Demo Abstract: Demonstrating Resource-Efficient SLAM in Virtual Spacecraft Environments

Ying Chen*, John Sarik†, Hazer Inaltekin‡, Maria Gorlatova*
*Duke University, Durham, †Nanohmics, Inc., Austin, TX ‡Macquarie University, North Ryde, NSW, Australia
*{ying.chen151, maria.gorlatova}@duke.edu, †jsarik@nanohmics.com, ‡hazer.inaltekin@mq.edu.au

***Abstract*—The performance of simultaneous localization and mapping (SLAM) systems is impacted by the constrained computation capabilities of mobile devices. Given that the advancement of these systems relies on accurate evaluation of SLAM performance, this issue is exacerbated by the difficulty in evaluating SLAM performance in practice, due to the unavailability of ground truth data. In this demo, we present SpacecraftWalk, a resource-efficient SLAM framework that constructs maps (of the environments) with minimal uncertainty under resource budgets. SpacecraftWalk is evaluated within virtual spacecraft environments (meeting NASA lighting standards) in game engine-based emulators that generate ground truth automatically. Demo participants will navigate in virtual environments while creating their own moving trajectories for evaluating SLAM. They will develop an intuition for how uncertainty-based map construction improves resource efficiency. This demonstration accompanies [1].**

I. Introduction

Simultaneous localization and mapping (SLAM) is crucial for applications such as robotic navigation, autonomous driving, and augmented reality. It concurrently maps the environment and tracks the mobile device's pose. While remarkable progress has been made on SLAM, two major challenges remain. First, the high computational demands of SLAM lead to performance loss when implemented on resource-constrained mobile devices. Second, due to the costly and time-consuming nature of obtaining ground truth pose data, SLAM performance evaluation is challenging in practice.

Improving resource efficiency is essential for running SLAM under resource constraints. Recent SLAM systems use heuristics to limit the information retained (e.g., camera frames) when estimating the device poses. Examples include only retaining the most recent frames [2] or retaining information that maintains a strong connection between adjacent frames [3]. Complementing these heuristic-based works, we retain the information based on the quantification of pose estimation uncertainty, which characterizes how the retained information contributes to the uncertainty.

To efficiently evaluate the SLAM performance, most works [4], [5] use benchmark datasets (e.g., EuRoC, TUM, SenseTime). However, it is challenging to assess SLAM performance in other visual environments. Facilitated by the game engine's capabilities of photorealistic scene simulation and accurate ground truth generation, our interactive demonstration allows participants to navigate within our target environment for testing SLAM performance on their trajectories.

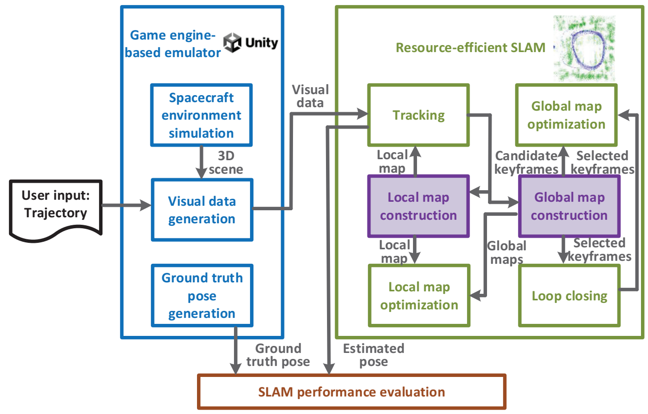


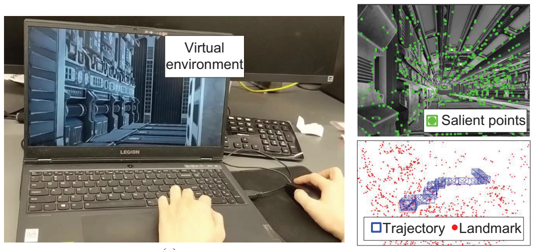
Fig. 1. SpacecraftWalk system overview. The game engine-based emulator we built contains realistic virtual spacecraft environments. The proposed resource-efficient SLAM is performed in the virtual environment to adapt to computation resource constraints.

We present SpacecraftWalk, a SLAM framework that achieves high performance under resource constraints and that is evaluated in game engine-based emulators on participants' trajectories. The demo accompanies [1], which formulates and solves an optimization problem that minimizes the pose estimation uncertainty under resource constraints. In SpacecraftWalk, resource-efficient SLAM is accomplished by adding new SLAM modules for selecting subsets of keyframes (i.e., representative camera frames) to construct limited-sized maps with minimal uncertainty. SpacecraftWalk runs in virtual spacecraft environments that meet NASA lighting standards [6]. It simulates the computation constraints of drones and robots that run SLAM to navigate in the spacecraft. We showcase that SpacecraftWalk can achieve higher pose estimation accuracy when evaluated on demo participants' trajectories in the virtual spacecraft environments, compared with the heuristic-based baseline which keeps recent keyframes under resource constraints.

II. System Design

The system architecture of SpacecraftWalk is shown in Fig. 1. We build a game engine-based emulator with realistic virtual spacecraft environments, in which the proposed resource-efficient SLAM is performed to maintain high tracking accuracy under resource constraints.

Game engine-based emulator. We *simulate a virtual spacecraft environment* in a game engine such as Unity. Within the environment, we control multiple parameters, such as illumination conditions, and texture, position, and orientation



(a) (b)

Fig. 2. SpacecraftWalk in action. (a) The demo participants use the mouse and arrow keys to move around and look in different directions in the virtual spacecraft. (b) The resource-efficient SLAM is executed on the visual data generated for the participants when they explore the virtual spacecraft.

of all objects. This facilitates the creation of realistic spacecraft environments with interior lighting that accommodates NASA lighting standards [6]. The emulator then lets users walk around and look in different directions in the virtual environments, where users control their trajectories by interacting with the environments (e.g., via a mouse, arrow keys, or a dedicated handheld controller). The emulator *generates visual data* which is rendered for users' specific points of view along the moving trajectories. The emulator also automatically *generates ground truth pose data* that is costly and time-consuming to obtain in practice.

Resource-efficient SLAM. We propose resource-efficient SLAM to combat the constrained computation resources on mobile devices navigating in the spacecraft environment. The components include canonical modules of most modern SLAM systems, and two added modules (local and global map construction highlighted in purple in Fig. 1).

We briefly introduce *canonical modules.* For each frame, the *tracking* module estimates the pose of the mobile device and determines if the frame should be a keyframe (e.g., based on similarity to previous keyframes). *Local and global map optimization modules* estimate a large number of states (e.g., keyframe poses, landmark positions), which are computationally expensive. *Loop closing module* fine-tunes the pose estimation by checking if the new keyframe revisits a place.

Two *added modules*, *local and global map construction modules*, aim to optimally construct the local and global maps (built using information contained in keyframes) under the computation resource constraints. Towards this end, we first quantify the uncertainty of local and global maps by characterizing how the keyframes and the connections between them contribute to the uncertainty. By minimizing the uncertainty under the constraints of available computation resources, the modules select a limited size of keyframes from candidate keyframes to build local and global maps.

SLAM performance evaluation. This module compares the ground truth pose automatically generated in the emulator and the estimated pose calculated by resource-efficient SLAM.

III. Interactive Demonstration

The demo follows the workflow shown in Fig. 1. Participants of our demo will gain an understanding of the advantages of using a game engine-based emulator for performing and evaluating SLAM algorithms, and will see how uncertainty-based map construction improves SLAM performance under resource limitations. A video of the demo is available online.[1]

The game engine-based emulator is built using Unity 2019.2.14f. We simulate the virtual spacecraft environment based on the Unity asset Modular Sci-fi Environment [7]. We run SpacecraftWalk on a Lenovo laptop with an AMD Ryzen 7 4700H CPU and an NVIDIA GTX 1660 Ti GPU.

Similar to the illustration in Fig. 2, participants interact with the virtual spacecraft environment using the mouse and arrow keys. In this way, they can view the visual data displayed on the laptop monitor while moving around and looking in different directions in the environment. The real trajectories of their positions and orientations within the virtual environment are automatically recorded. Upon completing the preset time interval for navigating the environment, participants can choose between two methods, the resource-efficient SLAM and the DropOldest baseline, to estimate their trajectories. DropOldest keeps recent keyframes to construct local and global maps for saving resources. For both methods, the participants will view two windows, one showing image processing results (e.g., identifying salient points) of visual image sequences, and the other displaying the estimation of the participant's trajectory and landmark positions in the virtual environment. The participants will then view the comparison between the real and estimated trajectories to get insights into the performance of the resource-efficient SLAM and the baseline method. The proposed resource-efficient SLAM reduces the tracking error by 62% compared with the DropOldest baseline.

Acknowledgments

This work was supported in part by NSF grants CSR-1903136, CNS-1908051, and CNS-2112562, NSF CAREER Award IIS-2046072, by an IBM Faculty Award, by the Australian Research Council under Grant DP200101627, and by NASA SBIR under Contract No 80NSSC22PB105.

References

[1] Y. Chen, H. Inaltekin, and M. Gorlatova, "AdaptSLAM: Edge-assisted adaptive SLAM with resource constraints via uncertainty minimization," in *Proc. IEEE INFOCOM*, 2023.

[2] P. Schmuck and M. Chli, "CCM-SLAM: Robust and efficient centralized collaborative monocular simultaneous localization and mapping for robotic teams," *J. Field Robot.*, vol. 36, no. 4, pp. 763–781, 2019.

[3] Y.-P. Wang, Z.-X. Zou, C. Wang, Y.-J. Dong, L. Qiao, and D. Manocha, "ORBBuf: A robust buffering method for remote visual SLAM," in *Proc. IEEE/RSJ IROS*, 2021.

[4] J. Xu, H. Cao, D. Li, K. Huang, C. Qian, L. Shangguan, and Z. Yang, "Edge assisted mobile semantic visual SLAM," in *Proc. IEEE INFOCOM*, 2020.

[5] A. J. Ben Ali, Z. S. Hashemifar, and K. Dantu, "Edge-SLAM: Edge-assisted visual simultaneous localization and mapping," in *Proc. ACM MobiSys*, 2020.

[6] NASA Johnson Space Center Habitability & Human Factors Lighting Environment Test Facility. (2016) Addressing Challenges to the Design & Test of Operational Lighting Environments for the International Space Station. [Online]. Available: https://ntrs.nasa.gov/api/citations/20160001038/downloads/20160001038.pdf

[7] Unity Asset Store. (2018) Modular Sci-fi Environment A. [Online]. Available: https://assetstore.unity.com/packages/3d/environments/sci-fi/modular-sci-fi-environment-a-107671

[1] https://sites.duke.edu/marialabyingchen/spacecraftwalk/