Reduce, Reuse, Recycle: Modular Multi-Object Navigation

Sonia Raychaudhuri[1], Tommaso Campari[2,3], Unnat Jain[4], Manolis Savva[1], Angel X. Chang[1,5,6]
[1]Simon Fraser University, [2]University of Padova, [3]FBK, [4]Meta AI, [5]Canada-CIFAR AI Chair, [6]Amii

Abstract

Our work focuses on the Multi-Object Navigation (MultiON) task, where an agent needs to navigate to multiple objects in a given sequence. We systematically investigate the inherent modularity of this task and develop a simple but effective modular approach with four modules: (a) an object detection module trained to identify objects from RGB images, (b) a map building module to build a semantic map of the observed objects, (c) an exploration module enabling the agent to explore its surroundings, and finally (d) a navigation module to move to identified target objects. We show that we can effectively reuse a PointGoal navigation model in the MultiON task instead of learning to navigate from scratch. Our experiments show that a PointGoal agent-based navigation module outperforms analytical path planning on the MultiON task. We also compare exploration strategies and surprisingly find that a uniform top-down sampling strategy significantly outperforms more advanced exploration methods. We additionally create MultiON 2.0, a new large-scale dataset as a test-bed for our approach.

1. Introduction

Embodied AI research has seen tremendous progress across various tasks with the availability of fast and high-fidelity simulators [24, 27, 11], deep reinforcement learning advances [25, 15], improved memory representation [14, 5, 28] and self-supervision schemes [9, 13, 26, 21], and parallel training infrastructure [29, 12, 18, 3]. Near-perfect performance on basic navigation tasks such as PointGoal where the agent navigates to a relative goal position has been achieved [29]. However, navigation tasks [4, 2, 17, 20, 8] where the agent needs to find objects or areas in the environment are far from solved. Such tasks require the agent to possess capabilities such as visual understanding, mapping and exploration in addition to basic navigation (see Fig. 1).

In this work, we study how we can leverage agents trained on the simpler PointGoal task in the context of more complex long-horizon navigation tasks. We propose a modular approach called *Modular-MON*, where each module is responsible for a specific task. In summary: i) we show that

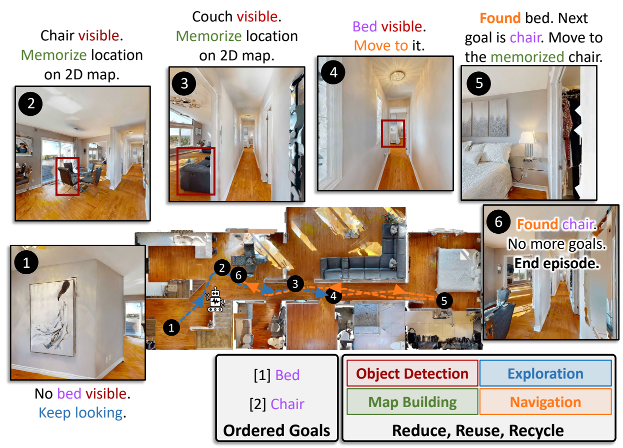

Figure 1: **Approach Overview.** We tackle long-horizon navigation tasks by proposing a modular approach, Modular-MON, to leverage their inherent modularity. Our approach consists of four modules: (1) *Object detection* ($\mathcal{O}$), (2) *Map building* ($\mathcal{M}$), (3) *Exploration* ($\mathcal{E}$) and (4) *Navigation* ($\mathcal{N}$).

our modular approach can effectively leverage pre-trained models and heuristics-based approaches to solve complex navigation task; ii) we show that a pretrained PointNav agent outperforms analytical path planners by a significant margin; iii) we compare rule-based exploration strategies and find that a simple strategy based on uniform top-down sampling outperforms more complex methods; and iv) we create MultiON 2.0, a new large-scale dataset as a test-bed for our approach. iv) we create MultiON 2.0, a new large-scale dataset for multi-object navigation.

2. Approach

In Modular-MON, we take a modular approach to multi-object navigation by employing the following modules: (1) *Object detection* ($\mathcal{O}$), (2) *Map building* ($\mathcal{M}$), (3) *Exploration* ($\mathcal{E}$) and (4) *Navigation* ($\mathcal{N}$). These modules are intuitively weaved together. The first two contribute to acquiring and storing semantic knowledge about the environment, while the latter two enable efficient embodied navigation. Modular-MON identifies objects ($\mathcal{O}$) by observing the environment and builds a semantic map ($\mathcal{M}$) by projecting the category labels of the objects (*i.e*. semantics) in the field of view. If the agent has not yet discovered the current goal, it will continue to explore ($\mathcal{E}$). Once the current goal has been discovered, Modular-MON plans a path from its current location to the goal, and generates actions to navigate ($\mathcal{N}$) towards the goal. We experiment with different exploration and navigation strategies to systematically investigate their contribution to the agent performance. For the Exploration ($\mathcal{E}$) module, we compare a simple Uniform Top-down Sampling with the more complex Stubborn [19] and Frontier [31], whereas for the Navigation ($\mathcal{N}$) modules, we compare a pre-trained PointNav module with heuristics-based analytical path planners such as Shortest Path Follower [24], BFS [10] and FMM [6]. For our Object detection ($\mathcal{O}$) module, we use two separate FasterRCNN [23] (finetuned offline) to identify cylinders and natural objects in the MultiON task. On the other hand, we use a pre-trained RedNet[16] from [5] for the ObjectNav experiments. As our Map building ($\mathcal{M}$) module, we project semantic labels of the objects onto a 2D grid map of the environment using depth observations following [7].

3. Experiments and Results

Task. In the MultiON task, the agent needs to navigate to a sequence of objects in a given order. Once the agent has reached each object and generated the *Found* action successfully, it is given the next goal. This continues until the agent has found all the goals in the episode. We also evaluate Modular-MON on the ObjectNav task from Batra et al. [4] which is an single-hop visual navigation task. ObjectNav allows the agent a maximum of 500 steps compared to 2500 in MultiON. the widely adopted Habitat platform [24] for our experiments.

Dataset. For our experiments, we prepared MultiON 2.0 – a large-scale dataset for the Multi-Object Navigation task. Compared to the original MultiON dataset [28], MultiON 2.0 is built on top of the large-scale HM3D [22] dataset containing 10x more scenes, uses an additional set of *Natural objects*[1], includes distractor objects, and has longer episodes. For the ObjectNav experiments, we use the ObjectNav dataset from Habitat challenge 2022[30] which is built using HM3D scenes and contains six object categories.

Metrics. In addition to the standard visual navigation metrics such as *success* and *SPL* [1] we use the metrics introduced by Wani et al. [28], the *progress* and *PPL*. We use a neural PointNav policy trained using the established distributed PPO [29] framework for efficient parallelization on HM3D scenes.

Baselines. We compare *OracleSem* agent, which builds a semantic map using egocentric depth observations to project the semantic labels directly from the Habitat simulator, with the *PredictedSem*, which builds the semantic map with predicted semantic labels using a pre-trained object detector.

	Dataset	Object detection	Val			
			Success	Progress	SPL	PPL
PredictedSem	MultiON 2.0 (CYL)	FRCNN [23]	50	65	21	26
	MultiON 2.0 (NAT)	FRCNN [23]	28	47	11	18
	ObjNav-2022[30]	RedNet[16, 5]	30	-	28	-
OracleSem	MultiON 2.0 (CYL)	GT	80	87	35	38
	MultiON 2.0 (NAT)	GT	80	85	35	38
	ObjNav-2022[30]	GT	64	-	30	-

Table 1: **Modular-MON performance.** OracleSem, using oracle semantic labels, outperforms PredictedSem in general. PredictedSem performs better on Cylinder objects than Natural objects in MultiON, and considerably well on the ObjectNav task. These experiments use Uniform Top-down Sampling as Exploration ($\mathcal{E}$), PointNav [22] ('PN') as Navigation ($\mathcal{N}$) and [7] as Map building ($\mathcal{M}$).

Results. In Tab. 1, first we observe that OracleSem agents outperform PredictedSem agents in general, which is intuitive since the former uses oracle semantic labels from the simulator. Performance drop in PredictedSem agents is due to the limitation of the object detector modules. Note that all these methods use Uniform Top-down Sampling w/ Fail-Safe ('Uf') as the Exploration module and PointNav [22] ('PN') as the Navigation module. Second, we observe that PredictedSem performs better on cylinder objects than the natural objects in MultiON 2.0 dataset, which can be intuitively explained by the fact that cylinder objects are easier to detect than the more diverse natural objects with varying shapes, colors and sizes. Our third observation is that we can effectively evaluate our Modular-MON on other navigation tasks, such as ObjectNav, by only swapping the object detector module and still achieve significant performance.

Furthermore, we find through various experiments that using a pre-trained PointNav is more effective than the analytical path planners as the Navigation module and using a simple Uniform Top-down Sampling outperforms the complex strategies, such as Stubborn and Frontier.

4. Conclusion and Future Work

We carried out a systematic analysis of the different modules of our Modular-MON to show that using a pre-trained PointGoal navigation agent is very effective in addressing the more complex MultiON task as well as ObjectNav task. We believe our findings will encourage the community to use modular approach towards solving complex tasks and thus leverage available pre-trained models for different modules, instead of training new end-to-end models from scratch.

[1]3D models from https://sketchfab.com/3d-models distributed under permissive licenses.

Acknowledgements. The members at SFU were supported by Canada CIFAR AI Chair grant, Canada Research Chair grant, NSERC Discovery Grant and a research grant by Facebook AI Research. Experiments at SFU were enabled by support from WestGrid and Compute Canada. TC was supported by the PNRR project Future AI Research (FAIR - PE00000013), under the NRRP MUR program funded by the NextGenerationEU.

References

[1] Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, et al. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*, 2018. 2

[2] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *CVPR*, pages 3674–3683, 2018. 1

[3] Mohammad Babaeizadeh, Iuri Frosio, Stephen Tyree, Jason Clemons, and Jan Kautz. Reinforcement learning through asynchronous advantage actor-critic on a GPU. In *ICLR*, 2017. 1

[4] Dhruv Batra, Aaron Gokaslan, Aniruddha Kembhavi, Oleksandr Maksymets, Roozbeh Mottaghi, Manolis Savva, Alexander Toshev, and Erik Wijmans. ObjectNav revisited: On evaluation of embodied agents navigating to objects. *arXiv preprint arXiv:2006.13171*, 2020. 1, 2

[5] Vincent Cartillier, Zhile Ren, Neha Jain, Stefan Lee, Irfan Essa, and Dhruv Batra. Semantic MapNet: Building allocentric semanticmaps and representations from egocentric views. In *AAAI*, 2021. 1, 2

[6] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural SLAM. In *ICLR*, 2019. 2

[7] Devendra Singh Chaplot, Dhiraj Prakashchand Gandhi, Abhinav Gupta, and Russ R Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. In *NeurIPS*, volume 33, pages 4247–4258, 2020. 2

[8] Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *CVPR*, pages 12538–12547, 2019. 1

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 1

[10] Matt Deitke, Dhruv Batra, Yonatan Bisk, Tommaso Campari, Angel X Chang, Devendra Singh Chaplot, Changan Chen, Claudia Pérez D'Arpino, Kiana Ehsani, Ali Farhadi, et al. Retrospectives on the embodied AI workshop. *arXiv preprint arXiv:2210.06849*, 2022. 2

[11] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Jordi Salvador, Kiana Ehsani, Winson Han, Eric Kolve, Ali Farhadi, Aniruddha Kembhavi, et al. ProcThor: Large-scale embodied AI using procedural generation. *NeurIPS*, 2022. 1

[12] Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *ICML*, 2018. 1

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 1

[14] Joao F Henriques and Andrea Vedaldi. MapNet: An allocentric spatial memory for mapping environments. In *CVPR*, pages 8476–8484, 2018. 1

[15] Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *AAAI*, 2018. 1

[16] Jindong Jiang, Lunan Zheng, Fei Luo, and Zhijun Zhang. Rednet: Residual encoder-decoder network for indoor rgb-d semantic segmentation. *arXiv preprint arXiv:1806.01054*, 2018. 2

[17] Jacob Krantz, Erik Wijmans, Arjun Majundar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision and language navigation in continuous environments. In *ECCV*, 2020. 1

[18] Iou-Jen Liu, Raymond Yeh, and Alexander Schwing. High-throughput synchronous deep RL. In *NeurIPS*, 2020. 1

[19] Haokuan Luo, Albert Yue, Zhang-Wei Hong, and Pulkit Agrawal. Stubborn: A strong baseline for indoor object navigation. *arXiv preprint arXiv:2203.07359*, 2022. 2

[20] Dipendra Misra, Andrew Bennett, Valts Blukis, Eyvind Niklasson, Max Shatkhin, and Yoav Artzi. Mapping instructions to actions in 3D environments with visual goal prediction. In *EMNLP*, 2018. 1

[21] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *ICML*, pages 2778–2787, 2017. 1

[22] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-matterport 3D dataset (HM3d): 1000 large-scale 3D environments for embodied AI. In *NeurIPS Datasets and Benchmarks Track (Round 2)*, 2021. 2

[23] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *NeurIPS*, 28, 2015. 2

[24] 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 *ICCV*, pages 9339–9347, 2019. 1, 2

[25] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 1

[26] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, and Sergey Levine. Time-contrastive networks: Self-supervised learning from video. In *ICRA*, 2018. 1

[27] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam,

Devendra Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. *NeurIPS*, 2021. 1

[28] Saim Wani, Shivansh Patel, Unnat Jain, Angel Chang, and Manolis Savva. Multion: Benchmarking semantic map memory using multi-object navigation. *NeurIPS*, 33:9700–9712, 2020. 1, 2

[29] Erik Wijmans, Abhishek Kadian, Ari Morcos, Stefan Lee, Irfan Essa, Devi Parikh, Manolis Savva, and Dhruv Batra. DD-PPO: Learning near-perfect pointgoal navigators from 2.5 billion frames. In *ICLR*, 2019. 1, 2

[30] Karmesh Yadav, Santhosh Kumar Ramakrishnan, John Turner, Aaron Gokaslan, Oleksandr Maksymets, Rishabh Jain, Ram Ramrakhya, Angel X Chang, Alexander Clegg, Manolis Savva, Eric Undersander, Devendra Singh Chaplot, and Dhruv Batra. Habitat challenge 2022. https://aihabitat.org/challenge/2022/, 2022. 2

[31] Brian Yamauchi. A frontier-based approach for autonomous exploration. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA). 'Towards New Computational Principles for Robotics and Automation'*, pages 146–151, 1997. 2