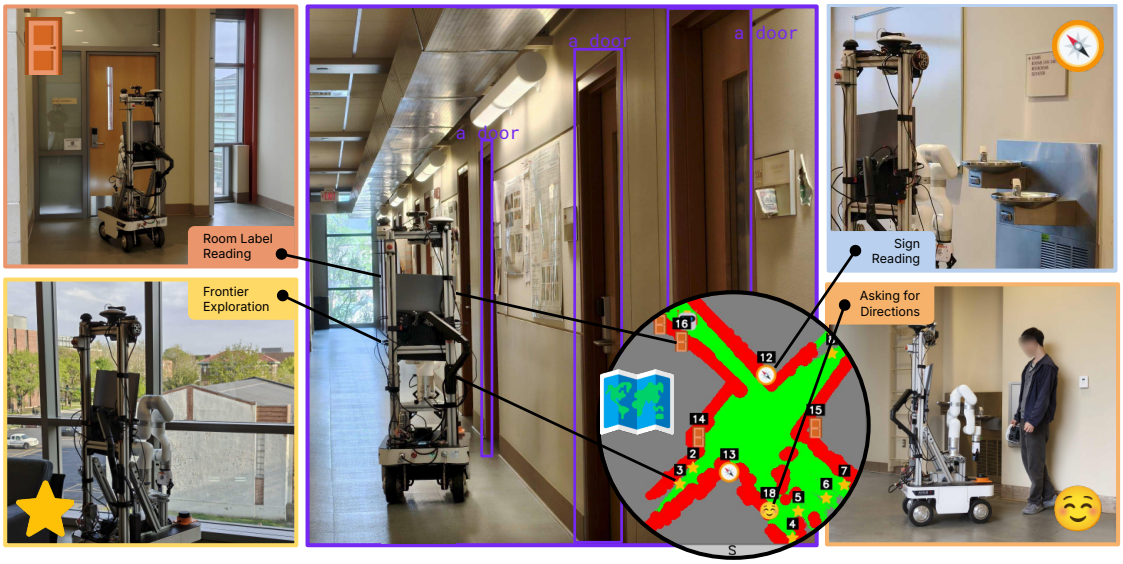

Figure 1. **Higher-order navigation skills.** Humans employ various skills involving higher-order reasoning in order to navigate to their destinations efficiently. These skills take advantage of key knowledge resources in the environment through high-level language and visual processing. We present a navigation method that imbues robots with these skills by integrating them in a VLM agent framework.

Abstract

When navigating in a man-made environment they haven't visited before—like an office building—humans employ behaviors such as reading signs and asking others for directions. These behaviors help humans reach their destinations efficiently by reducing the need to search through large areas. Existing robot navigation systems lack the ability to execute such behaviors and are thus highly inefficient at navigating within large environments. We present ReasonNav, a modular navigation system which integrates these human-like navigation skills by leveraging the reasoning capabilites of a vision-language model (VLM). We design compact input and output abstractions based on navigation landmarks, allowing the VLM to focus on language understanding and reasoning. We evaluate ReasonNav on real and simulated navigation tasks and show that the agent successfully employs higher-order reasoning to navigate efficiently in large, complex buildings. Our work is under submission at CoRL 2025.

1. Introduction

Imagine that you are an office worker and are asked to deliver a report to Jane Doe's office. What steps would you take to complete this task? First, you might search in a directory to find out the building and room number for Jane Doe's office. Then, you might look for signs that indicate the direction of that room. You can integrate the information you receive from each sign with the layout of the scene you see around you to decide where to look next. Along the way, you might ask people nearby for further clarifications.

Our civilization is built to be easy for humans to navigate within. There is an abundance of knowledge-offering resources around us that we leverage to navigate the world efficiently. Directional signs are placed deliberately at junctions to eliminate the risk of going the wrong way. Room labels follow orderly patterns so that reading a few can allow one to infer the locations of other rooms. Such guidance is necessary in order to deal with the inherent uncertainty of navigation in unseen environments.

Existing robot navigation systems lack the skills needed

to leverage these resources and thus lose out in navigation efficiency by spending unnecessary time exploring. We call these skills, which include sign reading and asking for directions, *higher-order navigation skills* because they require higher-order reasoning abilities and language processing. These skills become increasingly important in larger environments, where exploring in the wrong direction can cost a massive amount of time.

Our key insight is that such higher-order navigation skills can be integrated in a unified manner by taking advantage of recent advances in large vision-language models (VLMs). In this paper, we present ReasonNav, a modular system for human-like navigation that leverages the zero-shot reasoning capabilities of a VLM in an agentic manner. The system is comprised of two streams: a low-level stream that handles localization, mapping, and path planning, and a high-level stream where the VLM performs high-level planning on abstracted observation and action spaces. Specifically, we represent the environment using a memory bank of landmarks (e.g. map frontiers, doors, people, signs) with attached textual information. This simplifies both the input and output spaces for the VLM agent, allowing it to focus on higher-order reasoning.

We evaluate ReasonNav in real and simulated environments. In both cases, the robot is tasked with finding a given room in a large (unseen) building. This mimics a practical indoor delivery scenario. We show that our abstraction design allows the VLM to interpret information from signs and people and use it to guide its decision-making. We compare our full system with ablated versions and demonstrate that such higher-order navigation skills greatly impact navigation performance. Overall, the results suggest that our VLM agent framework is a promising path forward for achieving human-like navigation efficiency using higher-order reasoning skills.

2. Related Work

Agentic Foundation Models in Robotics. Task and motion planning (TAMP) approaches traditionally rely on predefined symbolic reasoning or optimization to plan for long-horizon tasks. Previous works [12, 26, 27, 29] have leveraged large language models (LLMs) to decompose high-level instructions into actionable subtasks, allowing for more user-friendly robotics systems. More recent approaches utilize Vision-Language Models (VLMs) to ground reasoning for more general and capable robot systems. VLMs have been shown to generalize across diverse objects and tasks in table-top manipulation [13, 15, 37], and enable zero-shot navigation to semantic goals across different environments [2, 18, 30]. Integrating these capabilities for mobile manipulation has seen improved potential in recent works [5, 31, 37], which are divided into two main categories: 1. prompt-based querying and 2. fine-tuning for direct perception to action pipeline. Our approach falls into the first category, querying a VLM for high-level task planning and using modular out-of-the-box controllers to execute actions. However, in contrast to the aforementioned methods that mainly rely on sensory inputs to perceive the world, our methods can leverage other resources, such as asking humans for help or actively seeking visual cues for navigation.

Open-world Navigation. In recent years, the rise in popularity of large-scale pre-trained foundation models has witnessed the emergence of open-world navigation. Particularly, early works of CLIP-on-Wheels (CoW) [9] and LM-Nav utilize CLIP [25] to establish a top-down confidence map for language-guided object-goal navigation or to precompute a language-embedded topological graph. Further works have expanded on this direction, using foundation models to pre-build highly expressive language-embedded semantic maps for long-horizon and fine-grained navigation tasks [3, 10, 11, 14, 22, 24, 33]. However, these approaches are computationally expensive, typically requiring multiple traversals over the operational area and hours of computing, and are unable to operate in unknown environments. Recent works [1, 4, 18, 26, 35, 36, 39] address this shortcoming by adopting LLMs and VLMs' high-level planners, taking advantage of their high-level reasoning capabilities to relax the requirement of costly pre-built maps. Our method falls into this category, enjoying the scalability and zero-shot transferability to the unknown world. However, we focus on practical navigation in man-made environments and the unique skills needed to succeed in such settings.

Interactive Navigation. Although these large foundation models have been trained on the vast majority of internet data and have shown promising results for robotic tasks, solely relying on them has proven to be inefficient. In recent years, the robotics community has been exploring human-in-the-loop feedback for corrections during robot's execution, especially for manipulation tasks [6, 16, 17, 38] and visual question answering tasks [7, 27, 28, 32]. Despite showing promising results, these methods typically require immediate human feedback, which is often not possible in real-world navigation scenarios. In contrast, our work mitigates this issue by leveraging more than just human feedback as an additional source of information, utilizing wayfinding cues (room labels, navigation signs, web searches) for more robust and efficient navigation.

3. Method

ReasonNav is a modular system for human-like navigation that heavily leverages the zero-shot reasoning capabilities of a Vision-Language Model (VLM) for efficient exploration

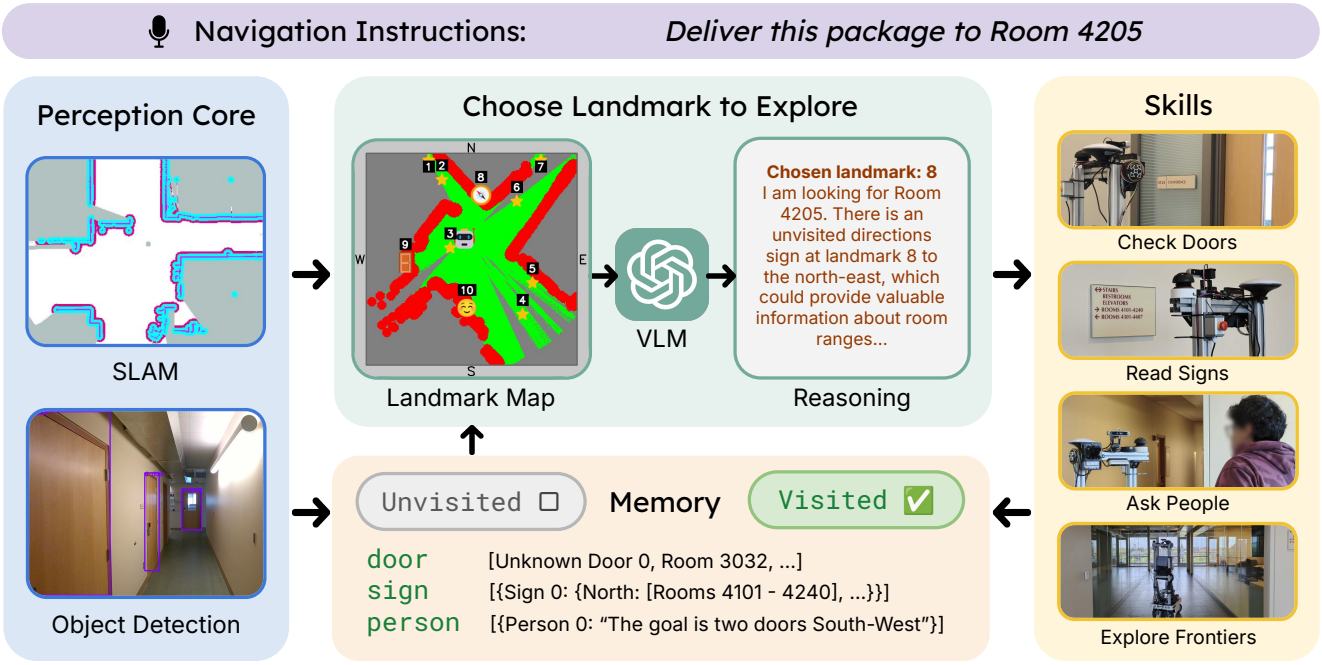

Figure 2. **Overview of ReasonNav.** The system is comprised of a low-level stream and a high-level stream. The low-level stream performs SLAM and object detection for key object categories (doors, signs, and people), feeding into a global memory bank. The high-level stream consists of a VLM planner that receives abstracted observations in the form of a JSON landmark dictionary and a map visualization. The VLM outputs the next landmark to explore, upon which predefined behavior primitives are executed based on the landmark category.

in unseen buildings. The system can be separated into a low-level stream and a high-level stream. The low-level stream includes standard localization and mapping modules that run at high frequency and an analytical path planner (Sec. 3.2). The high-level stream consists of a VLM agent that receives specially abstracted scene information to mimic the conscious decision-making processes used by humans during navigation (Sec. 3.3). The VLM chooses map frontiers to explore and decides when to perform skills such as sign reading, which are executed via predefined behavior primitives (Sec. 3.4).

3.1. Robot Hardware Setup

Our robot, shown in Fig. 3, is custom-built and consists of a mobile base, an arm, a computer, and various sensors. Components are attached to the base using aluminum t-slots and 3d-printed mounts.

We use the AgileX Ranger Mini 2.0, a mobile platform with 4-wheel steering and an onboard power supply, and an xArm 7 robot arm (currently unused). For obstacle avoidance and mapping, there is a Slamtech 2D lidar mounted near the base and a Hesai FT120 solid-state 3D lidar mounted on top. Two Realsense D455's are mounted on top for perception, one on a pan-tilt mechanism and the other angled downward toward the robot workspace. Only the pan-tilt camera is used in this work, and it is only used for object detection and text reading, not mapping. The

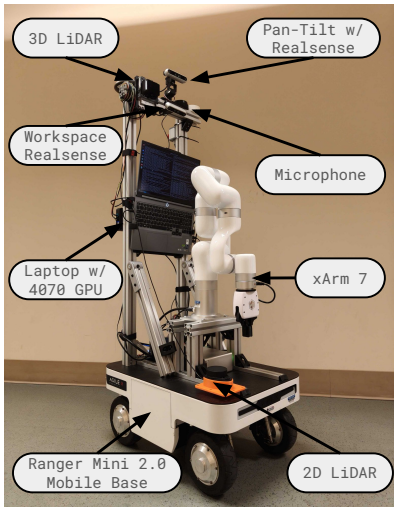

Figure 3. **Hardware System Overview.**

height of the camera is roughly aligned with most indoor signs, and the pan-tilt capability allows for viewing the surroundings quickly without turning the entire robot. For interacting with humans, we use a Respeaker omnidirectional microphone, mounted near the top for conversation with humans. An Ardusimple simpleRTK3B GPS is also mounted but is unused in this work.

Our onboard compute comes from a Lenovo Legion 5i laptop with an NVIDIA 4070 GPU. The entire system is

integrated using ROS2 Humble with all sensors/robot interfaces connected to the laptop. The laptop is connected to the internet in order to query GPT 4.1 via its API.

3.2. Localization and Mapping

We first describe our system's low-level processing stream, which is responsible for producing a top-down map of the environment, localizing the agent with respect to the map, detecting certain objects, and path planning. We perform 2D simultaneous localization and mapping (SLAM) using SLAM Toolbox [20] which merges the 3D lidar scan into the 2D scan and performs optimization to produce a top-down occupancy map of areas the agent has explored. Concurrently, we perform object detection on images from the RealSense camera using an open-vocabulary detector without retraining. We use NanoOWL, an optimized implementation of OWL-ViT [23], and query with three text labels: `door`, `person`, and `directions sign`. For path planning, we use NavFn, a wavefront Dijkstra planner from Nav2 [21], and execute the paths using an MPPI [34] controller.

3.3. VLM Observation and Action Abstraction

The key idea of our approach is to leverage VLMs in an agentic framework to integrate human-like behaviors that greatly improve navigation efficiency. VLMs excel at understanding language and conducting many forms of commonsense reasoning. However, they struggle at understanding complex spatial data and directly producing precise numerical outputs [8, 19]. Thus, we need to carefully design abstractions for both the input (observations) and output (actions) of the VLM in order to effectively leverage its reasoning capabilities.

Landmarks. Our abstraction design is centered heavily on the concept of *landmarks*, which refer to salient objects that are especially important in navigation tasks. Specifically, the landmarks refer to objects of the three categories mentioned above: doors, people, and directional signs, along with frontiers of the top-down map. Our system populates a memory bank of the objects from the output of the detector and attaches additional navigation-relevant information to each one as various skills are performed. For doors, we attach the text of the associated room label. For people, we attach a summary of the information received from them. For directional signs, we attach a list of cardinal directions and the sign text reading(s) associated with each direction. All of the objects are attached to a label of "Visited" or "Unvisited".

VLM Input and Output. We prompt the VLM with text instructions and two forms of abstracted scene information. One is the memory bank of landmarks, including both objects and map frontiers, in JSON format. Each landmark is assigned an index and may be attached to additional information as described above. The second form of information is an image visualization of the agent's top-down map. The map is colored based on occupancy and explored areas, and for each landmark, we plot its location on the map with a symbol of its category and its index number (see Fig.1). This gives a compact, high-level summary of the scene layout and the important objects the agent has seen thus far. We prompt the VLM to use these two forms of information to decide *which landmark* to visit next. This design ensures that the VLM can flexibly choose any reasonable high-level plan while not being tasked with predicting precise numerical coordinates.

3.4. Behavior Primitives

Each landmark category has an associated behavior primitive, which will be executed based on the VLM's choice. We describe each one below:

Frontier (Exploration). The agent moves to the desired frontier and turns 360-degrees around to scan its surroundings with Nav2's point goal planner and controller. Frontier navigation enables us to explore the unvisited region and identify more landmarks.

Door (Room Label Reading). The agent approaches the door and pans its camera while querying the object detector for `room label`. If a room label is detected, the agent moves closer and reads it via another call to the VLM. The text is attached to the door in the memory bank. If the goal is found, the episode ends here.

Person (Asking for Directions). The agent approaches the person and asks for directions using a text-to-speech model. It then records the person's response using speech-to-text. Next, it calls the VLM to produce a short note about the information it received, which is then attached to the person's landmark in the memory bank. Importantly, we request the VLM to use cardinal directions in the global map frame instead of relative directions such as "left" or "right" so that the note can be understood later without needing the agent's pose at the time of recording (Fig. 9).

Directional Sign (Sign Reading). The agent approaches the sign and reads it via a call to the VLM. The sign text is grouped based on arrow direction (binned into cardinal directions), and the directions are transformed into the global map frame for recording.

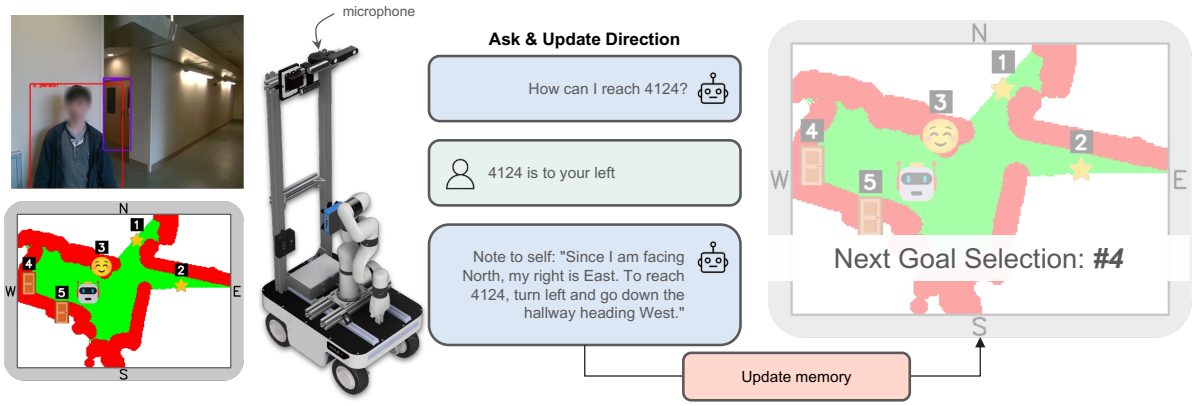

Figure 4. **Overview of the "Direction Asking" Skill:** The agent identifies nearby humans and logs them in its spatial memory (#3 in the map). When needed, it approaches and asks for goal directions via text-to-speech. The human's verbal response is transcribed and updated in memory, enabling a more informed search towards the target (#4) that avoids unvisited areas (#1 and #2 frontiers in the figure) unrelated to the goal and improves efficiency.

4. Experiments

We evaluate ReasonNav across a suite of real and simulated navigation tasks. We seek to answer the following questions: 1) *Can our VLM leverage higher-order skills to avoid wrong searches?* 2) *How does ReasonNav perform in unseen real-world navigation tasks?* 3) *How do sign-reading and human interactions impact navigation efficiency?* 4) *How does map visualization input influence the VLM's spatial understanding?* We answer these questions through a variety of qualitative and quantitative analyses of the system's performance in comparison to relevant baselines.

Task description. Our evaluation tasks are designed to mimic a realistic indoor delivery scenario. The agent is placed in a large unknown building and is tasked with finding a target room specified by a room number. The episode is considered successful if the target room label has been read by a VLM call, within a 15-minute time limit.

Real-world environment. In the real-world, we consider two complex university campus multi-purpose buildings, each over 80m in length. For most of the rooms, including the target room, there is a room label next to each door to the room. Each building contains signs and people scattered throughout, which all provide helpful information. We evaluate navigation performance over 12 trials, each with different start and goal locations.

Simulation environment. We construct an environment in simulation (IsaacSim) to enable reproducible evaluation. To the best of our knowledge, there is no existing simulation environment suitable for evaluating higher-order navigation skills in realistic man-made scenes. We use existing assets for an empty hospital and add room labels for each door, directional signs, and virtual humans who can provide directions via hard-coded conversational responses. We evaluate navigation performance over 14 trials, each with different start and goal locations. We plan to release all the code and assets needed for evaluation in this environment to accelerate future research on practical indoor navigation. Please refer to Fig. 6 for visualization of our simulation environment.

Baselines. To the best of our knowledge, there is no existing method available for navigating to specific rooms within buildings, as the task inherently requires integrating text reading capabilities into the navigation pipeline. Thus, we design baselines which can be thought of as ablations of our method. To determine the impact of higher-order navigation skills on navigation efficiency, we create one baseline in which signs and people are not processed into the landmark memory bank (No Signs/Humans Feedback, Fig. 7). Thus, the VLM has no option to read signs or ask people for more information – it only sees map frontiers and doors and decides which to visit. We also experiment with removing the map image input to the VLM (No Landmark Map Input, Fig. 7). In this case, the VLM only receives scene information via JSON text format.

Metrics. We measure success rate, average episode duration, and average distance traveled. A success is counted when the robot reaches the goal and recognizes that it has completed the task after reading the room number. For failed episodes (due to collisions or timeout after 15 minutes), we assign a maximum duration of 900s and maximum distance traveled of 100m as a penalty.

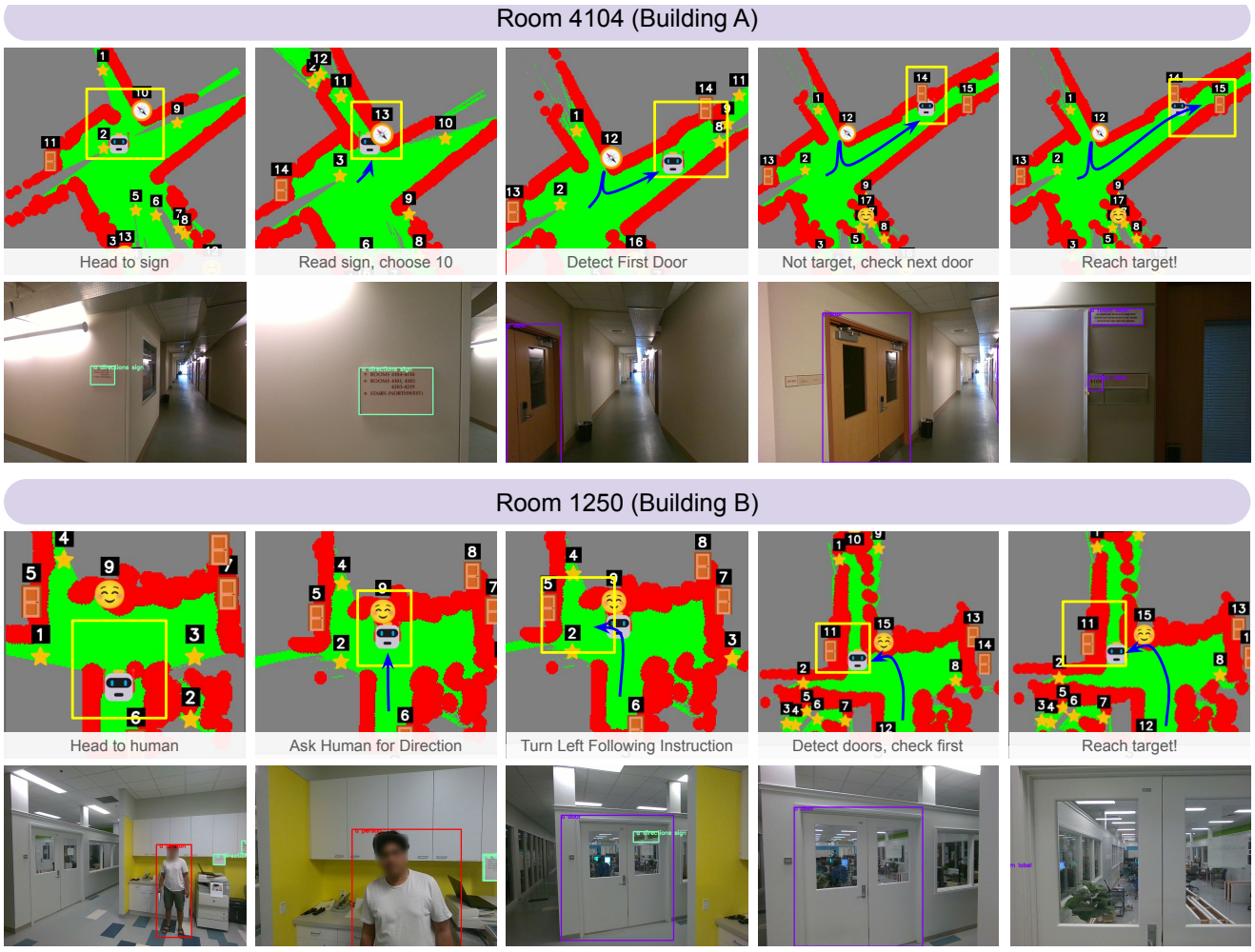

Figure 5. **Qualitative Results**: We present full step-by-step episode visualizations of our framework in two different real-world buildings. Thanks to its ability to reason over many sources of information, ReasonNav can accurately and efficiently navigate to the specified room number. Blue lines indicate the estimated traveled trajectories.

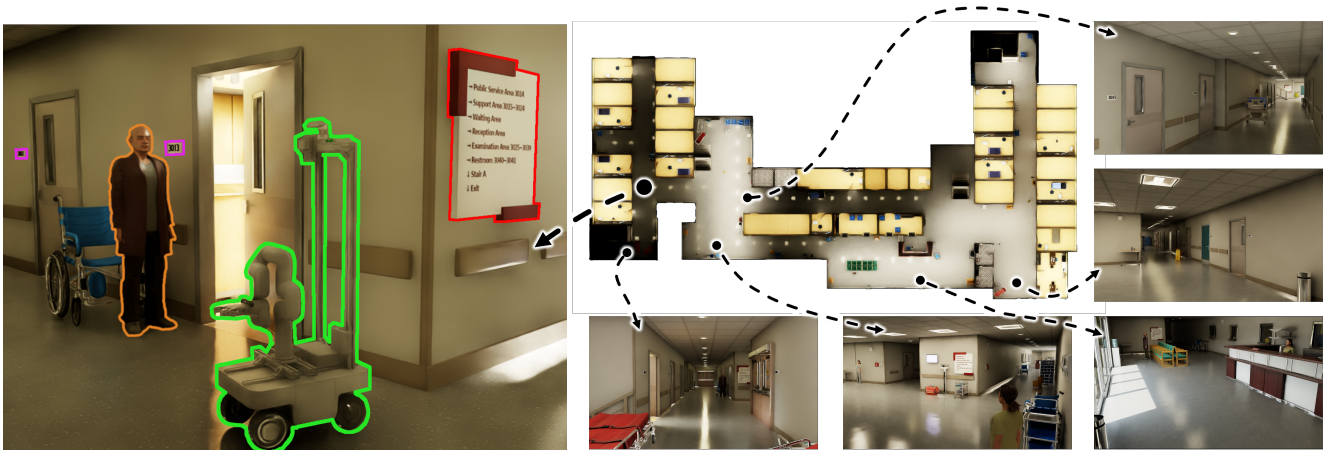

Figure 6. **Hospital Environment Visualization.** Existing open-world navigation benchmarks do not support large-scale building navigation tasks with human interaction. To fill this gap, we introduce an IsaacSim-based interactive navigation benchmark in a photorealistic hospital with over 30 rooms (offices, operation, examination, and patient rooms). The environment features realistic objects and layouts, informative signs, traversable rooms, and NPCs for human–robot interaction. We also provide a queryable website with an online staff directory.

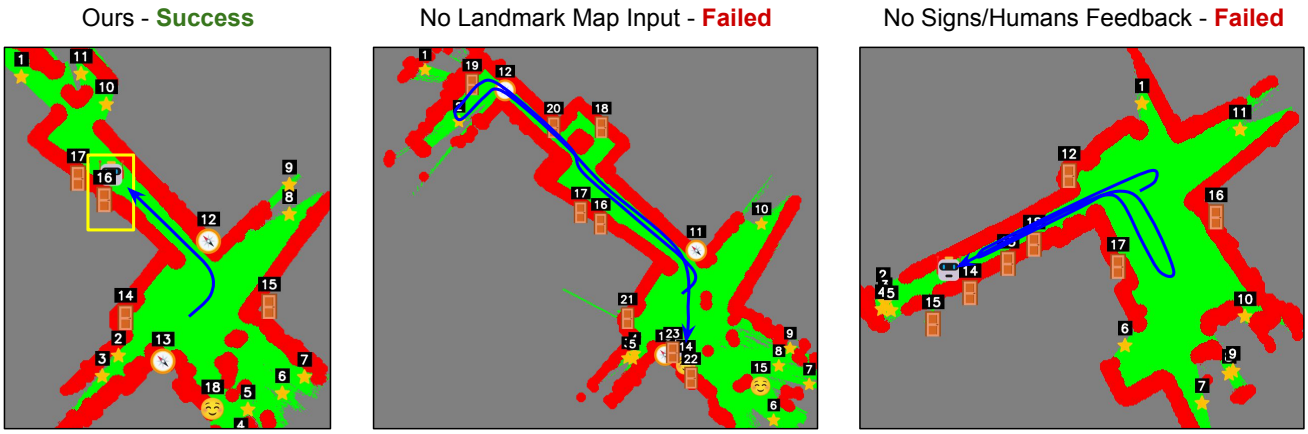

Figure 7. **Qualitative comparison with baselines.** We compare our method with ablative baselines to validate our visual prompting design and the importance of sign reading and communicating with humans. The visual map prompting enhances the spatial reasoning capabilities of the VLM, while the sign reading and communication gathers important directional information.

Table 1. Quantitative Results for Navigation in Real-World Environments (Academic Complexes)

	Success Rate (%)		Avg Duration (s)		Distance Traveled (m)		
Environment	Build A	Build B	Build A	Build B	Build A	Build B	Avg. (%)
No Signs/People	10	0	817.00	900	90.48	100	8.3
No Map Image	20	0	679.72	900	73.52	100	16.6
Ours	**50**	**100**	**572.35**	**232.63**	**60.28**	**12.61**	**58.3**

Table 2. Quantitative Results for Navigation in Simulation Environments (Large Hospital)

Environment	Success Rate (%)	Duration Traveled (s)	Distance Traveled (m)
No Signs/People	46.15	**710.76**	75.56
No Map Image	14.29	860.72	123.95
Ours	**57.14**	746.78	**72.53**

4.1. Qualitative Results

Real-world Results We provide step-by-step episode visualizations of ReasonNav's behavior in the real world in Fig. 5. Note that in each example, there are landmarks in many different directions that the agent can choose from. Choosing to explore in a direction that does not lead to the goal may result in wasting time by exploring very long hallways. We observe that our VLM agent is able to successfully read signs, interpret their directions with respect to the provided map, and use the information to pick frontiers that directly lead to the goal. Similarly, the agent can ask people for directions, record the received information in its memory bank, and use it effectively in subsequent high-level planning steps.

Simulated Results Along with the real-world demos, we demonstrate ReasonNav's performance in our simulated environment in Fig. 8. As our simulation is enriched with various NPC dialogues and directional signs to mimic the real-world, ReasonNav is able to leverage them to navigate succesfully.

Comparison with ablative baselines We compare our method with the aforementioned baselines qualitatively in Fig. 7. Removing the map image input significantly hinders the VLM's spatial reasoning capabilities, making it more likely to misunderstand which doors are close to the agent and are worth visiting. This confirms that modern VLMs are able to interpret top-down map images and use them for planning. On the other hand, removing the ability to read signs and ask people for directions makes the agent more likely to go in a completely wrong direction, causing failure due to timeout.

Step-by-step VLM reasoning We further showcase ReasonNav's reasoning capabilities in a step-by-step example in Fig. 9 of a real-world scenario. Given the landmark map (on the left), the VLM is prompted to choose a waypoint for the robot to follow. The text boxes (on the right) repre-

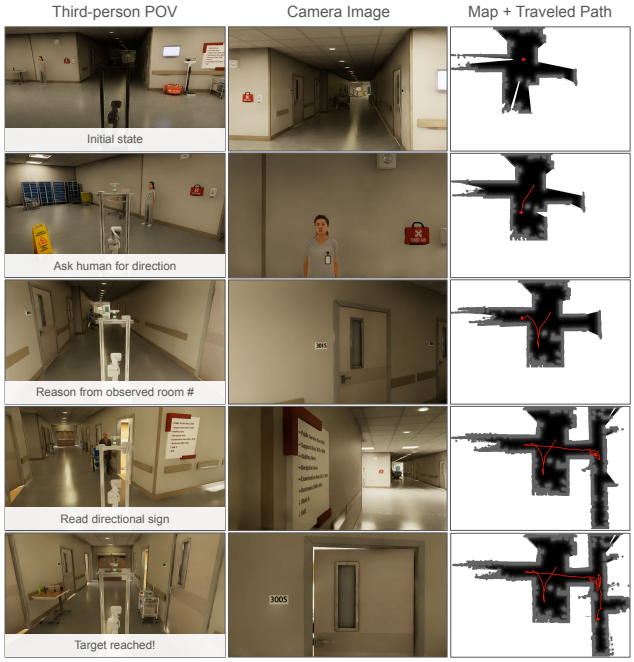

Figure 8. **Qualitative Simulation Results**: We present full step-by-step episode visualizations of our framework in simulation with exact path traveled highlighted in red.

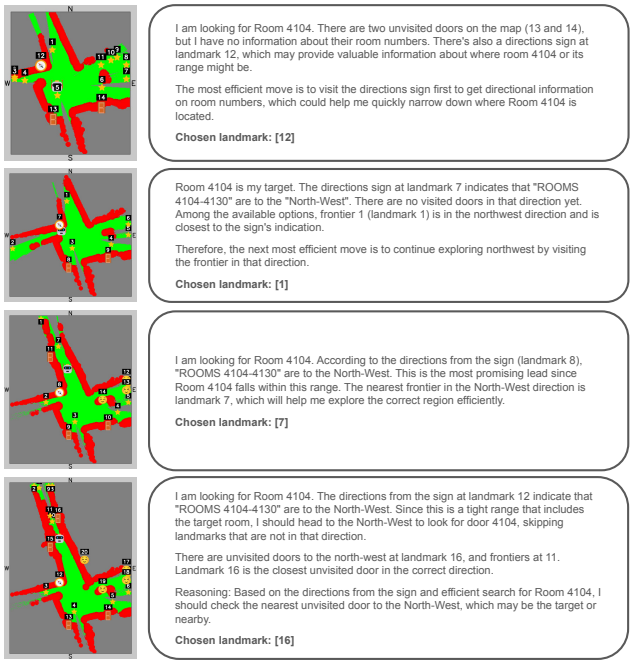

Figure 9. **Real-world VLM reasoning**: We present a step-by-step example of VLM's reasoning and decisions to navigate to room 4104. ReasonNav exhibits spatial reasoning capabilities given the direction guidance from direction signs, as showcased in the third and fourth rows.

sent the VLM's reasoning and decisions. We observe that ReasonNav can make reasonable decisions given the information, which successfully leading it to the goal.

4.2. Quantitative Results

We report quantitative results in both real-world and simulation environments (Tables 1 and 2). The results reveal several key insights. First, higher-order navigation skills—reading signs and asking people for directions—are critical: without these skills, the agent succeeds in only 8.3% of real-world trials and 46% of simulation trials. Second, omitting high-level reasoning via our unified spatial memory map as image prompts significantly degrades performance, underscoring its importance for decision making. In real-world experiments, baselines without skills or map prompting exhaust their time budgets, illustrating that missing high-level cues leads to timeout failures. Finally, by combining sign-reading, direction-asking, and spatial-map integration, our approach outperforms both the "No Signs/People" and "No Map Image" baselines by over 40 percentage points—achieving 58.33% in real-world trials and 57.14% in simulation.

5. Conclusion

We presented ReasonNav, a novel method for robot navigation that incorporates human-like navigation skills, such as sign reading and asking for directions, in an agentic VLM framework. ReasonNav abstracts low-level perceptual inputs into a memory bank of landmarks and uses a VLM to perform higher-order reasoning on these landmarks and plan high-level actions. We conduct experiments to validate the capabilities of the agent and show that higher-order navigation skills are important for efficient navigation in large buildings.

6. Limitations

ReasonNav successfully exhibits human-like navigation behaviors based on higher-order reasoning. However, since the system relies on an object detector to produce landmarks, the overall performance is bottlenecked by the detection performance. In addition, since the objects are limited to a predefined set of categories and the VLM only observes landmarks, the high-level planning does not maximally use the information contained in the camera observations. In the future, as detection capabilities become better integrated into VLMs themselves, the specialized detector could be replaced by a more powerful VLM-based detection stream. Lastly, because the VLM is restricted to exploring frontiers, it is not able to choose closer waypoints which may be sufficient for exploring an area while taking less time to reach. This could be mitigated by incorporating more sophisticated re-planning logic.

References

[1] Wenzhe Cai, Siyuan Huang, Guangran Cheng, Yuxing Long, Peng Gao, Changyin Sun, and Hao Dong. Bridging zero-shot object navigation and foundation models through pixel-guided navigation skill. In *ICRA*, 2023. 2

[2] Matthew Chang, Theophile Gervet, Mukul Khanna, Sriram Yenamandra, Dhruv Shah, So Min, Kavit Shah, Chris Paxton, Saurabh Gupta, Dhruv Batra, Roozbeh Mottaghi, Jitendra Malik, and Devendra Chaplot. Goat: Go to any thing. In *RSS*, 2024. 2

[3] Boyuan Chen, Fei Xia, Brian Ichter, Kanishka Rao, Keerthana Gopalakrishnan, Michael S Ryoo, Austin Stone, and Daniel Kappler. Open-vocabulary queryable scene representations for real world planning. In *ICRA*, 2023. 2

[4] Peihao Chen, Xinyu Sun, Hongyan Zhi, Runhao Zeng, Thomas H. Li, Gaowen Liu, Mingkui Tan, and Chuang Gan. a^2nav: Action-aware zero-shot robot navigation by exploiting vision-and-language ability of foundation models. *arXiv*, 2023. 2

[5] Hao-Tien Lewis Chiang, Zhuo Xu, Zipeng Fu, Mithun George Jacob, Tingnan Zhang, Tsang-Wei Edward Lee, Wenhao Yu, Connor Schenck, David Rendleman, Dhruv Shah, Fei Xia, Jasmine Hsu, Jonathan Hoech, Pete Florence, Sean Kirmani, Sumeet Singh, Vikas Sindhwani, Carolina Parada, Chelsea Finn, Peng Xu, Sergey Levine, and Jie Tan. Mobility vla: Multimodal instruction navigation with long-context vlms and topological graphs. In *CoRL*, 2024. 2

[6] Yuchen Cui, Siddharth Karamcheti, Raj Palleti, Nidhya Shivakumar, Percy Liang, and Dorsa Sadigh. No, to the right: Online language corrections for robotic manipulation via shared autonomy. In *HRI*, 2023. 2

[7] Yinpei Dai, Run Peng, Sikai Li, and Joyce Chai. Think, act, and ask: Open-world interactive personalized robot navigation. *ICRA*, 2024. 2

[8] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. In *ECCV*, 2024. 4

[9] 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. *CVPR*, 2023. 2

[10] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *ICRA*, 2024. 2

[11] Chenguang Huang, Oier Mees, Andy Zeng, and Wolfram Burgard. Visual language maps for robot navigation. In *ICRA*, 2023. 2

[12] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Noah Brown, Tomas Jackson, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models. In *CoRL*, 2022. 2

[13] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. In *CoRL*, 2023. 2

[14] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, Ayush Tewari, Joshua B. Tenenbaum, Celso Miguel de Melo, Madhava Krishna, Liam Paull, Florian Shkurti, and Antonio Torralba. Conceptfusion: Open-set multimodal 3d mapping. *RSS*, 2023. 2

[15] Hanxiao Jiang, Binghao Huang, Ruihai Wu, Zhuoran Li, Shubham Garg, Hooshang Nayyeri, Shenlong Wang, and Yunzhu Li. Roboexp: Action-conditioned scene graph via interactive exploration for robotic manipulation. In *CoRL*, 2024. 2

[16] Boyi Li, Philipp Wu, Pieter Abbeel, and Jitendra Malik. Interactive task planning with language models. *TMLR*, 2023. 2

[17] Huihan Liu, Alice Chen, Yuke Zhu, Adith Swaminathan, Andrey Kolobov, and Ching-An Cheng. Interactive robot learning from verbal correction. *arXiv*, 2023. 2

[18] Yuxing Long, Wenzhe Cai, Hongcheng Wang, Guanqi Zhan, and Hao Dong. Instructnav: Zero-shot system for generic instruction navigation in unexplored environment. In *CoRL*, 2024. 2

[19] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *ICLR*, 2024. 4

[20] Steve Macenski and Ivona Jambrecic. Slam toolbox: Slam for the dynamic world. *Journal of Open Source Software*, 6 (61):2783, 2021. 4

[21] Steve Macenski, Francisco Martín, Ruffin White, and Jonatan Ginés Clavero. The marathon 2: A navigation system. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. 4

[22] Nur (Mahi)Shafiullah, Chris Paxton, Lerrel Pinto, Soumith Chintala, and Arthur Szlam. Clip-fields: Weakly supervised semantic fields for robotic memory. In *RSS*, 2023. 2

[23] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection. In *ECCV*, 2022. 4

[24] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *CVPR*, 2023. 2

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2

[26] Abhinav Rajvanshi, Karan Sikka, Xiao Lin, Bhoram Lee, Han-Pang Chiu, and Alvaro Velasquez. Saynav: Grounding

large language models for dynamic planning to navigation in new environments. In *ICAPS*, 2024. 2

[27] Allen Z Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, et al. Robots that ask for help: Uncertainty alignment for large language model planners. In *CoRL*, 2023. 2

[28] Allen Z Ren, Jaden Clark, Anushri Dixit, Masha Itkina, Anirudha Majumdar, and Dorsa Sadigh. Explore until confident: Efficient exploration for embodied question answering. In *RSS*, 2024. 2

[29] Dhruv Shah, Blazej Osinski, Brian Ichter, and Sergey Levine. LM-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *CoRL*, 2022. 2

[30] Dhruv Shah, Ajay Sridhar, Arjun Bhorkar, Noriaki Hirose, and Sergey Levine. Gnm: A general navigation model to drive any robot. In *ICRA*, 2023. 2

[31] 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)*. IEEE, 2025. 2

[32] Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. Vision-and-dialog navigation. In *CoRL*, 2019. 2

[33] Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical open-vocabulary 3d scene graphs for language-grounded robot navigation. *RSS*, 2024. 2

[34] Grady Williams, Paul Drews, Brian Goldfain, James M Rehg, and Evangelos A Theodorou. Aggressive driving with model predictive path integral control. In *ICRA*, 2016. 4

[35] 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 *ICML*, 2024. 2

[36] Bangguo Yu, Hamidreza Kasaei, and Ming Cao. L3mvn: Leveraging large language models for visual target navigation. In *IROS*, 2023. 2

[37] Michał Zawalski, William Chen, Karl Pertsch, Oier Mees, Chelsea Finn, and Sergey Levine. Robotic control via embodied chain-of-thought reasoning. In *CoRL*, 2024. 2

[38] Lihan Zha, Yuchen Cui, Li-Heng Lin, Minae Kwon, Montserrat Gonzalez Arenas, Andy Zeng, Fei Xia, and Dorsa Sadigh. Distilling and retrieving generalizable knowledge for robot manipulation via language corrections. In *ICRA*, 2024. 2

[39] 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 *ICML*, 2023. 2