New Problems in Active Sampling for Mobile Robotic Online Learning

Anonymous

***Abstract*—AI models deployed in real-world tasks (e.g., surveillance, implicit mapping, health care) typically need to be online trained for better modelling of the changing real-world environments and various online training methods (e.g., domain adaptation, few shot learning) are proposed for refining the AI models based on training input incrementally sampled from the real world. However, in the whole loop of AI model online training, there is a section rarely discussed: how to sample training input from the real world. In this paper, we show from the perspective of online training of AI models deployed on edge devices (e.g., robots) that several problems in sampling of training input on the device are affecting the time and energy consumption for the online training process to reach high performance. Notably, the online training relies on training input consecutively sampled from the real world and the consecutive samples from nearby states (e.g., position and orientation of a camera) are too similar and would limit the training accuracy gain per training iteration; on the other hand, while we can choose to sample more about the inaccurate samples to better final training accuracy, it is costly to obtain the accuracy statistics of samples via traditional ways such as validating, especially for AI models deployed on edge devices. These findings aim to raise research effort for practical online training of AI models, so that they can achieve resiliently and sustainably high performance in real-world tasks.**

I. Introduction

Online learning [1]–[4] refers to real-time training a pretrained Artificial Intelligence (AI) model on training inputs consecutively sampled from the real world for various edge AI applications (e.g. transportation, human language processing, implicit SLAM) deployed on edge devices (e.g., robots), so that the model can adapt to changing real-world environments and retain high performance. Various online learning methods have been proposed (e.g., domain adaptation [1], self training [4]) in pursuit of high training accuracy on given samples. While their samples are traditionally collected by human labors or along a given routine that lacks interaction with training, leveraging the mobility of robots to automate the sampling process in reaction to real-time training statistics (i.e., active sampling) has the potential to further boost training accuracy and power consumption reduction, beside saving human labor.

More specifically, sample for online learning are traditionally collected manually or along a preset routine hoping to cover as many as possible scenarios, where sample quality is not guaranteed [5]–[7]. Recent studies show that training input quality plays a vital role in training accuracy of AI models [5]–[7], and in robotic online learning, sample quality further influences critical requirements of high *training efficiency* (i.e., training accuracy gain per training iteration) and high *training throughput* (i.e., number of training iterations in unit time) so that the AI applications deployed on the robot can as fast as possible retain high performance in the real world with as few as possible training iterations and power consumption. Active sampling discussed in this paper aims to increase the sample quality of online learning by assessing the quality of possible samples, planning the path to collect the samples and finally executing the path plan on the robot, in pursuit of high training efficiency and throughput.

Active sampling was first proposed in the problem of active simultaneous localization and mapping (SLAM) [8]: a robot automatically decides its sampling destination in real-time reaction to estimated localization quality and mapping quality: either sample areas of high mapping accuracy to optimize localization quality or samples areas of low mapping accuracy to optimize mapping quality. Instead of aimlessly circulating, active sampling in active SLAM boosts both mapping speed and accuracy. Although active SLAM methods shed light on the design of active sampling for mobile robotic online learning, in this paper, we show that several problems are hindering ist training throughput and training efficiency.

First, training efficiency and training throughput seem empirically contradictory in active sampling for mobile robotic online learning, because the sampling states of a robot is consecutive, and we named it as the robot locality problem. While the robot is moving around local state space (e.g., nearby position and orientation), the consecutive samples and training optimize training throughput, but the consecutive samples are often too similar, limiting training efficiency between consecutive samples [9], [10], depicted in the lowland of training accuracy gain around the starting state of the robot in an implicit SLAM task (building dense 3D map via online learning) in Fig. 1. In our evaluation, training on 38.55% of the key samples selected from the consecutive samples achieved the same level of training accuracy as the consecutive samples. On the other hand, only sampling key samples improves training efficiency but harms training throughput, since the robot has to pause training and move around local states between the key samples. It leaded to almost no mapping error advantage after training for the same wall-clock time compared with consecutive sampling in our evaluation.

Second, real-time estimation of potential training accuracy gain of the training AI model for real-time active sampling decision making is difficult. While the estimation of accurate potential training accuracy gain in AI training is yet an open problem, an approximate way to quantify potential training accuracy gain is to validate training samples and those with

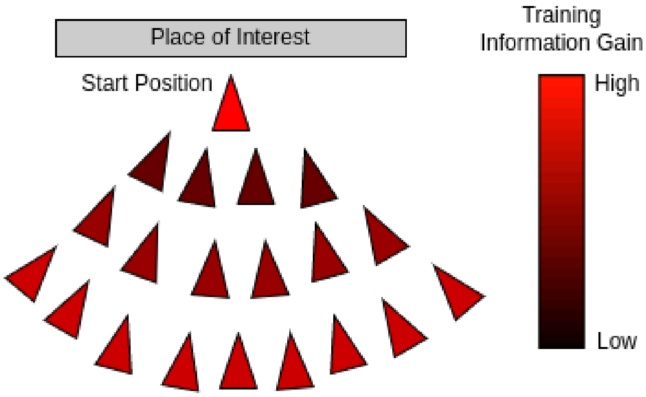


Fig. 1: The lowland of training accuracy gain around the starting state. The triangle represents the position and orientation that the sample is taken.

low training accuracy have high potential training accuracy gain [9], [10]. But it is too computation-intensive for edge devices and easily breaks the control loop. We evaluated that the validation of the training accuracy in an implicit SLAM task typically takes 30 to 200 seconds and the robot has to wait for such a long time before each decision making.

The key reason of these problems is that AI training process is probabilistic and implicit, different from the traditional deterministic and explicit SLAM process. The lowland of training accuracy gain around the starting sampling pose implies AI training as a probabilistic data driven method requires key sampling from multiple angles and distances to attain full potential training accuracy gain, which suffers severe low training efficiency when moving in and out the low land of training accuracy gain. Worse, the time-consuming validation of training model for active sampling decision making further slows down the whole online learning process and affects training throughput. These problems together caused the implicit SLAM process in our evaluation be slowed down by 87.68% to reach a same high accuracy and increased mapping error by 2.73X after training for the same time, compared with an idealized case with consecutive sampling on selected key frames and no validation time cost.

In this paper, we take the first step to reveal and evaluate in both quality and quantity the problems hindering the active sampling for online learning from achieving both high training efficiency and high training throughput. As we are borrowing the idea of active sampling from active SLAM, we choose implicit SLAM as the main evaluation item for simplicity since implicit SLAM shares a similar task with active SLAM. These findings aim to raise research effort for practical active sampling for mobile robotic online learning of AI models, so that the AI models deployed in real-world tasks can achieve resiliently and sustainably high performance against the changing real-world environments with minimal time and energy consumption of online learning.

The rest of the paper is organized as follows: the second chapter provides background; the third chapter describes in detail about the problems and the estimation methodology; the final chapter concludes.

II. Background

Online Training. Artificial Intelligence (AI) models are typically trained for a specific task on a dedicated training dataset. However, in many real-world applications,labeling datasets are very expensive, and the data distributions in the real-world can differ or even change over time. Therefore, it is a common requirement to refine the pre-trained model on real-world unlabeled data collected in the deployed environments to retain high performance [1]–[4]. Many semi-supervised or unsupervised methods [11], [12] are proposed to learn knowledge from such unlabeled data collected from the real world and adapt the AI model to the changing environment. Among them, unsupervised domain adaptation methods [3], [13] often align the distribution shifts between pre-training datasets and real-world data by retraining the model using discrepancy losses or adversarial training. Self-training methods [4], [14] train a student model that adapts to changing real-world environments by generating pseudo labels for the unlabeled data from the pre-trained model. With the rapid development of methods, the AI models deployed in the real world can adapt to unstructured and even ever changing real-world environments

Data centric AI and active learning. Data centric AI [5] refers to optimizing the training data quality and quantity in AI training, besides traditional ways of optimizing AI model structures. Recently researchers' attention towards data centric AI is magnified with the recent successes of GPT models that share a similar model structure but continuously improve in training data size and data quality.

The idea of data centric AI can be traced back to active learning [9]: active learning is a classic AI problem to select high quality training data out of the dataset for training under the idea that a machine learning algorithm can achieve greater accuracy with fewer training labels if it is allowed to choose the data from which it learns. A typical way to assess training data quality in active learning is informativeness measure [9], [10] that tests whether the training data lies in the instance space that is still ambiguous to the training AI model. The informativeness measure pipeline is specific for different AI tasks, but mostly relies on validation, e.g., in a binary classification task, validate the samples and the samples whose posterior probability of being positive is nearest 0.5 are the uncertain samples that desire further sampling. This theory comprises with the lowland of training accuracy gain around a starting sampling state we observed since the samples from local states around the starting state would be less ambiguous after training and thus be of lower uncertainty and lower quality for online training.

Active SLAM Methods. Active SLAM methods [8], [15], [16] is targeted at a different task from AI training: simultaneously localizing a robot and build its surrounding map, but active SLAM and active learning share a similar idea towards inputs: valuing the quality of inputs. While traditional SLAM methods passively accept samples collected along a given routine, active

SLAM methods actively estimate the quality of samples and plan the robot motions to collect the samples of high quality, optimizing both SLAM speed and accuracy [8]. The typical idea to estimate sample quality in active SLAM is estimating the changes of building map after the current sample is input and calculated, where areas that change evidently imply higher level of uncertainty and more information gain [15], [16]. The combination of active SLAM and active learning shed light on the design of active sampling for mobile robotic online learning, with active learning methods guiding the estimation of sample quality and active SLAM methods guiding the motion planning of robot. This can be considered as a data-centric cyber physical system for optimizing the time and energy consumption of online training of AI models deployed on robots for real-world tasks and we are sharing about the problems we discovered during our implementation of a such system.

Implicit SLAM. Implicit SLAM [17], [18] is a neural-network-based method to simultaneously localizing a robot and building a dense 3D map of its surrounding environments (dense SLAM). Note that this dense SLAM task is much more complex than traditional SLAM tasks since it includes 3D mapping together with RGB information, while traditional SLAM only considers a 2D map. Traditional algorithms of SLAM tasks suffer high computation complexity in dense SLAM, accounting for the introduction of AI training in this task. A state-of-the-art implicit SLAM method, NICE SLAM [17], can be considered as a self-training method. It takes robot RGBD camera images together with the robot poses where the images are taken as training input and trains a neural network which is an implicit representation of the dense 3D map. While the RGBD images are the ground truth, the input robot poses are pseudo labels generated by optimizing the previous robot poses against the RGBD image and training model. NICE SLAM is able to train the neural network with RGBD image stream from the robot camera and incrementally build the neural network of an implicit dense 3D map in real time. But same as informativeness measure in active learning, it needs to validate the model against sampled images to produce an explicit dense 3D map for traditional active SLAM to estimate sample quality.

III. Problems in Active Sampling for Online Training

This section explores how the two main problems in active sampling for mobile robotic online training, robot locality problem and high time consumption of estimation of potential training accuracy gain, affect online training in both quality and quantity.

Testbed. We use a four-wheel robot equipped with an NVIDIA Jetson Xavier NX [19] as the main testbed and set the robot's maximum moving speed to 20cm/s suitable for mapping tasks. We select NICE SLAM [17] as our main evaluation item and integrates it as a ROS package deployed over the robot. We connect the robot with RGBD input from a popular dense SLAM dataset, Habitat [20], [21], and port the real-time RGBD input to NICE SLAM backend to online train an implicit model of a dense map of the environments. The used implicit model has 7 million parameters. At best effort, roughly every 10 seconds the NICE SLAM backend takes an RGBD input as a key frame and trains the implicit model against the key frame together with previous sampled key frames for 30 iterations. We disable tracking in NICE SLA and use the ground-truth positions and orientations in online training, because we want to explore how the aforementioned problems affect mapping error and avoid the influence of tracking error.

Consecutive Sampling. If a sampling plan (plan of the following movements) is made, the robot executes the sampling plan by moving around its local state before the next sampling plan is made, which results in consecutive sampling that suffers low training efficiency (shown in Fig. 2a). To make sampling plans, every 10 key frames being sampled, we validate the implicit model of NICE SLAM to get the explicit 3D dense map, and then input the dense map to iRotate [22], a traditional active SLAM algorithm, to decide the following sampling positions and orientations that optimize training accuracy gain.

Sparse Sampling. To reveal how consecutive sampling affects sample quality, we randomly select sampling positions and orientations from those computed in the same computation process of consecutive sampling, but enforcing a lower bound of distances from the previous sampling positions and orientations (shown in Fig. 2b). A real robot has to move along the path that covers these selected poses and pause sampling and training between these poses (i.e., sparse sampling), which improves training efficiency but harms training throughput. Note sparse sampling is meant to contrast consecutive sampling and not meant to be optimal. We also create a idealized case where a robot that can teleport in the environment (move beyond its local state) to avoid the local lowland of training accuracy gain and directly use the selected poses in sparse sampling as target sampling poses, thus these is not need to pause sampling and training and not need to pause training for model validation. We refer to the idealized case as teleported sampling. This is achieved by spawning the RGBD camera at any given positions and orientations in Habitat dataset.

A. Robot Locality Problem

We first demonstrate the phenomenon that after training a sample at a starting pose, there would be a lowland of training accuracy gain of training input samples around the starting pose, as shown in Fig. 1. This was achieved by first training the implicit model in NICE SLAM at the starting pose and then spawning the camera around the starting pose. Each triangle in Fig. 1 represents sampling pose and its color represents the reduction of mapping error (i.e., average difference in position and RGD value of each pixel between the inferred 3D dense map and the ground truth) after training on samples in that pose. The mapping error reduction value is normalized. This lowland of training accuracy gain we observed implies two key information: 1. beyond certain distances there is still

(a) Consecutive sampling: robot consecutively sampling around local states.

(b) Sparse / teleported sampling: robot only samples in the selected poses.

Fig. 2: Consecutive and sparse / teleported sampling.

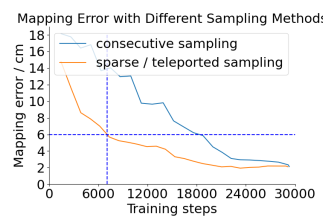
considerable training accuracy gain about the same area of interest, which means it often requires sampling from multiple angles and distances to fully attain full potential training accuracy gain for higher final training accuracy; 2. moving in and out such a lowland (as required by sampling from multiple angles and distances) for a robot would inevitably cause samples midway suffer low quality and low training accuracy gain (low training efficiency).

Then we carried out end to end experiments on consecutive / sparse / teleported sampling to examine how robot locality problem affects training efficiency and training throughput as shown in Fig. 3. The data of mapping error in Fig. 3 was recorded by checkpointing the implicit model every five minutes and comparing the inferred 3D dense map with the ground truth. The value is the addition of position drift in centimeters and RGD value drift averaged among each pixel. As shown in Fig. 3a, sparse / teleported sampling reduced average mapping error from 18 to 6 after training for about 7063 training steps, compared with about 18322 training steps taken by consecutive sampling. This represents sparse teleported sampling took only 38.55% of the number of training steps (and power consumption in online training) as consecutive sampling to reach the same level of training accuracy and represents 61.55% lower training efficiency in consecutive sampling compared with sparse teleported sampling.

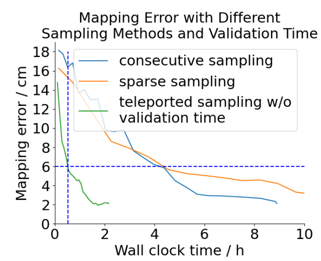
On the other hand, pausing the sampling midway in sparse sampling reduced the training throughput from 0.93 training steps per second to 0.36 training steps per second in sparse sampling as shown in Table I. Note that the training throughput in consecutive and sparse sampling is affected by the time consumption of model validation for decision making. This drop of training throughput leads to almost zero wall-clock time advantage for sparse sampling to reach mapping error of 6 compared with consecutive sampling. The reason is although each training step in sparse sampling produces higher training accuracy gain, intuitively sparse sampling can be approximated by replacing the non-selected samples in consecutive sampling with samples that has zero training accuracy gain, which does not better or even harms the final training accuracy from the view of wall-clock time. Ideally we should minimize the model validation time for decision making and achieve the training throughput of 2.82 training steps per second in teleported sampling. And the robot locality problem as discussed above and high time consumption of estimation of potential training accuracy gain all caused 8.02X and 8.21X training wall-clock time for consecutive sampling and sparse sampling to reach mapping error of 6 compared with teleported sampling as shown in Fig. 3b. In a word, innovation to better tradeoff between training efficiency and training throughput in active sampling for mobile robotic online training is required to optimize both time consumption and energy consumption for online training to reach high training accuracy.

TABLE I: Recorded Average Training throughput (iters/s)

Sampling Method	Training Throughput
consecutive sampling	0.93
sparse sampling	0.36
teleported sampling	2.82



(a) Consecutive sampling lowers training accuracy gain per training step



(b) Validation time cost further deteriorates training accuracy gain against wallclock time.

Fig. 3: Robot locality problem and validation time cost evidently slow down online training process.

B. Estimation of Potential Training Accuracy Gain

From the above discussion we know that although problematic, traditional active SLAM methods combined with validation of implicit model successfully direct the robot to construct 3D dense map of a complex room using implicit SLAM methods. We then further explore what affects the time consumption of validation of an AI model and found that the time consumption of model validation increased both as the number of sampled key frames in NICE SLAM increases and as the model size increases. Note that the number of sampled key frames in NICE SLAM needed to be validated increases as the online training and active sampling process proceed, and such pattern is common in the estimation of potential training accuracy gain other forms of AI applications, such as in a binary classification task, validating the collected samples to find out the samples whose posterior probability of being positive is nearest 0.5. With the time cost of model validation increasingly becoming the bottleneck of the active sampling and online training process, this finding calls for renovation of methods in estimation of potential training accuracy gain beyond traditional validation methods to drive this time cost away from the online training critical path.

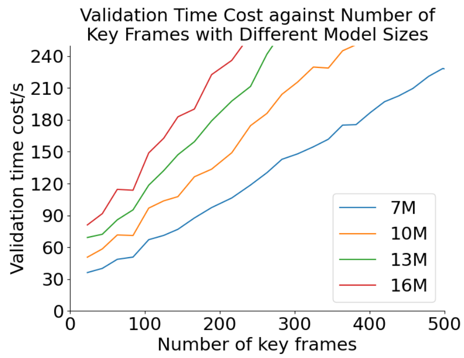


Fig. 4: Time consumption validating Implicit SLAM model with various model sizes (M: millions of parameters) and number of key frames

IV. Conclusion

In this paper, we explored the problems that hinder the practical active sampling for mobile robotic online training. They include the contradiction between training efficiency and training throughput rooted in the constraint in motion pattern of real-world robots and the excessive cost in introducing methods to estimate potential training accuracy gain in traditional active learning methods, and we are intended to raise research effort to tackle these problems. With practical active sampling for mobile robotic online training enabled, the AI models and applications deployed on mobile edge devices would be able to resiliently and sustainably retain long term high performance against changing real-world environments with as short as possible time and as little as possible energy consumption through active sampling.

References

[1] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual Domain Adaptation: A survey of recent advances," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 53–69, May 2015, conference Name: IEEE Signal Processing Magazine.

[2] P. Zhao and S. C. H. Hoi, "OTL: a framework of online transfer learning," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ser. ICML'10. Madison, WI, USA: Omnipress, Jun. 2010, pp. 1231–1238.

[3] Q. Wang, O. Fink, L. Van Gool, and D. Dai, "Continual test-time domain adaptation," pp. 7201–7211.

[4] L. Yang, W. Zhuo, L. Qi, Y. Shi, and Y. Gao, "ST++: Make Self-trainingWork Better for Semi-supervised Semantic Segmentation," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022, pp. 4258–4267, iSSN: 2575-7075.

[5] D. Zha, Z. P. Bhat, K.-H. Lai, F. Yang, Z. Jiang, S. Zhong, and X. Hu, "Data-centric Artificial Intelligence: A Survey," Apr. 2023, arXiv:2303.10158 [cs]. [Online]. Available: http://arxiv.org/abs/2303.10158

[6] S. Sadiq, T. Dasu, X. L. Dong, J. Freire, I. F. Ilyas, S. Link, M. J. Miller, F. Naumann, X. Zhou, and D. Srivastava, "Data Quality: The Role of Empiricism," *ACM SIGMOD Record*, vol. 46, no. 4, pp. 35–43, Feb. 2018. [Online]. Available: https://dl.acm.org/doi/10.1145/3186549.3186559

[7] N. Sambasivan, S. Kapania, H. Highfill, D. Akrong, P. Paritosh, and L. M. Aroyo, ""Everyone wants to do the model work, not the data work": Data Cascades in High-Stakes AI," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, ser. CHI '21. New York, NY, USA: Association for Computing Machinery, May 2021, pp. 1–15. [Online]. Available: https://dl.acm.org/doi/10.1145/3411764.3445518

[8] I. Lluvia, E. Lazkano, and A. Ansuategi, "Active Mapping and Robot Exploration: A Survey," *Sensors*, vol. 21, no. 7, p. 2445, Apr. 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/7/2445

[9] B. Settles, "Active Learning Literature Survey."

[10] S. Argamon-Engelson and I. Dagan, "Committee-Based Sample Selection for Probabilistic Classifiers," *Journal of Artificial Intelligence Research*, vol. 11, pp. 335–360, Nov. 1999, arXiv:1106.0220 [cs]. [Online]. Available: http://arxiv.org/abs/1106.0220

[11] G. Wilson and D. J. Cook, "A survey of unsupervised deep domain adaptation," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 5, jul 2020. [Online]. Available: https://doi.org/10.1145/3400066

[12] Y. Chen, M. Mancini, X. Zhu, and Z. Akata, "Semi-Supervised and Unsupervised Deep Visual Learning: A Survey," Aug. 2022, arXiv:2208.11296 [cs]. [Online]. Available: http://arxiv.org/abs/2208.11296

[13] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain Adaptation via Transfer Component Analysis," *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199–210, Feb. 2011, conference Name: IEEE Transactions on Neural Networks.

[14] P. Bachman, O. Alsharif, and D. Precup, "Learning with pseudo-ensembles," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, p. 3365–3373.

[15] M. Xu, Y. Song, Y. Chen, S. Huang, and Q. Hao, "Invariant EKF based 2D Active SLAM with Exploration Task," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. Xi'an, China: IEEE, May 2021, pp. 5350–5356. [Online]. Available: https://ieeexplore.ieee.org/document/9561951/

[16] E. Bonetto, P. Goldschmid, M. Pabst, M. J. Black, and A. Ahmad, "iRotate: Active Visual SLAM for Omnidirectional Robots," *Robotics and Autonomous Systems*, vol. 154, p. 104102, Aug. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889022000550

[17] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, "Nice-slam: Neural implicit scalable encoding for slam," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 786–12 796.

[18] E. Sucar, S. Liu, J. Ortiz, and A. J. Davison, "imap: Implicit mapping and positioning in real-time," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6229–6238.

[19] "The World's Smallest AI Supercomputer." [Online]. Available: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-xavier-nx/

[20] A. Szot, A. Clegg, E. Undersander, E. Wijmans, Y. Zhao, J. Turner, N. Maestre, M. Mukadam, D. Chaplot, O. Maksymets, A. Gokaslan, V. Vondrus, S. Dharur, F. Meier, W. Galuba, A. Chang, Z. Kira, V. Koltun, J. Malik, M. Savva, and D. Batra, "Habitat 2.0: Training home assistants to rearrange their habitat," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[21] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, D. Parikh, and D. Batra, "Habitat: A Platform for Embodied AI Research," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[22] E. Bonetto, P. Goldschmid, M. Pabst, M. J. Black, and A. Ahmad, "irotate: Active visual slam for omnidirectional robots," *Robotics and Autonomous Systems*, vol. 154, p. 104102, 2022.