Exploring Generalized Robotics Navigation: Novel Environment Navigation and Robot Adaptability Assessment

50.047 Mobile Robotics Spring 2024: Final Report

Yeo Ke Wei
1005165
SUTD Design and Artificial Intelligence

***Abstract*—This report evaluates the hypothesis that the General Navigation Model (GNM) [1], trained on a multi-robot dataset, can generalize navigation capabilities to novel robots and environments. The investigation involved implementing the GNM repository [2] within a Gazebo simulation using a robot not included in the model's original training dataset, thus testing the model's adaptability and transferability.**

I. Introduction

A. Reference Paper

Recent advances in machine learning have facilitated broad generalization across various domains such as natural language processing and visual perception, primarily by leveraging large, Internet-scale datasets to learn general patterns applicable to numerous tasks. This approach, often involving techniques like zero-shot transfer or fine-tuning, has yet to be fully realized in robotics due to the unique challenges posed by the field's diversity in environments and platforms. Robotics typically suffers from fragmented data collection efforts, where each research setup involves robot-specific datasets and control policies, preventing the accumulation of universally applicable, large-scale datasets. The reference study addresses this limitation by proposing a framework for an embodiment-agnostic General Navigation Model (GNM). This model is trained on a diverse multi-robot dataset to achieve broad generalization in visual navigation, demonstrating effective policy transfer across various robot platforms with different sensor setups and physical configurations. The work underscores the potential of shared, large-scale datasets in creating versatile navigation models that can adapt to new robots and environments [1].

B. Motivation and Contribution

Building on the foundational work of the reference paper, this report explores further into the adaptability and transferability of the GNM. My contribution centers around deploying the GNM in a novel setting in Gazebo simulation along with a custom-designed 2-wheeled differential-drive mobile robot, which was not included in the original training dataset.

II. Related Work

The pursuit of generalizable machine learning models in robotics has seen diverse approaches, primarily centered on learning from large, heterogeneous datasets. Previous research has demonstrated the scalability of data sharing across robotic platforms to improve learning outcomes in complex environments. Such studies underscore the potential of multi-robot datasets in overcoming the limitations of traditional, robot-specific data collections, which often suffer from lack of diversity and scale, confining the applicability of learned policies to narrow operational contexts [3]–[5].

A significant aspect of this research area involves transfer learning, where the objective extends beyond mere generalization across similar robots to include diverse dynamics, environments, and physical embodiments [6]–[8]. Unlike domain-specific adaptation strategies that rely heavily on engineered solutions, our work and others aim to explore how simple, high-capacity models, when trained on real-world multi-robot data, can achieve broad applicability across varied navigation tasks [9].

Additionally, while prior work has also explored the utility of passive data sources like online videos for learning visual representations or policies [10]–[12], our approach leverages on-robot data to directly train policies. This method aligns with ongoing efforts in the field that integrate topological graphs and image-goal policies to enhance navigation systems' adaptability and long-range operational capabilities [13], [14].

This study builds upon these foundational efforts by implementing a unified model capable of navigating complex environments and achieving specific goals across multiple robotic platforms. The GNM's robustness against diverse challenges such as sensor variations and physical wear further underscores its utility in real-world applications, setting a precedent for future research in robotic navigation.

III. Proposed Approach

A. GNM Methodology

Our work centers on advancing the GNM to study common navigation tasks across diverse robotic platforms and environments. The GNM is a image-goal navigation, a framework that does not depend on semantic labels or precise localization, making it adaptable to a wide variety of visual navigation datasets. The core objective is to develop a goal-reaching policy that navigates using only egocentric visual observations, conditioned on the target goal image.

To accommodate the inherent variability in robot dynamics, sensor configurations, and control schemes, the reference paper propose a methodology that involves two critical elements:

1. **Action Representation Transformation**: The action space is standardised across different robots by using mid-level action representations, such as relative way-points and yaw changes. These actions are derived from local odometry available across platforms and are expressed in a normalized format that abstracts away specific robot dynamics, facilitating learning from diverse data sources. For example, actions are scaled by a factor corresponding to each robot's top speed, enabling the policy to generalize across robots with different movement capabilities.
2. **Embodiment Context Conditioning**: To tailor the navigation policy to specific robot capabilities, the policy was conditioned on an embodiment context,. This context is dynamically inferred from a sequence of past observations, providing a real-time summary of the robot's physical and operational characteristics without requiring manual definition of parameters.

The GNM is trained with 60 hours of real-world navigation trajectories collected from six distinct robotic platforms. The architecture of the GNM modifies the typical goal-conditioned policy framework by incorporating two separate MobileNetv2 encoders. The first encoder processes the current observational data to generate context-conditioned representations, essential for adapting the policy to immediate environmental inputs. The second encoder focuses on the conditional goal observation, enabling the model to maintain a continuous understanding of the objective throughout the navigation process. This dual-encoder setup enhances the GNM's ability to integrate and react to dynamic changes in both the robot's immediate surroundings and its end goal.

B. Deployment

The goal-reaching policy is integrated with a topological map, where the nodes encapsulate the robot's sensor observations, enriched with the embodiment context derived from recent robot states. The edges between these nodes are defined by the temporal distances, estimated by the policy, reflecting the proximity in terms of time rather than physical space. The deployment process operates as follows [2]:

1. **Node Association**: At each timestep, the robot evaluates its current position relative to its goal by identifying the node in the topological map M that minimizes the temporal distance to the goal. This approach enables the robot to understand its progress towards the goal dynamically.
2. **Path Planning**: Utilizing Dijkstra's algorithm, the system calculates the most time-efficient path to the goal. This is achieved by determining an optimal sequence of sub-goals, which guide the robot through the map.
3. **Navigation Execution**: For each identified sub-goal in the sequence, the policy is invoked with the current observation and the immediate sub-goal. This results in a sequence of way-points that are specific to the sub-goal and contextually relevant to the robot's current state.
4. **Low-Level Control**: The generated way-points are then executed by the robot's specific low-level controller, which translates these way-points into actionable commands tailored to the robot's operational dynamics and capabilities.

C. Contributions

This research aims to test and validate the hypothesis that the reference paper's GNM can achieve effective generalization across novel environments and robotic platforms. To this end, the pre-trained Nomad model [15] from the reference study was integrated and adapted within a custom Gazebo simulation environment. This environment featured both small and large configurations to challenge the navigation capabilities of an open-sourced custom-built 2-wheeled differential-drive robot [16].

The performance of the Nomad model in these diverse settings was critically assessed through a qualitative analysis, employing the performance evaluation metric from the reference paper. The metric used focuses on success rates which are quantified by the progress made towards the goal. To achieve this, a custom Python script was tailored to accurately capture the model's navigational efficacy in reaching defined targets within the simulation through the capture and analysis the robot odometry data.

Further contributing to the field, we extended the model's utility by adapting the deployment framework from ROS1 to ROS2. This adaptation not only improves the model's compatibility with current robotic software standards but also broadens its practical utility, setting a foundation for future explorations into versatile robotic navigation systems.

D. Process Flow

The experimental setup is designed for reproducibility and simplicity through the use of custom script files at each step of the testing process.

Manual Operation: The experimentation begins with the manual operation of the robot within the Gazebo simulation environment. Using generic keyboard tele-operation, we navigate the robot across different predefined environments. During this phase, critical data such as odometry and camera inputs are captured in a ROS bag. This initial step is crucial as it establishes a baseline trajectory for the robot, which is in "path-following" mode, aimed at replicating a specific path to reach the designated goal. It is to note that the odometry data collected is not used for the navigation model but for performance analysis in subsequent step.

Creation of Topological Map: Subsequently, the captured ROS bag is replayed, and snapshots from the camera feed are taken at fixed intervals of 1.5 seconds. Each snapshot represents a node in the topological map, creating a sequential arrangement from the starting point (node 0) to the goal (node N). This map serves as a foundational element in our navigation system, providing spatial and visual context to the navigation model.

Deployment:With the topological map established, the pre-trained Nomad navigation model is then deployed. The model interacts with the Gazebo simulation by issuing navigation commands to the /cmd_vel topic, guiding the robot through the environment based on the established map. The robot's movement and performance data are again captured in a ROS bag for comprehensive post-experiment analysis.

Calculate Performance: We analyze the odometry data from the manual operation (referred to as Goal_odom) to extract the total distance traveled and the endpoint. The performance of each navigation attempt is then quantified using a custom Python script which calculates the Euclidean distance between the goal's endpoint and the endpoint of each attempt. Additionally, we compute the mean distance across all attempts and the completion percentage relative to the total distance traveled.

E. Environments and Robot Platform

1) Robot Platform: Our experimental setup features a novel, custom robot platform, specifically a 2-wheeled differential robot known as "Dumpster". This open-source platform [16] includes a rigid chassis with two independently driven wheels positioned on a common horizontal axis, complemented by two passive support caster wheels, one at the front and one at the back. For sensory input, it incorporates an RGB camera, pivotal for visual navigation tasks.

2) Environment: To evaluate the robustness and adaptability of the navigation model, we selected three distinct simulated environments, each offering varying levels of complexity and navigational challenges (Table I).

IV. EXPERIMENTAL RESULTS

Based on the experimental results II, it is evident that the navigation model demonstrates satisfactory performance in navigating through novel environments that were not part of the training dataset. Additionally, the model exhibits competent control over a novel robot platform. The baseline test conducted in the cafe environment showcases promising outcomes, with the highest mean completion rate observed, aligning with the expected difficulty levels of the environments.

A. Observations

The navigation model demonstrates a characteristic left-right scanning pattern, colloquially referred to as 'wiggling,' while advancing forward. Figure 1 illustrates that all five trajectories successfully reached the goal position, however it is worth noting that the experiments were prolonged until the robot either reached the goal node or encountered significant deviation from the intended path.

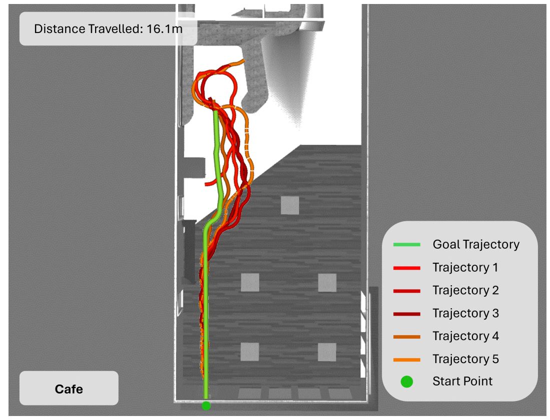

Fig. 1. Robot Trajectories in Cafe Environment

Fig. 2. Robot Trajectories in City Environment

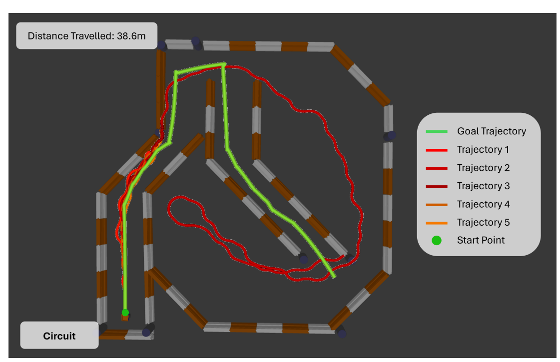

Fig. 3. Robot Trajectories in Circuit Environment

B. Challenges

During the experimental trials, several challenges were encountered, including:

Obstacle Collision Recovery: The navigation model struggled to recover from instances where the robot collided with obstacles, leading to prolonged periods of immobilization. This occurs most prominently in the Circuit Environment (Figure 3)

Drastic Turns: High failure rates were observed when the robot executed turns with excessive speed or sharp angles. Such rapid maneuvers often caused the snapshot algorithm to fail in capturing in-between sequences. This prevented the model from connecting the trajectory context, resulting in navigation errors.

V. Discussion and Conclusion

A. Findings

This study provide valuable insights into the applicability of the referenced paper's hypothesis regarding the generalization capabilities of navigation models across diverse environments and robot platforms. Our experimental results demonstrate that the pre-trained navigation model, originally proposed in the reference paper, exhibits promising performance in navigating novel environments and controlling an unfamiliar robot platform. Specifically, the model showcased commendable adaptability and competence in traversing previously unseen terrains, with notable success rates recorded across various difficulty levels of simulated environments.

B. Contributions

The contributions complement the referenced paper's hypothesis by providing empirical evidence of the practical viability of deploying pre-trained navigation models in real-world scenarios involving novel environments and robot platforms. By integrating the navigation model into a custom Gazebo simulation environment and a unique 2-wheeled differential-drive robot platform, I extended the scope of the study and validated the model's effectiveness across diverse settings.

Acknowledgment

The author expresses sincere gratitude to Dr. Malika Meghjani, Assistant Professor at the Singapore University of Technology and Design (SUTD), Information Systems Technology and Design (ISTD) Pillar, for her invaluable guidance and mentorship throughout the Mobile Robotics course in Spring 2024. This project has greatly benefited from her extensive knowledge and insightful feedback.

The author also wishes to thank the teaching assistants, Virbhadrappa Swaminath Kalburgi and Marcel Bartholomeus Prasetyo, for their unwavering support and assistance. Their dedication and timely help were crucial to the successful completion of this project and greatly enhanced the learning experience.

References

[1] D. Shah, A. Sridhar, A. Bhorkar, N. Hirose, and S. Levine, "GNM: A General Navigation Model to Drive Any Robot," 2023. [Online]. Available: https://arxiv.org/abs/2210.03370

[2] D. Shah, "General navigation models: Gnm, vint and nomad," https://github.com/robodhruv/visualnav-transformer/tree/main, 2023.

[3] T. D. P. A. C. Devin, A. Gupta and S. Levine, "Learning modular neural network policies for multi-task and multi-robot transfer," *International Conference on Robotics and Automation (ICRA)*, pp. 2169–2176, 2017.

[4] F. E. e. a. S. Dasari, ""robonet: Large-scale multirobot learning," *Conference on Robot Learning (CoRL)*, 2020.

[5] X. W. W. X. Y. C. F. L. V. M. F. Yu, H. Chen and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[6] C. K. L. W. Yu, J. Tan and G. Turk, "Preparing for the unknown: Learning a universal policy with online system identification," *Robotics: Science and Systems*, 2017.

[7] W. Z. X. B. Peng, M. Andrychowicz and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," *International Conference on Robotics and Automation (ICRA)*, 2018.

[8] Y. L. P. A. A. Gupta, C. Devin and S. Levine, "Learning invariant feature spaces to transfer skills with reinforcement learning," *International Conference on Learning Representations (ICLR)*, 2017.

[9] C. R. d. B. A. Loquercio, A. I. Maqueda and D. Scaramuzza, "Dronet: Learning to fly by driving," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1088–1095, 2018.

[10] A. G. M. Chang and S. Gupta, "Semantic visual navigation by watching youtube videos," in *Advances in Neural Information Processing Systems*, vol. 33. Curran Associates, Inc., 2020, pp. 4283–4294.

[11] S. T. M. M. J. M. R. M. Hahn, D. S. Chaplot and A. Gupta, "No rl, no simulation: Learning to navigate without navigating," in *Advances in Neural Information Processing Systems*, vol. 34. Curran Associates, Inc., 2021, pp. 26 661–26 673.

[12] V. K. C. F. S. Nair, A. Rajeswaran and A. Gupta, "R3m: A universal visual representation for robot manipulation," 2022.

[13] G. K. N. R. D. Shah, B. Eysenbach and S. Levine, "Ving: Learning open-world navigation with visual goals," in *International Conference on Robotics and Automation (ICRA)*, 2021.

[14] N. Savinov, A. Dosovitskiy, and V. Koltun, "Semi-parametric topological memory for navigation," 2018.

[15] A. Sridhar, D. Shah, C. Glossop, and S. Levine, "NoMaD: Goal Masked Diffusion Policies for Navigation and Exploration," *arXiv pre-print*, 2023. [Online]. Available: https://arxiv.org/abs/2310.xxxx

[16] R. Z. R., "rtab$_{d}umpster,'' https : //github.com/rfzeg/rtab_{d}umpster$, 2019.

Appendix

TABLE I
Description of Testing Environments

Environment	Difficulty	Characteristics	Purpose
Cafe	Low	Mostly open spaced with few obstacles, short distance navigation, low amount of turning (<45 degrees)	Establish baseline performance metrics
City	Medium	Open spaced with frequent obstacles, long distance navigation, medium amount of turning (up to 90 degrees)	Test model's performance in urban-like settings
Circuit	High	Tightly spaced with numerous obstacles, long distance navigation, high amount of turning (exceeding 90 degrees)	Assess model's capability in highly constrained scenarios

TABLE II
NAVIGATION RESULTS

Environment	Total Distance (m)	Mean Distance from Goal (m)	Percentage Completed (using mean) (%)
Cafe	16.1	2.2	86.1
City	46.96	7.8	83.4
Circuit	38.66	12.4	67.9

TABLE III
NAVIGATION RESULTS (CONTINUED)

Environment	Attempt 1 (m)	Attempt 2 (m)	Attempt 3 (m)	Attempt 4 (m)	Attempt 5 (m)
Cafe	4.0 (75.4%)	2.7 (83.2%)	0.8 (95.3%)	0.6 (96.4%)	3.2 (80.3%)
City	9.7 (79.4%)	22.9 (51.3%)	0.1 (99.8%)	5.9 (87.5%)	0.5 (99.0%)
Circuit	15.1 (60.9%)	1.4 (96.3%)	15.1 (61.0%)	15.1 (61.0%)	15.3 (60.3%)