A MUILTISENSOR DATA FUSION APPROACH FOR SIMULTANEOUS LOCALIZATION AND MAPPING

Zhekai Jin *
Department of Electrical Engineering
Cooper Union for the Advancement of Science and Art

Yifei Shao **Minjoon So**
Department of Mechanical Engineering
Cooper Union for the Advancement of Science and Art

ABSTRACT

Simultaneous localization and mapping (SLAM) has been an emerging research topic in the fields of robotics, autonomous driving, and unmanned aerial vehicles over the last 30 years. Unfortunately, SLAM research is generally inaccessible for student researchers due to expensive hardware and painful software setup. By introducing a loosely coupled, modular multi-sensor data fusion architecture, we present an autonomous driving research platform that can be adapted to computing platforms with various computational constraints and serve multiple applications and educational purposes. Our goal is to create an easily accessible SLAM module with cost-friendly hardware dependencies and minimal software setup for SLAM researchers, teachers, and learners.

1 Introduction

Simultaneous localization and mapping (SLAM) is a well-developed research topic regarding the problem of estimating the trajectory of a moving robot and building a map of its environment simultaneously. To make a robot be aware of its location and navigate in an environment where a prior map is not available, one has to solve the SLAM problem. Though the formulation of the SLAM problem has been well established and the robotics research community has seen tremendous progress over the past few decades, there are still a lot of open problems left unsolved including fail-safe SLAM algorithms, efficient map representations, and resource-aware SLAM systems [1]. Furthermore, a general SLAM solution that can run in real time and adapt to the available computing platforms has not yet been proposed. Also, a lot of existing SLAM algorithms fail to identify previously visited locations and correct the corresponding odometry estimations (known as loop closure).

We have introduced a robust and flexible multi-sensor data fusion architecture that leverages state-of-the-art Lidar algorithms. Our system provides custom configurations to allow further research in innovative image registration algorithms, frame matching algorithms, backend nonlinear least-squares pose-graph solvers, etc. We have also supplemented the multi-sensor data fusion model with the necessary hardware, control, and planning module to provide a cost-friendly autonomous driving platform. This platform, with the physical form of a differential drive robot, is capable of driving around in an unknown environment, creating a map of its surroundings and performing autonomous navigation to any targeted location in the self-created map.

The rest of this paper is organized as follows. In Section 2 we give a high-level overview of multi-sensor data fusion. Then, we present our multi-sensor data fusion pipeline in Section 3. The system implementation, along with hardware and software dependencies, is then described in Section 4. Our experimental results are shown in Section 5. Finally, we present our conclusions and explore possible future work.

2 Background

An autonomous mobile robot operates by processing the information of its surroundings and then making intelligent and accurate driving decisions. This means that the perception system, the very first module to acquire peripheral information on which other parts of the platform depend, needs to be as robust and accurate as possible to safeguard the performance of the whole system. A system operating with a single sensor often fails to capture the rich physical attributes of the environment. The camera, a typical visual perception sensor, is likely to fail in environments where the lighting intensity is dramatically changed or the lighting intensity is particularly low. On the other hand, a radar sensor

*The code and models presented in this work have been publicly available at https://github.com/ZhekaiJin/the-Cooper-Mapper has a longer sensing distance and lower computational demands but is less accurate than the light imaging detection and ranging (Lidar) in terms of angular accuracy. Due to the inherent vulnerability of the single-sensor system, multi-sensor data fusion has become an overarching paradigm for avoiding single-point failure and enhancing the system with reduction in ambiguity and uncertainty, increase in accuracy, robustness against interference, etc [4]. For instance, Tesla's Autopilot leverages a hardware suit of eight cameras, a forward-looking radar, and twelve ultrasonic sensors to ensure 360 degrees of visibility for its perception system.

The introduction of the multi-sensor data fusion model, though well-built in theory, does lead to some practical challenges including how to handle noise in the operation, data imputation, the determination of where in the processing pipeline to perform the fusion algorithm, and how and when to keep or drop the previously acquired information. Moreover, due to inevitable sensor manufacturing variations, extreme external calibration effort across the sensors is often needed to ensure the performance of the fusion architecture.

3 Project Description

To address the problems mentioned in Section 2 and to maximize the cost efficiency along with the mapping accuracy, we introduced a loosely coupled time-stamp-based multi-sensor data fusion architecture which leverages camera and Lidar data as default. On top of the default fusion setup, the system provides custom configuration freedom for researchers to add additional sensor modality and experiment with various fusion algorithms. The default fusion architecture, which leverages the state-of-the-art Lidar SLAM pipeline and multiple visual place recognition algorithms, ensures the basic functionality for accurate mapping with global closure and reduces the unnecessary exterior calibration effort.

While state-of-the-art visual and Lidar SLAM algorithms are equivalent in terms of accuracy, visual pipelines are more robust in dynamic scenes and less expensive computationally. Lidar SLAM systems, on the other hand, are more consistent and less sensitive to changes in illumination and appearance due to their heavy dependence on the geometric structure of the surrounding. Even though most of modern Lidar SLAM algorithms have shown impressive results [2], they failed to address the drift problem over time with the assumption that the world is an "infinite corridor" [1]. Therefore, we propose a fusion mechanism that supplements the Lidar SLAM algorithm with visual stereo image data for place recognition and drift correction. In this section, we introduce our perception system by explaining the selection of the sensors, the base SLAM module, and the data fusion model proposed for the heterogeneous sensors involved.

3.1 Sensor Choices

RPLiDAR A2: Whereas Hokuyo UST-20LX scanners are now the standard 2D Laser scanners for SLAM research, we found the RPLIDAR A2 scanners to be a cheaper option. Though 3D Laser scanners have the advantages of high resolution and a 360 degree range for 3D SLAM algorithm research, their high cost made the actuated 2D Lidar more suitable for our purpose. With a reasonable cost, the RPLIDAR A2 can perform 360 degree scans within a range of 12 meters or 18 meters and generate 8000 points per second with a 15 Hz sampling rate. Also DJI has released Livox Mid-40, a 3D Lidar with a reasonable price, which future researchers with a generous budget could consider for dense mapping purposes.

ZED stereo camera: ZED is the best-suited camera for our platform not only for an educational purpose with its detailed API documentation but also for the smooth integration with the Robotics Operating System (ROS) which most of robotics research uses. With its high resolution and frame rate, ZED can serve multiple applications such as depth perception, positional tracking and 3D mapping.

3.2 Base SLAM module

We present a SLAM module which extends the state-of-the-art Lidar odometry estimator, LOAM [6], with back-end pose-graph optimization to correct drift and a place recognition system to allow global loop closure. LOAM, with its high accuracy, robustness and real-time operation, takes in raw 3D point clouds, calculates the rigid transformation due to the corresponding sensor motion and outputs the global pose estimation, a local representation of the map, and the registered point clouds. The original work has been refactored, optimized, and made modular in this work to support custom configuration and allow smooth adaption to other SLAM backend solutions.

Due to the inherent drifting error in incremental odometry estimators like LOAM, an online pose estimation back-end is needed in the system to build the pose graph based on the LOAM odometry estimation and correct LOAM odometry estimation from drifting error by performing re-localization based on the visual data. Re-localization takes place if the system identifies previously visited places. To identify previously visited places in the existing internal map, the system has to periodically query the place recognition module, which relies on the visual stereo data; this will be explained in Section 3.3.

3.3 Data Fusion Mechanism

We propose a modular multi-sensor data fusion pipeline, where Lidar is set as the default sensor for odometry estimation and visual stereo data is leveraged to perform place recognition. The Lidar-based SLAM backend keeps a set of keyframes to represent the sensor trajectory, each has an associated time stamp. With the stereo camera running constantly, the system registers stereo image data to the latest keyframe and performs cloud matching with all previously registered keyframes to find potentially matched frames. A pose graph optimization backend is running constantly to manage the environment mapping and correct odometry estimation by querying the visual place recognition system. We provide multiple state-of-the-art visual frame matching algorithms such as visual bag of words and SegMatch. Additionally, the system is able to incorporate other real-time matching algorithms and fuse with the result of existing matching algorithms.

When the system detects a matched frame, it calculates the transformation between the clouds of the associated keyframes using the iterative closest point (ICP) algorithm and adds a new edge to the pose graph representation of the existing map. Then, the estimated pose is fed back to the incremental odometry estimator to correct its internal motion estimation and perform the re-localization functionality. The whole sensor fusion pipeline is summarized in Figure 1.

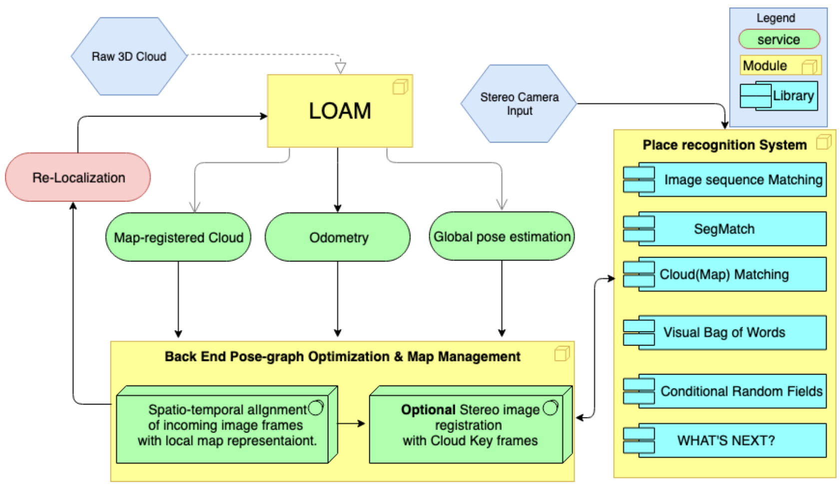

Figure 1: Sensor Fusion Architecture

4 System Implementation

To make our vehicle reasonably affordable and easier to assemble, we designed our vehicle to be an educational and cost-friendly research platform with minimal software setup on which versatile applications could be run. The hardware and computing platform are introduced in this section with key design features including affordability and versatility. It is worth noting that the components used for the vehicle are resources that are inexpensive, with a total cost approximated to $1200.

4.1 Hardware

We designed a custom differential drive chassis on which any electronics and hardware can be installed. We relied on easily accessible computer aided design (CAD) software and prototyping tools. Then two Pololu 12V gear motors are used to drive the rear wheels with a 2000-count-per-rev encoder mounted on each motor. Having two independently-driven rear wheels gives the platform two degrees of freedom for intuitive manipulation and control. In addition, the built-in encoders enable wheel speed control and could provide inaccurate odometry information to the system for reference. Also, a custom PCB board is used to connect electronic components and divide an electrical power feed from the batteries into subsidiary systems. Two 12V Lithium-ion batteries are used to supply power to the computing hardware and motors separately, which prevents the motor's transient voltage from interfering with the computing hardware.

4.2 Computing Platform

The Nvidia Jetson TX2 is a fast, power-efficient embedded computing device which is used as the on-board computing processor. Jetson supports Ubuntu naively for ROS integration and provides the necessary processing power for online

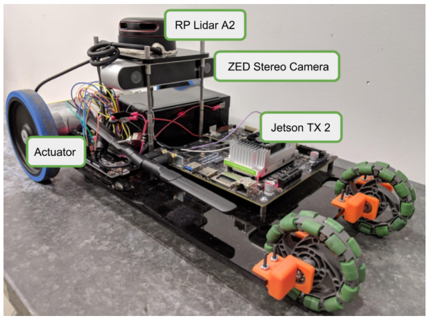

Figure 2: Platform Overview

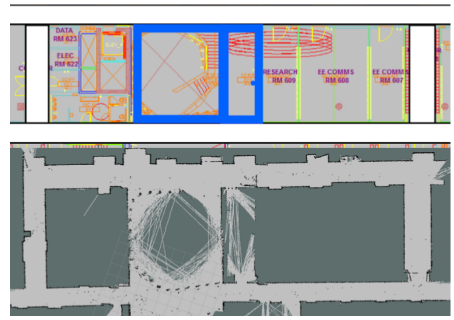

Figure 3: Top: Floor Plan, Bottom: Mapping Result

3D mapping algorithms. The Arduino Uno is used along with the Jetson computer as an expansion to the GPIO and interrupt pins of the Jetson. Acting as a middleman, it exchanges messages between the hardware and the Jetson board. A photograph of our device with components labeled can be seen in Figure 2

5 Experimental Evaluation

To evaluate the accuracy of the odometry estimation of our proposed multisensor data fusion architecture, we fully tested our algorithm against the publicly available KITTI odometry benchmark dataset [2]. The result was evaluated by the metrics employed by KITTI and compared with the LOAM module's result. Our architecture, with the default setup, has shown equivalent results with LOAM for KITTI sequence 00 and better performance than LOAM for KITTI sequence 05 by generating a trajectory map closer to the ground truth value in places where loop closure takes place. We also ran our algorithm with a real-world indoor environment, the 6th floor of our academic building by adapting the state-of-the-art 2D mapping algorithm, Cartographer [3] to our pipeline. A 2D occupancy grid map was generated as the result and is shown in Figure 3 with a comparison to the ground truth floor plan.

6 Conclusion

A new, integrated, and modular sensor fusion architecture has been developed and fully tested against a publicly available data set. Experiments have validated the hypothesis that by leveraging the redundancy across heterogeneous sensors, multi-sensor data fusion improves accuracy and robustness for applications such as mapping and motion estimation. In addition, the modular pipeline provides robotics researchers freedom to adapt and experiment with related algorithms.

There are many directions that this work can be expanded. For the multisensor data fusion model, pre-built models for sensors of different modalities can be developed. For example, a model can be built for the inertial measurement Unit (IMU), which is often used in modern SLAM algorithms to improve the accuracy and robustness of mapping [5]. While we primarily focused on the perception system of the autonomous driving platform, the control and planning modules of the platform can be further developed to provide more research possibilities for the future users of our platform.

References

[1] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, 2016.

[2] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[3] Wolfgang Hess, Damon Kohler, Holger Rapp, and Daniel Andor. Real-time loop closure in 2d lidar slam. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1271–1278. IEEE, 2016.

[4] KS Nagla, Moin Uddin, and Dilbag Singh. Multisensor data fusion and integration for mobile robots: A review. *IAES International Journal of Robotics and Automation*, 3(2):131, 2014.

[5] Meixiang Quan, Songhao Piao, Minglang Tan, and Shi-Sheng Huang. Tightly-coupled monocular visual-odometric slam using wheels and a mems gyroscope. *arXiv preprint arXiv:1804.04854*, 2018.

[6] Ji Zhang and Sanjiv Singh. Loam: Lidar odometry and mapping in real-time.