ENG 573 AR – Capstone Project
Proposal

STUDENT NAME: Narathip Rodwarna 672048079, Sridharan Subramanian 661699688, Yifei Li 657321173, Amey Gavale 652917301

INDUSTRY PROJECT ADVISOR: Paul Ardis

UIUC FACULTY PROJECT SUPERVISOR: -

UIUC FACULTY ADVISOR: **Professor Kris Hauser, Computer Science – Autonomy & Robotics**

INDUSTRY SPONSOR/CAMPUS RESEARCH GROUP: GE Aerospace Research

Attach a 1-2 page proposal that outlines the expected scope of the capstone project.

Information to include:

- Project Objectives
- Professional/Personal Objectives
- Plan for conduct as it relates to achieving overall internship objectives such as conceptual studies, design and build activities, feasibility studies, experimental work, and result analysis.
- Anticipated Deliverables
- Possible literature and research to be used

APRROVALS:

Industry Project Advisor: ____________________ **Date:** 25 June 2024

Faculty Project Supervisor: ____________________ **Date:** __________

Faculty Advisor: ____________________ **Date:** __________

Notes:

BENCHMARKING DIFFERENT SLAM APPROACHES

I. Introduction and Motivation

A. **Motivation**

To support autonomous avionics assets, maintaining continuous location awareness, even in the absence of reliable GPS and inertial sensor data, is crucial. The primary goal is to benchmark various VSLAM (Visual Simultaneous Localization and Mapping) techniques and explore effective combinations that can provide high-quality mapping. Our objective is to achieve this without placing excessive computational demands on the platform.

B. **Problem Setting**

The setup of this work is a VSLAM utilizing solely a monocular camera. The evaluation primarily focuses on camera angles ranging from a horizontal perspective(could start from 0 degrees) to a downward angle of 90 degrees. The obstacles range from 10 to 300 meters. Hence this includes top-down view and also view at horizon level.

II. Related Work

There are multiple advanced VSLAM algorithms, each with different pros and cons, as will be discussed.

A. **OpenVSLAM:** Stella_VSLAM is a fork for active development of [1]. This is an indirect SLAM implementation based on ORB-SLAM2, and UcoSLAM in terms of feature extraction. One of the main advantages that makes this approach attractive is that this already supports perspective, equirectangular and fisheye camera models, making this robust to diverse camera models used in aerial systems at GE Aerospace. Localizing in other environments with pre-built maps is another advantage when exploring new datasets.

B. **LSD SLAM:** LSD SLAM introduces direct SLAM techniques by leveraging dense depth maps for real-time operation and high accuracy in large system environments. The omni-directional LSD SLAM algorithm is being tested by capturing videos with fisheye lenses and its accuracy and robustness are evaluated under strong rotational movements [2].

C. **ORB-SLAM3:** It integrates ORB (Oriented FAST and Rotated BRIEF) feature detection to track keypoints and construct a map of the environment in real-time. Additionally, ORB-SLAM3 incorporates loop closure and relocalization to correct for drift and improve long-term mapping stability [3].

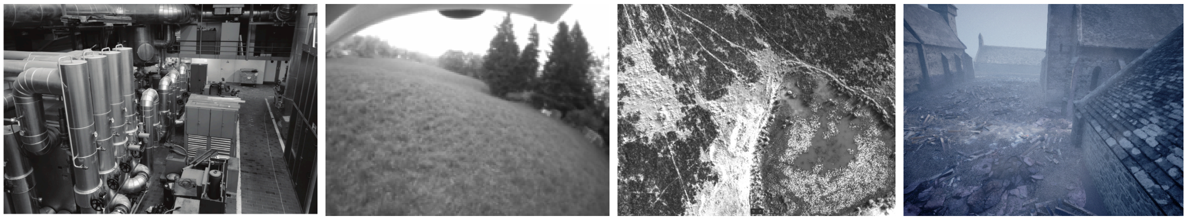
III. Goals

A. Evaluate and compare various VSLAM algorithms across multiple datasets to assess their performance and robustness.

B. Fuse different VSLAM methods to optimize results and enhance performance.

IV. Datasets

Four distinct datasets are utilized for evaluation purposes. Firstly, the EuRoC MAV Dataset [4], widely employed in Visual SLAM research, offers 11 indoor datasets recorded using a Micro Aerial Vehicle (MAV). Secondly, the UZH-FPV Drone Racing Dataset [5] emphasizes extreme accelerations, rotations, and apparent motion, particularly taxing for vision sensor state estimation. Thirdly, the Kagaru Airborne Stereo Dataset [6] provides a downward-facing perspective, facilitating evaluations from a specific camera angle. Lastly, the TartanAir Dataset [7] leverages photo-realistic simulation environments within Unreal Engine, featuring varied lighting conditions and weathers to test robustness across different scenarios.



(a) (b) (c) (d)

Fig.1 Example of dataset (a) EuRoC MAV (b) UZH-FPV Drone Racing (c) Kagaru Airborne (d)TartanAir

V. Evaluation Metrics

A. Absolute Trajectory Error (ATE) is used to evaluate the accuracy of the estimated trajectory. It is utilized by all of our target algorithms, so it can be compared with the baseline in the original paper.
B. Relative Pose Error (RPE) measures the error in the relative pose between pairs of poses at different time steps along the trajectory.
C. Running time of each process: As running in real-time is also crucial for certain applications, we aim to investigate the running time of all algorithms.

VI. Data Post-processing

To evaluate the results against the ground truth and ensure comparability between algorithms, the Evo package is utilized for alignment and for calculating ATE and RPE [8]. The Sim(3) Umeyama alignment is implemented to handle rotation, translation, and scale transformation.

VII. Fusion Idea

There are a few possible categories of ideas on fusion.

A. **Trajectory Level Averaging:** After obtaining the time synchronized trajectories from multiple SLAM modules, weighted average can be applied on them with the help of the ATEs mentioned above in the metrics. In this way, trajectory level fusion takes the final trajectories from multiple methods as input without modifying the core algorithm. Based on the corresponding ATE, weighted average can be done(for instance, α*ORB-SLAM3 +(1-α)*OpenVSLAM). Plotting different complementary filter values against the performance allows for a better and quicker analysis of which α to select for an environment similar to the dataset used. Adapting this method to multiple camera models and datasets will improve robustness.
B. **Long Short Term Memory:** LSTM is another way to potentially improve the accuracy of the final output of a SLAM system as they are good at learning on sequential data. Here, the idea is to train the RNN with trajectory inputs from datasets like EuRoC as mentioned above and target data as the trajectory output from method A. This will be eventually tested with trajectories from a variety of datasets to evaluate the RNN's performance in

this context.

C. **Feature Level:** In certain conditions, such as low-texture environments, solely relying on a particular feature matching may not be sufficient. In these cases, incorporating other feature extraction backends like ORB3 into OpenVSLAM while keeping the same optimization and mapping modules will be considered to enhance robustness.

VIII. Anticipated Challenges

A. **Computational Constraints:** One of the primary challenges will be ensuring that the selected SLAM algorithms operate efficiently within the limited computational resources available on the autonomous platforms. Balancing performance with computational efficiency will be critical.

B. **Environmental Factors:** The performance of SLAM algorithms can be significantly affected by environmental factors such as lighting conditions, presence of dynamic objects, and varying terrain. Addressing these factors to maintain consistent performance will be a key challenge.

IX. References

[1] S. Sumikura, M. Shibuya, and K. Sakurada, “OpenVSLAM: A Versatile Visual SLAM Framework,” doi: https://doi.org/10.48550/arXiv.1910.01122, 2019.
[2] J. Engel, T. Schöps, and D. Cremers, “LSD-SLAM: Large-Scale Direct Monocular SLAM,” European Conference on Computer Vision (ECCV), 2014.
[3] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M.M. Montiel, and J. D. Tardós, “ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM,” IEEE Transactions on Robotics, vol. 37, pp. 1874-1890, December 2021.
[4] M. Burri et al., “The EuRoC micro aerial vehicle datasets,” The International Journal of Robotics Research, vol. 35, pp. 1157-1163, 2016.
[5] J. Delmerico, T Cieslewski, H. Rebecq, M. Faessler, and D. Scaramuzza, “Are We Ready for Autonomous Drone Racing?,” IEEE International Conference on Robotics and Automation (ICRA), Montreal, 2019.
[6] M. Warren et al., “Large Scale Monocular Vision-only Mapping from a Fixed-Wing sUAS,” International Conference on Field and Service Robotics, 2012.
[7] W. Wang et al., “Tartanair: A dataset to push the limits of visual slam,” IEEE/RSJ International Conference on Intelligent Robots and Systems, 2020.
[8] M. Grupp, “Python package for the evaluation of odometry and SLAM,” michaelgrupp.github.io/evo/ (accessed June 12, 2024).