

# ScholarWorks@GSU

## Towards Vision and Language Models Aided Object Navigation

Authors	Liu, Weizhen
Citation	Liu, Weizhen (2024). "Towards Vision and Language Models Aided Object Navigation." Thesis, Georgia State University. <a href="https://doi.org/10.57709/36972831">https://doi.org/10.57709/36972831</a>
DOI	<a href="https://doi.org/10.57709/36972831">https://doi.org/10.57709/36972831</a>
Download date	2026-04-11 02:24:01
Link to Item	<a href="https://hdl.handle.net/20.500.14694/4110">https://hdl.handle.net/20.500.14694/4110</a>

Towards Vision and Language Models Aided Object Navigation

by

Weizhen Liu

Under the Direction of Jonathan Shihao Ji, Ph.D.

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Science

in the College of Arts and Sciences

Georgia State University

2024

## ABSTRACT

In this work, we present a novel hierarchical navigation policy for object navigation that leverages both object detection models and large language models (LLMs) to enhance the interpretation and interaction with complex indoor environments. Our approach integrates object detection to accurately assess the surrounding space and employs a layout reconstruction strategy to model the environment’s structure. By defining our navigation strategy hierarchically, we separate the decision-making into long-term and short-term goals, effectively utilizing the existing concept of ”frontier-based goal selection.” We refine this method by representing frontiers through a series of observations transformed into language via object detection models. Each frontier is then scored using LLMs, allowing for a reasoned selection of the most promising navigational targets. Our framework, simple yet effective, not only aligns with the demands of dynamic and unknown environments but also surpasses existing baselines in terms of efficiency and accuracy, offering significant advancements in the field of robotic navigation. Code can be found at <https://github.com/weizhenFrank/ObjNav>.

INDEX WORDS: Object navigation, LLMs, Frontier representation, Decision-making

Copyright by  
Weizhen Liu  
2024

Towards Vision and Language Models Aided Object Navigation

by

Weizhen Liu

Committee Chair:

Jonathan Shihao Ji

Committee:

Raj Sunderraman

Xiaojun Cao

Electronic Version Approved:

Office of Graduate Services

College of Arts and Sciences

Georgia State University

May 2024

## DEDICATION

To my beloved sister and brother-in-law,

Your unwavering support and encouragement have been the pillars upon which I've built my determination and perseverance. The countless sacrifices you've made and the faith you've shown in me have been the driving force behind my journey. Your belief in my abilities has been a constant reminder of the strength and love that surrounds me, propelling me forward through every challenge.

And to Christine,

Your presence in my life has been a source of joy and comfort. The happiness we've shared and the memories we've created together have provided me with the balance and motivation needed to pursue my dreams with vigor. Your love has been my sanctuary, a place of peace and inspiration amidst the whirlwind of academic pursuit.

As I stand on the precipice of this significant milestone, graduating with a Master's degree in Computer Science in May 2024, I look back on the journey that began in August 2022 with gratitude and love. This achievement is not solely mine; it belongs to us all. It is a testament to the power of support, love, and belief in one another.

Thank you for being my guiding stars, my comfort, and my strength. This dedication is but a small token of my immense gratitude and love for you.

## ACKNOWLEDGMENTS

My deepest gratitude goes to Dr. Jonathan Shihao Ji, whose guidance and support throughout my Master's program were invaluable. His mentorship in the fascinating world of Embodied AI, particularly working with Spot from Boston Dynamics, provided a rich, inspiring learning experience. Dr. Ji's lab was not just a place of study but a foundation for innovation and growth.

I am equally thankful to my lab mates, Qing Su, Yang Li, and Hui Ye. Their friendship, shared wisdom, and support made our time in the lab both productive and enjoyable, significantly enriching my educational journey.

My heartfelt thanks also extend to the professors whose courses have deeply enhanced my understanding and passion for computer science. Their rigorous academic standards and supportive teaching methods have been instrumental in my development.

This journey, marked by the collective wisdom and encouragement of my mentors, instructors, and peers, has been transformative. I am profoundly grateful for their contributions to my academic and personal growth.

## TABLE OF CONTENTS

ACKNOWLEDGMENTS . . . . .	v
LIST OF TABLES . . . . .	viii
LIST OF FIGURES . . . . .	ix
<b>1 INTRODUCTION . . . . .</b>	<b>1</b>
<b>2 RELATED WORKS . . . . .</b>	<b>3</b>
2.1 Visual Navigation . . . . .	3
2.2 Large Language Model-Guided Navigation . . . . .	4
<b>3 METHOD . . . . .</b>	<b>6</b>
3.1 Task Definition and Overview . . . . .	6
3.2 Layout Map . . . . .	7
3.2.1 <i>Layout Map Representation</i> . . . . .	7
3.3 Frontier Calculation Module . . . . .	8
3.3.1 <i>Frontier Map</i> . . . . .	8
3.3.2 <i>Contextual Frontier Representation</i> . . . . .	8
3.4 LLM-based Exploration Policy . . . . .	10
3.4.1 <i>Language Description of Frontiers</i> . . . . .	10
3.4.2 <i>LLM Scoring and Frontier Selection</i> . . . . .	10
3.5 Local Policy and Navigation . . . . .	11
<b>4 EXPERIMENTS . . . . .</b>	<b>12</b>
4.1 Simulation Setup . . . . .	12
4.1.1 <i>Datasets</i> . . . . .	12
4.1.2 <i>Implementation Details</i> . . . . .	12

4.2	Evaluation Criteria . . . . .	13
4.2.1	<i>Success Rate (SR)</i> . . . . .	13
4.2.2	<i>Success weighted by Path Length (SPL)</i> . . . . .	13
4.2.3	<i>Distance to Goal (DTG)</i> . . . . .	13
4.3	Results . . . . .	14
4.3.1	<i>Quantitative comparison</i> . . . . .	14
4.3.2	<i>Ablation Study</i> . . . . .	15
5	CONCLUSION . . . . .	17
	REFERENCES . . . . .	20

**LIST OF TABLES**

Table 4.1	Compare to other methods . . . . .	16
Table 4.2	Ablation study . . . . .	16

## LIST OF FIGURES

<p>Figure 3.1 The methodological framework illustrating the workflow for object navigation. RGB-D inputs are used to reconstruct a layout map of the environment. The frontier calculation module processes the map to identify and represent candidate frontiers. The LLM scores these frontiers relative to the target object’s category, and the local path planning component then guides the agent to the selected frontier, which is deemed most likely to contain the target object. . . . .</p>	7
<p>Figure 3.2 Visualization of the frontier calculation process within the layout map. The red circle denotes the positions of the agent, while the dotted line illustrates the trajectory of the agent’s path as it explores the environment. The blue shaded areas represent the triangular regions covered by the agent’s field of view from different positions. ‘Frontier X’ marks an example of a detected frontier cluster, showcasing how observations are spatially related and associated with potential exploration targets within the map. . . . .</p>	9

## CHAPTER 1

### INTRODUCTION

Object navigation is a critical task in the domains of robotics and embodied AI, requiring an agent to autonomously navigate through unknown environments to locate specific objects. This capability is essential for robots to interact effectively with their surroundings and perform a variety of practical tasks. Despite its significance, object navigation often faces challenges, particularly in operating within diverse, unseen environments without prior exposure (zero-shot generalization) and in handling objects in varied configurations and contexts.

Recent developments in foundational AI models, particularly large language models (LLMs) and object detection technologies, have shown substantial promise in overcoming these hurdles. LLMs are renowned for their advanced commonsense reasoning abilities (5; 13), which can significantly enhance navigation and exploration strategies. Object detection models provide robust capabilities for recognizing and localizing objects within complex scenes (25; 17), facilitating precise interaction with the environment.

Our proposed framework leverages the precision of pre-trained object detection models combined with the reasoning prowess of LLMs to enable effective and efficient navigation in unseen settings. This method avoids the extensive training regimes typically associated with reinforcement learning approaches, using already trained models to understand the environment and make intelligent navigation decisions. Our system constructs detailed semantic maps from visual inputs, which inform LLM-driven decision-making processes. This ap-

proach focuses on selecting the most promising areas for exploration based on the semantic context and specific characteristics of the target objects.

We validate our approach through rigorous testing on the challenging HM3D (23) and Gibson (32) datasets, which simulate complex real-world environments where the agent must navigate to locate specific objects based on textual descriptions.

This innovative integration of object detection and LLMs marks a significant advancement in enabling robotic agents to navigate efficiently through novel environments and handle a variety of object scenarios seamlessly. Our work sets the stage for the next generation of object navigation systems, which are poised to offer enhanced assistance in a variety of real-world applications without the need for extensive retraining.

## CHAPTER 2

### RELATED WORKS

#### 2.1 Visual Navigation

Visual navigation is a crucial capability for robots to perform various embodied tasks (38; 2; 26). In these tasks, the agent receives a goal specification, such as a geometric target (26) or a semantic objective (8; 21), along with RGB observations from its camera. The agent must navigate to the goal based on these observations at each time step. To train visual navigation agents, simulators (26; 32) with photo-realistic environments (7; 30) and physical engines are commonly used. These simulators provide the agent with goal indications and corresponding trajectory annotations. However, labeling objects in 3D scenes is labor-intensive, and recent works (8; 35; 34) have explored designing object-goal navigation agents without human annotations, making them easily scalable (34) and potentially addressing the open set challenge in object localization.

Classic approaches to visual navigation involve map building, localization, and path planning (3). While some scenarios provide pre-built maps, allowing approaches like RTAB-Map (16) to perform localization and path planning, most real-world scenarios require simultaneous map building and localization using SLAM systems (6). Although classic methods like ORB-SLAM (20) or LSD-SLAM (14) perform well, there is a growing trend of incorporating differentiable models, such as deep neural networks, into SLAM systems (4; 12). Additionally, recent work has shown that directly training reactive policies using recurrent neural networks like GRUs (11) can achieve excellent performance without explicit map building

and path planning (8; 19).

## 2.2 Large Language Model-Guided Navigation

While language models have shown great potential in guiding visual navigation tasks by leveraging their powerful commonsense reasoning abilities (18; 31; 10), the integration of large language models (LLMs) in navigation has more recently emerged as a compelling approach. Various research efforts have explored the use of LLMs to enhance navigation strategies. For instance, LM-Nav (29) utilizes language models to extract landmarks from navigation instructions, enhancing path planning with these contextual cues. L3MVN (36) leverages LLMs to calculate entropy in frontier selection, while ESC (37) and StructNav (9) integrate LLMs for complex scene understanding and decision-making based on semantic maps.

A particularly relevant study (28) showcases the need for intensive prompt engineering to effectively employ LLMs in navigation tasks. These approaches, however, often involve complex setups and extensive prompt engineering to fine-tune the models for specific navigation tasks.

In contrast, our method simplifies the use of LLMs in navigation by employing pre-trained models in conjunction with straightforward object detection. By avoiding the complexity of prompt engineering and model fine-tuning, we streamline the process, making it more accessible and less resource-intensive. Our approach harnesses the inherent capabilities of LLMs to understand and reason about the environment based on simple, direct queries linked

with the semantic output from object detection models. This simplicity allows for robust navigation decisions in a variety of environments, demonstrating effectiveness without the overhead associated with more complex systems.

## CHAPTER 3

### METHOD

#### 3.1 Task Definition and Overview

The object navigation task requires an agent to navigate an environment to find an object belonging to a specified category. The category set is denoted by  $C = \{c_0, \dots, c_m\}$ , and the scene is represented by  $S = \{s_0, \dots, s_m\}$ . Each episode begins with the agent initialized at a random position  $p_i$  in the scene  $s_i$  and receives the target object category  $c_i$ . Thus, an episode can be denoted as  $T_i = \{s_i, c_i, p_i\}$ . At each time step  $t$ , the agent observes the environment and takes an action  $a_t$ . The observation includes RGB-D images, the agent’s location and orientation, and the target object category. The action space, denoted by  $A$ , includes six actions: move forward, turn left, turn right, look up, look down, and stop. The episode is considered successful if the agent takes the stop action when the distance to the target is less than 0.1m, with a maximum of 500 time steps per episode.

Our framework consists of four main components: a reconstructed layout map, a frontier calculation module, an LLM scoring part, and a local path planning method. The framework is shown in figure 3.1. The agent first obtains observations of the environment to build a layout map, then based on the layout map, the frontier calculation module extracts the frontiers and calculates the frontier representation. The LLM scoring part then selects a next frontier based representation of frontiers and the object goal. Finally, a local policy plans a path and takes actions to explore the environment and search for the target object.

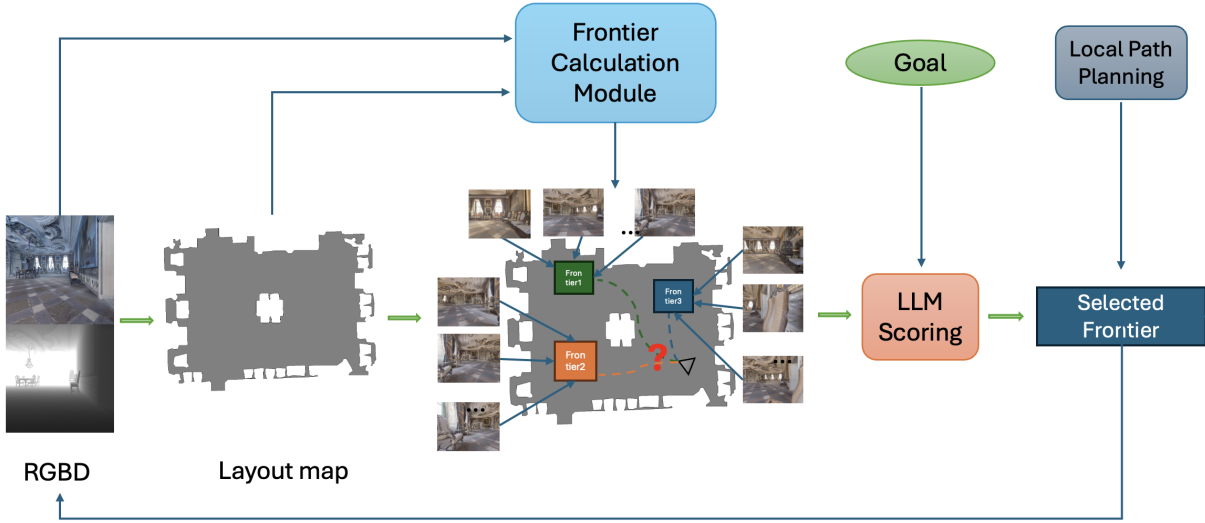


Figure 3.1: The methodological framework illustrating the workflow for object navigation.

RGB-D inputs are used to reconstruct a layout map of the environment. The frontier calculation module processes the map to identify and represent candidate frontiers. The

LLM scores these frontiers relative to the target object’s category, and the local path planning component then guides the agent to the selected frontier, which is deemed most likely to contain the target object.

## 3.2 Layout Map

### 3.2.1 Layout Map Representation

In our revised framework, we streamline the construction of the layout map by solely utilizing depth images coupled with the agent’s positional data. This approach deviates from methods that integrate semantic information within the map, such as those by Chaplot et al. (8), focusing instead on structural features of the environment. The layout map is structured as a  $K \times M \times M$  tensor, with  $M \times M$  denoting the map’s dimensions and  $K$  corresponding to the number of channels; however, in our case,  $K$  is limited to information relevant to navigation, specifically obstacle and explored space representations.

Point clouds derived from depth images are processed and transformed into a top-down 2D perspective, forming the foundational layers of the map. This includes an 'obstacle' channel, identifying the physical barriers within the environment, and an 'explored' channel, marking the regions within the agent's field of discovery. This binary layout map provides a clear and uncluttered representation of the environment, essential for the subsequent steps in the navigation process. The absence of semantic categorization in our layout map simplifies the input to the navigation system, reducing computational complexity and focusing on spatial understanding and traversal feasibility.

### 3.3 Frontier Calculation Module

#### 3.3.1 *Frontier Map*

The frontier map is obtained from the reconstructed layout map, similar to method in (36). We extract the explored edge by identifying the maximum contours from the explored map and generate the frontier map as the difference between the explored and obstacle maps. Frontier cells are clustered into chains, and small clusters are removed. The subset of candidate destinations  $F$  is composed of the cells in the center of the remaining cluster chains. Each frontier cell  $f \in F$  is scored using the cost-utility approach proposed in (36).

#### 3.3.2 *Contextual Frontier Representation*

In contrast to the previous approach of using individual items within each frontier, we propose a novel method to obtain a holistic description of the entire scene for each frontier. As the agent explores the environment, it acquires observations in the form of a triangular region

in the top-down 2D layout map. The vertex of the triangle represents the agent's location, and the angle corresponds to the agent's field of view. By leveraging the agent's GPS and compass information, we know the agent's location  $(x, y, z)$  and orientation. This allows us to determine which region each observation covers and construct the spatial distribution of the observations, so that we can associate a series of observations with different frontier, this method is shown in figure 3.2. When selecting which frontier to explore, we use a series of observations to represent each frontier, capturing the spatial relationships and context of the objects within the frontier.

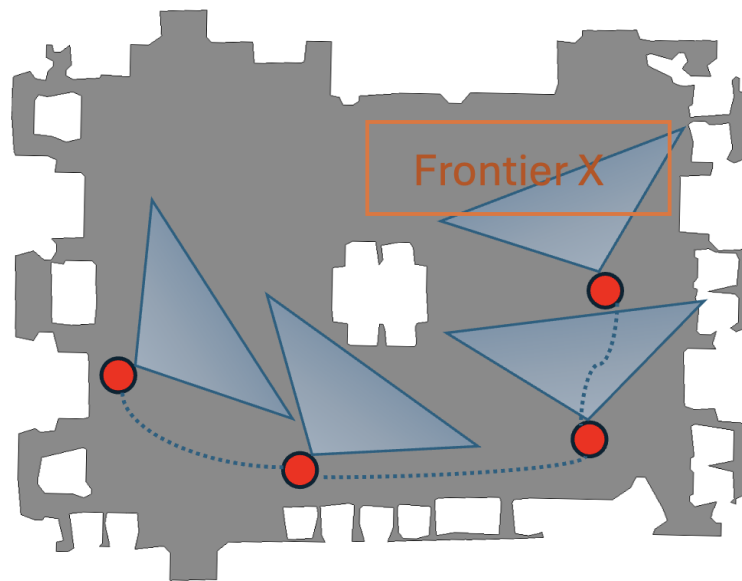


Figure 3.2: Visualization of the frontier calculation process within the layout map. The red circle denotes the positions of the agent, while the dotted line illustrates the trajectory of the agent's path as it explores the environment. The blue shaded areas represent the triangular regions covered by the agent's field of view from different positions. 'Frontier X' marks an example of a detected frontier cluster, showcasing how observations are spatially related and associated with potential exploration targets within the map.

### 3.4 LLM-based Exploration Policy

The selection of the most promising frontier for exploration is guided by an LLM-based exploration policy. This policy incorporates language descriptions derived from object detections within each frontier, followed by a language model’s assessment to predict the likelihood of the goal object’s presence. Below we outline the process in further detail.

#### *3.4.1 Language Description of Frontiers*

Utilizing object detection models, each observation within a frontier is analyzed to identify and list objects. These detected objects are then described in natural language, providing a descriptive summary that encompasses all observations associated with a particular frontier. This descriptive approach allows for a human-readable understanding of the frontier, turning visual data into a linguistic context that can be effectively processed by a language model.

#### *3.4.2 LLM Scoring and Frontier Selection*

Once a language description is generated, we employ a pre-trained language model to estimate the joint probability distribution between the detected objects and the goal object category across all frontiers. This process involves computing the likelihood that the goal object is present within the observed scene described by the language summary. The resulting probabilities inform our decision-making process, enabling the selection of a frontier for exploration. Similar to the method proposed by (36), we prioritize frontiers with a higher likelihood of containing the goal object, balancing the exploratory cost against the potential utility. The selection is made based on a calculated range of probabilities, ensuring a

strategic and informed approach to navigating the environment in search of the target object.

Through this LLM-based exploration policy, we leverage the rich semantics captured by language models to enhance the navigation process, without the need for extensive training or fine-tuning of the models on navigation-specific data. This approach underscores the adaptability and efficiency of using pre-trained models in a novel application domain.

### 3.5 Local Policy and Navigation

To navigate from the agent’s current location to a long-term goal, we employ the Fast Marching Method (FMM) (27). The agent selects a local goal within a restricted range of its current position and executes the final action  $a_t \in A$  to reach it. At each step, the local map and local goal are updated based on new observations.

By incorporating these frontier representation and processing techniques into the overall methodology, we aim to enhance the agent’s ability to navigate efficiently and make informed decisions based on the given goal and the contextual information available in the environment. Our proposed approaches offer several advantages, such as capturing contextual information, providing a richer representation, and being robust to sparsity compared to item-based representations used in previous methods.

## CHAPTER 4

### EXPERIMENTS

We present an empirical evaluation of our proposed method, which integrates object detection models with large language models, and compare its performance against established map-based navigation baselines within a simulated environment.

#### 4.1 Simulation Setup

##### *4.1.1 Datasets*

Our evaluation utilizes two high-resolution, photorealistic 3D reconstructions of real-world environments. The Gibson dataset includes 25 training and 5 validation scenes from the Gibson tiny split, offering semantic annotations for comprehensive testing. The HM3D dataset, formatted for Habitat, comprises standard splits of 75 training and 20 validation scenes. Six object categories are specified for the navigation task: chair, couch, potted plant, bed, toilet, and TV, as identified in prior work (8).

##### *4.1.2 Implementation Details*

We employ the Habitat simulator (26) for our 3D indoor navigation experiments, utilizing an observation space of  $480 \times 640$  RGBD images, paired with a base odometry sensor. The target objects are represented as integer categories. Our framework uses GPT-2 (22) for its effective language modeling capabilities in providing likelihood estimations, and YOLO v8 (15) for object detection, striking a balance between inference speed and accuracy.

## 4.2 Evaluation Criteria

Our evaluation is guided by the following metrics, commonly accepted in the domain of navigation research:

### 4.2.1 *Success Rate (SR)*

Defined as the proportion of episodes where the agent successfully stops within a threshold distance from the target object. It is calculated as  $\frac{1}{N} \sum_{i=1}^N S_i$ , with  $S_i$  being the success indicator of episode  $i$ .

### 4.2.2 *Success weighted by Path Length (SPL)*

A metric that balances the success rate against the optimality of the path taken by the agent. It considers both the success of the navigation and the efficiency of the path relative to the shortest possible path. SPL is computed as  $\frac{1}{N} \sum_{i=1}^N S_i \frac{\max(l_i, p_i)}{l_i}$ , where  $l_i$  denotes the shortest path length to the goal and  $p_i$  is the path length traversed by the agent in episode  $i$ .

### 4.2.3 *Distance to Goal (DTG)*

This metric measures the final distance between the agent and the target object at the conclusion of an episode, providing insight into the closeness of the agent to the goal regardless of the episode's success.

## 4.3 Results

### 4.3.1 Quantitative comparison

The efficacy of our object navigation framework was evaluated against various established methods in the Gibson and HM3D datasets. Table 4.1 presents a quantitative comparison of our method with several baselines.

Our method outperforms other approaches in terms of the Success Rate (SR), Success weighted by Path Length (SPL), and Distance to Goal (DTG) across both the Gibson and HM3D datasets. In the Gibson dataset, we achieved an SR of 0.785 and an SPL of 0.396, with the DTG reduced to 0.747m. For the HM3D dataset, our method reached an SR of 0.512 and an SPL of 0.240, with a notable decrease in DTG to 0.401m.

Notably, our approach demonstrates a significant improvement over the L3MVN zero-shot method, which was the closest competitor. The improved performance can be attributed to the effective frontier calculation module and LLMs scoring method, which work in conjunction to provide a robust representation of frontiers and an intelligent selection strategy that closely aligns with the target goals.

The simplicity of our method, which eschews complex prompt engineering and fine-tuning usually associated with LLMs, suggests that leveraging pre-trained models in a straightforward, yet innovative manner can yield substantial benefits in navigation tasks. Moreover, the use of YOLO v8 for object detection proves to be advantageous in terms of both speed and accuracy, as evidenced by the decreased DTG compared to other methods.

### 4.3.2 Ablation Study

An ablation study was conducted on the Gibson dataset to evaluate the individual contributions of the frontier calculation module and the LLM scoring method to the overall performance of our navigation system.

Firstly, to demonstrate the efficacy of the frontier calculation module, we conducted an experiment where the specific observations associated with each frontier were replaced with random RGB images from the scene. This variant, referred to as "Random RGB," serves to confirm the importance of the spatially relevant observations that are typically used to represent the frontiers.

Secondly, to assess the impact of our LLM scoring method, we introduced a baseline named "VLM-query" that utilizes a vision-language model (GPT-4V) (1). In this baseline, the series of observations for each frontier are fed directly into the model, which is then prompted to select the optimal frontier relative to the navigation goal.

The results of the ablation study, detailed in Table 4.2, illustrate a clear hierarchy in performance, with our method outstripping the alternative approaches in all metrics. The decrease in success rate and SPL, and the increase in DTG for "Random RGB," validate the significance of using contextual observations for frontier representation. Similarly, the "VLM-query" method's lower metrics as compared to our full method underscore the advantage of employing LLM for frontier scoring over vision-language models in this specific task.

Table 4.1: Compare to other methods

Method	Gibson			HM3D		
	Success	SPL	DTG	Success	SPL	DTG
Random Walking	0.030	0.030	2.580	0.000	0.000	7.600
Frontier Based Method(33)	0.417	0.214	2.634	0.237	0.123	5.414
Random Sample on Map	0.544	0.288	1.918	0.300	0.143	4.761
SemExp(8)	0.652	0.336	1.520	0.379	0.188	2.943
PONI(24)	0.736	0.410	-	-	-	-
L3MVN (Zero-Shot)	0.761	0.377	1.101	0.504	0.231	4.427
<b>Ours</b>	<b>0.784</b>	<b>0.393</b>	<b>0.750</b>	<b>0.512</b>	<b>0.240</b>	<b>0.401</b>

Table 4.2: Ablation study

Method	Success	SPL	DTG
Random RGB	0.549	0.102	2.097
VLM-query	0.739	0.330	1.118
<b>Our Method</b>	<b>0.784</b>	<b>0.393</b>	<b>0.750</b>

## CHAPTER 5

### CONCLUSION

In this paper, we introduced an innovative object navigation framework that intelligently leverages the strengths of pre-trained object detection models and large language models (LLMs). By adopting a simplified yet effective approach, our method has demonstrated a notable advancement in navigating agents through unknown environments to locate specified objects. Our comprehensive experiments on the Gibson and HM3D datasets have established the effectiveness of our approach, which forgoes the need for complex prompt engineering or extensive fine-tuning typically associated with LLMs. The proposed method outperforms several established baselines by considering a strategic combination of layout map construction, frontier representation, and scoring frontiers through the utilization of object detection outputs and LLM capabilities. The ablation studies further reinforced the significance of each component in our framework, highlighting how contextually relevant representations and the intelligent scoring of frontiers contribute to superior navigation performance. Our results show promise for real-world applications where autonomous agents are expected to understand and interact with their environment dynamically and effectively.

**Limitations and Future Work.** Despite the success of our method in simulated environments, several challenges remain before it can be effectively implemented in real-world scenarios. For instance, the method has not been tested against the variability and unpredictability of real-world dynamics, including changing lighting conditions, moving obstacles, and unmodeled environmental factors. Future research will aim to address these challenges

by incorporating real-time sensory feedback and improving the robustness of the models to such changes. Additionally, integrating our approach with more complex multimodal data and testing on physical robotic platforms will be crucial for assessing its practical applicability and performance in real-world conditions. These steps will help bridge the gap between simulation-based testing and real-world deployment, paving the way for more autonomous and intelligent robotic systems.

Further considerations in future work include addressing the limitations related to object detection and agent behavior. The current method detects and acts on the first object found, without consideration for proximity. Future research will aim to enhance object recognition to allow for more detailed targeting, like finding the closest chair instead of the first detected. This enhancement would improve interaction with real-world environments where object placement is dynamic and unpredictable.

Another area for improvement is time optimization in agent movements. The current system maintains a uniform speed, but in certain situations, such as navigating through a hallway, the agent could increase speed to save time. Future development will focus on creating adaptive speed controls, allowing agents to respond to environmental contexts and complete tasks more efficiently.

Lastly, our method currently constructs a new layout map at the beginning of each episode, even when it starts in a familiar scene. This approach can be resource-intensive and time-consuming. Future work will explore the reuse of previously constructed layout maps when appropriate, allowing for faster initialization and more efficient memory use. This

improvement will facilitate smoother transitions between episodes and contribute to a more seamless experience in long-term, continuous deployment scenarios.

## REFERENCES

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Floren-  
cia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anad-  
kat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- [2] Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh  
Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Mano-  
lis Savva, et al. On evaluation of embodied navigation agents. *arXiv preprint  
arXiv:1807.06757*, 2018.
- [3] Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping (slam):  
Part ii. *IEEE robotics & automation magazine*, 13(3):108–117, 2006.
- [4] Michael Bloesch, Jan Czarnowski, Ronald Clark, Stefan Leutenegger, and Andrew J  
Davison. Codeslam—learning a compact, optimisable representation for dense visual  
slam. In *Proceedings of the IEEE conference on computer vision and pattern recognition*,  
pages 2560–2568, 2018.
- [5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla  
Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini  
Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya  
Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric  
Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner,  
Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models

- are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.
- [6] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332, 2016.
- [7] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *2017 International Conference on 3D Vision (3DV)*, pages 667–676. IEEE, 2017.
- [8] 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.
- [9] Junting Chen, Guohao Li, Suryansh Kumar, Bernard Ghanem, and Fisher Yu. How to not train your dragon: Training-free embodied object goal navigation with semantic frontiers. *arXiv preprint arXiv:2305.16925*, 2023.
- [10] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer, 2020.
- [11] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Fethi Bougares, Holger

- Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*, 2014.
- [12] Jan Czarnowski, Tristan Laidlow, Ronald Clark, and Andrew J Davison. Deepfactors: Real-time probabilistic dense monocular slam. *IEEE Robotics and Automation Letters*, 5(2):721–728, 2020.
- [13] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. In *International Conference on Machine Learning*, pages 8469–8488. PMLR, 2023.
- [14] Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *European conference on computer vision*, pages 834–849. Springer, 2014.
- [15] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics yolov8, 2023.
- [16] Mathieu Labbé and François Michaud. Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *Journal of field robotics*, 36(2):416–446, 2019.
- [17] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016.
- [18] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural infor-*

- mation processing systems*, 32, 2019.
- [19] Piotr Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, Andrew J Ballard, Andrea Banino, Misha Denil, Ross Goroshin, Laurent Sifre, Koray Kavukcuoglu, et al. Learning to navigate in complex environments. *arXiv preprint arXiv:1611.03673*, 2016.
- [20] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015.
- [21] Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. Reverie: Remote embodied visual referring expression in real indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9982–9991, 2020.
- [22] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [23] Santhosh K Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alex Clegg, John Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. *arXiv preprint arXiv:2109.08238*, 2021.
- [24] 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.

- [25] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.
- [26] 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.
- [27] James A Sethian. A fast marching level set method for monotonically advancing fronts. *proceedings of the National Academy of Sciences*, 93(4):1591–1595, 1996.
- [28] Dhruv Shah, Michael Robert Equi, Błażej Osiński, Fei Xia, Brian Ichter, and Sergey Levine. Navigation with large language models: Semantic guesswork as a heuristic for planning. In *Conference on Robot Learning*, pages 2683–2699. PMLR, 2023.
- [29] Dhruv Shah, Błażej Osiński, Sergey Levine, et al. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *Conference on robot learning*, pages 492–504. PMLR, 2023.
- [30] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019.
- [31] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019.
- [32] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese.

- Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9068–9079, 2018.
- [33] 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.
- [34] Wei Yang, Xiaolong Wang, Ali Farhadi, Abhinav Gupta, and Roozbeh Mottaghi. Visual semantic navigation using scene priors. *arXiv preprint arXiv:1810.06543*, 2018.
- [35] Joel Ye, Dhruv Batra, Abhishek Das, and Erik Wijmans. Auxiliary tasks and exploration enable objectgoal navigation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16117–16126, 2021.
- [36] 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.
- [37] 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.
- [38] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In *2017 IEEE international conference on robotics and automation (ICRA)*,

pages 3357–3364. IEEE, 2017.