Visual Inertial Lidar Odometry and Mapping (VI-LOAM) fusion for UAV-based Parcel Delivery

by

©Didula Dissanayaka

A Thesis submitted to the School of Graduate Studies in partial fulfillment of the requirements for the degree of

Master of Engineering

Memorial University of Newfoundland

20 October 2022

St. John's Newfoundland

Abstract

This study presents the design, implementation, and validation of a robust GPS-aided multi-sensory odometry and mapping system for an end-to-end UAV-based parcel delivery application. There are two main approaches for UAV navigation, GPS navigation and GPS denied navigation. However, the existing GPS navigation solutions sometimes produce degenerative results due to GPS loss, multipath signals, and spoofing events. On the other hand, GPS denied navigation solutions suffer from inherent drift and sensor degradation scenarios. Additionally, the existing navigation solutions do not comply with UAV safety regulations when possible sensor failures end-to-end navigation requirements is considered.

Therefore, this thesis focus on developing a robust UAV navigation system by integrating visual, inertial, and lidar sensors with GPS to overcome the limitations of existing navigation solutions. Three significant contributions are produced in this study. First, a numerical study to evaluate the possibility of incorporating GPS with a GPS denied navigation solution for improved performance and safety regulatory compliance. Second, the development of a novel UAV navigation architecture combining visual, lidar, and inertial sensors that is robust for environmental degradation and aggressive motion. Third, integrating GPS with the novel UAV navigation architecture for improved accuracy. Additionally, this study presents results and a comparison study of the proposed navigation system and state-of-the-art navigation systems for different online benchmark datasets and in-house datasets. Moreover, the proposed GPS-aided UAV navigation system is evaluated against compliance with the associated safety regulations under different GPS scenarios.

keywords: Unmanned aerial vehicle (UAV), Navigation, Simultaneous localization and mapping (SLAM), UAV-based parcel delivery, Optimization, Visual inertial lidar odometry and mapping (VI-LOAM).

Acknowledgements

I wish to express my sincere appreciation to my supervisors Dr. Oscar De Silva, Dr. George Mann, Dr. Thumeera Wanasinghe and Dr. Ray Gosine for guiding me through my Master of Engineering degree. Their knowledge, experience and encouragement was immensely helpful to successfully complete my research and write up the thesis.
I place on record, my sincere thanks to National Research Council of Canada's artificial intelligence for logistics program, Natural Sciences and Engineering Research Council of Canada and Memorial University of Newfoundland for their financial and intellectual support for this research project.
I would like to extend my gratitude to all colleagues at Intelligent systems lab namely, Mr. Nushen Seneviratne, Mr. Ravindu Thalagala, Mr. Eranga Fernando, Mrs. Sachithra Attapattu, Mr. Mihiran Galagedarage Don, Dr. Mahmoud Abd El Hakim and Mr. Sahan Gunawardena for their support, advise, and motivation throughout the course.
I am also grateful to my parents, other family members, and friends for their moral support.

Table of Contents

Abstract **ii**

Acknowledgments **iii**

Table of Contents **vi**

List of Tables **vii**

List of Figures **x**

1 Introduction **1**

1.1 Motivation 1
1.2 Problem Statement 5
1.2.1 Problem I: GPS errors and mapping capability 6
1.2.2 Problem II: Inherent drift in GPS denied navigation solutions 6
1.2.3 Problem III: Unavailability of public packages for state-of-the-art robust UAV VLOAM navigation algorithms 7
1.2.4 Problem IV: GPS is not incorporated with robust multi-sensory UAV navigation architectures 8
1.3 Objective and Expected Contributions of the Research 8
1.4 Organization of the Thesis 9

2 Background **11**

2.1 UAV Safety Regulations 11
2.1.1 Performance Evaluation 13
2.1.1.1 Root-Mean-Square Error (RMSE) 13
2.1.1.2 Percentage Position Drift 14
2.2 UAV Navigation Methods 14
2.3 Types of Sensor Combinations 15
2.3.1 GNSS-based Navigation Methods 15
2.3.2 Vision-based Navigation Methods 16
2.3.3 Lidar-based Navigation Methods 17
2.3.4 Vision, lidar Combined Navigation Methods 17
2.4 Sensor Fusion Approaches 18

2.4.1 Filter-based Approaches 18
2.4.2 Optimization-based Approaches 19
2.4.3 AI-based Approaches 19

3 Numerical Study of Navigation Methods for UAV-based Parcel Delivery 22
3.1 Introduction 22
3.2 GPS/INS Based Navigation Methods 23
3.3 GPS Aided VLOAM Navigation 26
3.3.1 VLOAM Navigation 26
3.3.2 GPS and VLOAM Integration 29
3.4 Evaluation of Navigation Methods 31
3.4.1 Sensor Parameters 31
3.4.2 Results 35
3.4.3 Summary 37

4 VI-LOAM Version 1.1 40
4.1 Introduction 40
4.2 VI-LOAM Version 1.1 Architecture 40
4.2.1 Visual-Inertial Odometry (VIO) 41
4.2.1.1 Ceres Solver 42
4.2.1.2 Sliding Window 43
4.2.1.3 VIO Motion Estimation 44
4.2.2 Scan Registration 45
4.2.2.1 Planar and Edge Point Extraction 46
4.2.3 Lidar Odometry (LO) 46
4.2.3.1 LO Motion Estimation 47
4.2.3.2 Initialization 48
4.2.4 Lidar Mapping 49
4.2.4.1 Visual pose constraints 50
4.3 Results 51
4.3.1 Datasets 51
4.3.1.1 KITTI Dataset 51
4.3.1.2 AI4L Payload Data 53
4.3.2 KITTI Dataset Results 56
4.3.3 AI4L Payload Data Results 58
4.3.3.1 AI4L Payload Dataset #1 60
4.3.3.2 AI4L Payload Dataset #2 61
4.3.4 Execution Time Results 63
4.4 Summary 65

5 VI-LOAM Version 1.2 **67**
5.1 Introduction 67
5.2 VI-LOAM Version 1.2 Architecture 67
5.2.1 Depth-Enhanced Visual Inertial Odometry 68
5.2.2 VIO Frame to Frame Constraints 69
5.2.3 VIO Global Orientation Constraints 70
5.2.4 Global Fusion 71
5.3 Results 73
5.3.1 LVI-SAM Dataset 73
5.3.2 LVI-SAM Dataset Results 74
5.4 Summary 77

6 Conclusion and Future Work **79**
6.1 Research Summary Based on Objective I 79
6.2 Research summary Based on Objective II 80
6.3 Research Summary Based on Objective III 80
6.4 Publications 81
6.5 Future Directives 81

Bibliography **xi**

List of Tables

2.1 Transport Canada RPAS Safety Assurance Accuracy Requirements . 12
2.2 Type of sensor combinations and their performance 21

3.1 IMU sensor parameters 32
3.2 Sensor performance parameters 34
3.3 Maximum errors for each navigation solution 37

4.1 Performance evaluation of each navigation method for KITTI benchmark online dataset 58
4.2 Performance evaluation of each navigation method for Payload dataset sequences 63
4.3 Execution time statistics for the laptop computer (in seconds) 63
4.4 Execution time statistics for the NVIDIA Jetson Xavier module (in seconds) 64

5.1 Performance of GPS aided VI-LOAM version 1.2 for different regions of GPS availability 78

List of Figures

1.1 The Key modules of UAV autonomy: guidance, navigation, and control (Guidance) systems 3

3.1 Loosely-coupled GPS/INS integration [1] 24
3.2 Tightly-coupled GPS/INS integration [1] 25
3.3 Overview of the VLOAM pipeline [2] 26
3.4 Overview of framework to fuse the local and global sensors 29
3.5 An illustration of the global pose graph structure with local and global factors 30
3.6 Actual Trajectory of the simulation 32
3.7 Percentage position error for the distance traveled of VLOAM system for sequence 13 of KITTI dataset [3] 33
3.8 Rotation error for the distance traveled of VLOAM system for sequence 13 of KITTI dataset [3] 34
3.9 Results of the simulated VLOAM system for 00th sequence of KITTI dataset 35
3.10 Absolute position error of IMU Only, VLOAM Only, GPS/INS and GPS/VLOAM with respect to time 36
3.11 Orientation error of IMU Only, VLOAM Only, GPS/INS and GP-S/VLOAM with respect to time 36
3.12 Error states with two standard deviations of the mean for IMU only navigation. Green: Region with DGPS (0s to 723s time interval or 0 to 5km), Blue: Region without DGPS correction (723s to 1340s time interval or 5km to 15km), Red: Region with multipath errors (1061s to 1118s time interval or 10km to 11km) 38
3.13 Error states with two standard deviations of the mean for VLOAM only navigation. Green: Region with DGPS (0s to 723s time interval or 0 to 5km), Blue: Region without DGPS correction (723s to 1340s time interval or 5km to 15km), Red: Region with multipath errors (1061s to 1118s time interval or 10km to 11km) 38

3.14 Error states with two standard deviations of the mean for GPS/INS navigation. Green: Region with DGPS (0s to 723s time interval or 0 to 5km), Blue: Region without DGPS correction (723s to 1340s time interval or 5km to 15km), Red: Region with multipath errors (1061s to 1118s time interval or 10km to 11km) 39
3.15 Error states with two standard deviations of the mean for GPS/VLOAM navigation. Green: Region with DGPS (0s to 723s time interval or 0 to 5km), Blue: Region without DGPS correction (723s to 1340s time interval or 5km to 15km), Red: Region with multipath errors (1061s to 1118s time interval or 10km to 11km) 39

4.1 Architecture of VI-LOAM version 1.1 41
4.2 Visual inertial odometry module 41
4.3 Illustration of sliding window optimization for VIO with IMU and camera factors 44
4.4 Scan registration module 45
4.5 Lidar odometry module 47
4.6 Lidar mapping module 49
4.7 Illustration of mapping process [4] 50
4.8 Recording zones of the KITTI dataset. This figure shows the GPS traces of the recordings [5] 52
4.9 KITT dataset sensor setup with their coordinate frames [5] 53
4.10 AI4L Payload sensor setup 54
4.11 GPS traces of one of the datasets captured around the Memorial university of Newfoundland, Canada 55
4.12 KITTI dataset sequence 10 results for each navigation solution 57
4.13 Estimated paths and ground truth with the generated map for the AI4L payload dataset #1. Colors illustrate different navigation methods and ground truth. Yellow track is for ground truth. Blue track is for standalone VIO and Green track is for VI-LOAM version 1.1 estimation. 59
4.14 Estimated paths and ground truth with the generated map for the AI4L payload dataset #2. Colors illustrate different navigation methods and ground truth. Yellow track is for ground truth. Blue track is for standalone VIO and Green track is for VI-LOAM version 1.1 estimation. 59
4.15 Results of each navigation solution for AI4L payload dataset #1. . . . 60
4.16 Results of each navigation solution for AI4L payload dataset #2. . . . 61
4.17 Box and whisker plot of execution times for the laptop computer . . . 64
4.18 Box and whisker plot of execution times for the NVIDIA Jetson Xavier module 65

5.1 Architecture of VI-LOAM version 1.2 68

5.2 Depth association to visual features using lidar depth map [6] 69
5.3 Factor graph of the global fusion optimization 72
5.4 Estimated path and GPS position for the LVI-SAM dataset #1. Yellow track is for GPS position and Green track is for GPS aided VI-LOAM estimation. 75
5.5 Estimated path and GPS position for the LVI-SAM dataset #2. Yellow track is for GPS position and Green track is for GPS aided VI-LOAM estimation. 75
5.6 Estimated path when random GPS error spikes occur.Yellow track is for GPS position and Green track is for GPS aided VI-LOAM estimation. 76
5.7 Estimated path in the region of continuous GPS errors.Yellow track is for GPS position and Green track is for GPS aided VI-LOAM estimation. 77

Abbreviations

AI Artificial intelligence

BVLOS Beyond Visual Line of Sight

DNN Deep Neural Network

DoF Degree Of Freedom

EKF Extended Kalman filter

FAA Federal Aviation Administration

FAST Features from Accelerated Segment Test

GNSS Global Navigation Satellite System

GPS Global Positioning System

ICAO International Civil Aviation Organization

IMU Inertial Measurement Unit

INS Inertial Navigation Systems

LO Lidar Odometry

LOAM Lidar Odometry and Mapping

RMSE Root Mean Square Error

RPAS Remotely Piloted Aircraft System

SLAM Simultaneous Localization and Mapping

UAV Unmanned Aerial Vehicle

UKF Unscented Kalman filter

UWB Ultra Wide Band

VI-LOAM Visual Inertial Lidar Odometry and Mapping

VINS Visual-inertial Navigation Systems

VIO Visual Inertial Odometry

VLOAM Visual Lidar Odometry and Mapping

Chapter 1

Introduction

In this chapter, the motivation for this thesis, and an overview of available navigation methods for Unmanned aerial vehicle (UAV) based end-to-end delivery and their associated limitations are presented. The problem statement of the thesis will then be introduced, and the outstanding issues of existing UAV navigation methods for end-to-end delivery addressed in this thesis will be outlined. Finally, objectives and expected contributions of this thesis will be highlighted along with the organization of the thesis.

1.1 Motivation

UAV-based delivery has gained popularity among many industries as a cost-effective, low-carbon-footprint solution for goods delivery [7–11]. Example operational scenarios include last-mile goods delivery [12], regional air transit [13], delivery to remote communities [14], emergency medical supplies [15], and providing services to offshore and marine platforms [16]. These applications demand UAVs to operate close to critical infrastructure and human traffic during transit [17]. Due to the risks of causing human injuries and damage to properties, these UAVs need to be fault-tolerant while adhering to the highest safety standards imposed by regulatory bodies [18].

To facilitate end-to-end delivery operations, semi-autonomous or fully-autonomous capability of the UAV systems is desired depending on the level of human pilot intervention that is necessary [19]. The levels of autonomy can vary from low-level controls such as attitude heading control of remotely-piloted UAVs to fully autonomous execution of preprogrammed delivery operations. UAV autonomy functions mainly consist of three main modules [20]:

1. Guidance: planing an optimal mission from the current location to a destination avoiding any pre-known obstacles and controlled air-spaces. This can also include local predictive trajectory planning to handle behaviours such as detect and avoid (DAA) and landing.

2. Navigation: finding pose[1], speed, and other states of interest (locations of obstacles, map of the environment) of the platform with respect to a pre-defined navigation frame.

3. Control: generating a sequence of control commands that drives the platform along an optimal route.

As shown in Figure 1.1, the guidance system is responsible for generating instructions related to path planning and mission planning, i.e., what state trajectory that UAV should follow to accomplish the given mission. The control system in turn operates the aircraft controls such as thrust, elevators to follow the trajectory generated by the guidance systems. The guidance systems updates its path and mission plans according to the current state given by the navigation system. Also, state vector provided by navigation system acts as a feedback to the control system. Therefore, guidance and control systems rely on having an accurate estimation of the UAV

[1] Pose is the position and orientation in 3D space.

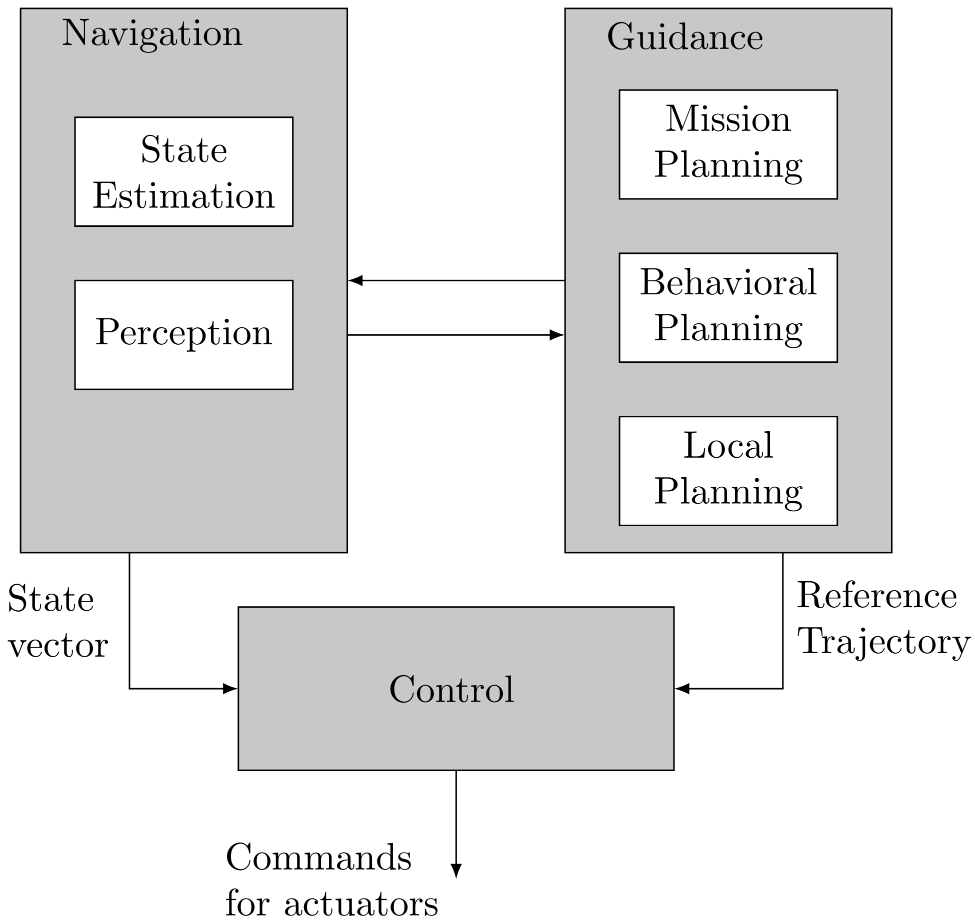


Figure 1.1: The Key modules of UAV autonomy: guidance, navigation, and control (Guidance) systems

platform's states (pose, speed, sensor biases). Additionally, a reliable pose estimation module is necessary to rectify human perspective errors and maintain correct pose in beyond visual line of sight (BVLOS) operations. Therefore, reliable navigation becomes a fundamental requirement for all levels of autonomy of UAVs. Additionally, when operating in dynamic environments, UAVs need to be integrated with obstacle detection and landing zone detection capabilities to ensure the safety of humans and infrastructure. To this end, UAVs require to have the ability to build an accurate map of the traversed environment.

Among UAV navigation solutions, Global positioning system (GPS) and inertial measurement unit (IMU) aided inertial navigation systems (INS) are at the forefront [21–24]. These systems use inertial data (accelerometer and gyroscope) and GPS pseudo-range measurements to estimate the platform states [25]. Additionally, they may use magnetometers for heading estimation. However, GPS-based methods can result in faulty navigation solutions, especially in urban environments, due to signal obstructions and multipath errors [26, 27]. A study assessing the risk of UAVs reported that the loss of GPS signals contributes to 12%-17% of UAV crashes, posing a significant safety risk for UAV-based delivery [28].

This risk can be reduced by aiding GPS navigation with GPS denied navigation solutions [29, 30], including ultra-wideband (UWB) positioning-based [31–33], radar-based [34–36], vision-based [37–39], lidar-based [4, 40], and multi-sensory combined navigation solutions [2, 41]. UWB positioning systems are commonly used for indoor navigation and can achieve centimeter-level accuracies [31]. These systems require physical anchors to be installed within the UAV navigation space, making it an applicable solution for zones where infrastructure support can be added. The long-range capability of radio sensors and the robustness of radar waves to weather conditions such as mist, rain, and fog make radar-based navigation solutions suited for outdoor navigation [34]. However, these solutions are at the developing stage and require other additional sensors such as camera and lidar to overcome the challenges of low accuracy, low resolution, and delayed sensor response [42].

Vision-based and lidar-based navigation solutions are emerging as GPS denied UAV navigation methods due to their map building, obstacles avoidance, and improved pose estimation capabilities [43]. The challenge of vision-based navigation solutions is that they are ineffective in low light conditions and environments with a low number of visually distinct features [2]. Lidar-based navigation solutions tend to fail in structureless environments and during aggressive motions [2]. Multi-sensory combined methods have managed to overcome these challenges by developing a visual, lidar, and INS integrated navigation solution [2,44]. It is robust to aggressive motions and can handle sensor degradation due to low light, featureless and structureless environments. One such realization is the visual lidar odometry, and mapping (VLOAM) system, which provides a multi-sensory robust solution for the GPS denied navigation problem [2,45]. This visual-lidar combined solution demonstrates the second-highest performance among navigation methods for benchmark GPS denied navigation challenge, KITTI odometry benchmark [5]. However, the source code of the VLOAM architecture is not publicly available and it does not incorporate GPS in the pipeline. Several research work focus on combining GPS/INS navigation with GPS denied navigation methods. Work in [46] and [29] present fusing GPS/INS with vision sensors and [47] presents fusing GPS/INS with lidar. Work in [48] proposes a general framework to integrate local and global sensors. Nonetheless, they have only presented results for the visual-inertial navigation system (VINS) and GPS integration. In this work, we propose a robust UAV navigation system by combining GPS with visual, inertial and lidar sensor information addressing the main drawbacks of the state of the art systems discussed in section 1.2.

1.2 Problem Statement

This study presents a novel GPS-aided visual lidar combined navigation method for end-to-end UAV-based delivery applications by addressing the following key challenges.

1.2.1 Problem I: GPS errors and mapping capability

GPS can provide precise location information. However, the accuracy of GPS positioning can be degraded by the multipath effect, electromagnetic interference, spoofing and other interruptions [26, 27]. Especially in urban environments, the multipath effect severely reduces the location accuracy of GPS. Moreover, GPS-based methods are limited to pose information and they do not provide information about the surrounding area such as a 3D map of the environment. Therefore, GPS-based methods alone are not sufficient for ensuring safe navigation. This thesis provides a visual and lidar combined navigation method that can build an accurate map of the environment and also assist with the navigation when GPS signals are erroneous or unavailable.

1.2.2 Problem II: Inherent drift in GPS denied navigation solutions

GPS denied navigation solutions that only rely on local sensors suffer from position drift accumulated over time. The best-reported accuracy of VLOAM has a drift of 0.22% of the distance traveled [2] and actual values of this drift can vary significantly based on sensor and calibration errors. Therefore, if used alone, a UAV with VLOAM navigation will only be permitted to fly under 5 km distances as the position drift error can exceed the safety regulation limits beyond the 5 km range. Consequently, GPS denied navigation methods could not be used alone for end-to-end delivery operations and is suitable for locally navigating an environment avoiding any obstacles within the local map. Therefore, this study proposes a visual and lidar combined method aided with GPS to correct the drift error when GPS signals are available.

1.2.3 Problem III: Unavailability of public packages for state-of-the-art robust UAV VLOAM navigation algorithms

Work in [2] and [49] are at the forefront of the multi-sensory UAV navigation architectures in terms of robustness and accuracy. Their robustness is determined by the ability to handle sensor degradation and aggressive motion that can cause sensor failures. The camera is sensitive to lighting changes and may fail in low light or featureless environments or when significant motion blur is present. The laser cannot handle structureless environments. However, navigation methods presented in [2] and [49] have enabled different combinations of sensors by having a modular architecture to bypass the failed sensor module, making them adaptable to different environments and motion. Even though these algorithms claim good performance, the modules are not publicly available for use in testing. Given the complexity of robotic navigation algorithms, it is common practice for navigation system researchers to publish their code on Github or similar. This allows unbiased evaluation of the methods with new datasets and compare with new algorithms. However, the work of [2] and [49] are commercially protected and not made available. Moreover, architecture in [2] uses loosely-coupled IMU, and it does not utilize modern efficient optimization libraries such as Ceres [50], GTSAM [51] for their implementation. There are robust navigation methods that use either vision or lidar sensors that work with many datasets. These include ALOAM for lidar only navigation [4], LIO-sam for lidar-inertial navigation [52], VINS-mono for visual-inertial navigation [37], and VINS-fusion for visual-inertial GPS fusion [48]. However, in these publicly available methods, the combination of all visual, lidar, imu, and GPS is not incorporated to be applicable to custom indoor/outdoor datasets. Therefore, a robust multi-sensory architecture needs to be developed for UAV navigation.

1.2.4 Problem IV: GPS is not incorporated with robust multi-sensory UAV navigation architectures

To the best of authors knowledge, work in [6] is the only implementation at the time of writing that has incorporated GPS with visual, lidar and inertial sensor information for UAV navigation and this architecture is publicly available for evaluation [github.com/LVI-SAM]. However, this implementation runs into robustness issues and does not properly handle dynamic conditions. Even though, work in [48] proposes a general framework to integrate local and global sensors, the results are only presented for the visual-inertial and GPS integration. Therefore, GPS aided robust multi-sensory UAV navigation with visual, lidar and inertial sensors need to be developed.

1.3 Objective and Expected Contributions of the Research

This thesis proposes a novel UAV navigation architecture to estimate the UAVs' pose (3D position and 3D orientation) and build a map of the traversed environment. The main objectives of this thesis can be outlined as follows:

Objective 1 Propose a novel robust visual, lidar and inertial integrated odometry and mapping system for UAV navigation which includes following features.

- Implementing more accurate tighly-coupled IMU pretintegration.
- Computationally efficient modern optimization libraries are used.
- Combining features of the existing VINS-mono and ALOAM packages to improve the robustness to sensor degradation.

Objective 2 Propose a novel GPS aided multi-sensory navigation system with GPS outlier rejection.

Objective 3 Experimentally validate and compare the performance of the proposed method with state-of-the-art UAV navigation methods [4, 25, 37].

- Experimental validation of the odometry and mapping accuracy of the proposed method for different datasets, including KITTI benchmark dataset [5] and LVI-SAM dataset [6].
- Compare odometry estimation accuracy of the proposed method with state-of-the-art navigation methods such as VINS-mono [37], LOAM [4] and GPS/INS [25].
- Performance evaluation of the proposed method for different GPS denied scenarios and validate the regulatory compliance. .

1.4 Organization of the Thesis

Chapter 1 presents an overview of the research area, highlights the research statement, and outlines the objectives and associated contributions of this study.

Chapter 2 presents the literature review in the area of UAV navigation methods and highlights the limitations of the existing systems.

Chapter 3 presents the results of the numerical study to evaluate the suitability of state-of-the-art navigation systems and proposed system for UAV-based parcel delivery.

Chapter 4 presents the novel visual lidar combined navigation pipeline and its results for different datasets with a comparison with the state-of-the-art UAV navigation methods. This implementation is referred to as Visual inertial lidar odometry and mapping (VI-LOAM) version 1.1,

Chapter 5 presents the improvements carried out to develop VI-LOAM version 1.2 from the previous version. Also, this chapter discusses the methodology for incorporating GPS with the VI-LOAM navigation pipeline and presents comparison results for different GPS scenarios.

Chapter 6 presents the conclusion and directions for future studies.

Chapter 2

Background

2.1 UAV Safety Regulations

To ensure safety and compliance in air transport, many international, national, and local governing bodies around the world regulate UAV operations [18]. The International Civil Aviation Organization (ICAO) [53] is a specialized agency under United Nations that regulates international air navigation. The Federal Aviation Administration (FAA) [54] of the USA and Transport Canada [55] in Canada regulate air transportation in North America. These regulations address a wide variety of issues, including airspace control, navigation, remote pilot licensing, UAV registration, privacy, data security, and public safety. Among these, the regulations that are directly related to the navigation system of UAVs are discussed below.

"*Standard 922, Remotely Piloted Aircraft Systems (RPAS) Safety Assurance - Canadian Aviation Regulations (CARs)*" of transport Canada describes the technical requirements of a UAV for compliance [55]. According to section 04 of standard 922, the remotely piloted aircraft system must have a lateral position accuracy of at least $\pm 10\,\mathrm{m}$ and altitude accuracy of at least $\pm 16\,\mathrm{m}$ while operating within a controlled airspace[1]. The UAV operations should ensure that this accuracy can be maintained, accounting for possible degraded sensory modes of operation and within the entire operational space. Further, for operations near people, section 05 of standard 922 states that the occurrence of any single failure of the RPAS, which may result in a severe injury to a person on the ground within 30 m, must be very unlikely.

Table 2.1 summarizes Transport Canada's safety assurance accuracy requirements for RPASs. In subsequent sections, this thesis discusses each navigation solution in the context of regulatory compliance.

Table 2.1: Transport Canada RPAS Safety Assurance Accuracy Requirements

State	Required Accuracy	Evaluation Standard
Lateral position	$\pm 10\,\mathrm{m}$	Absolute Error
Altitude	$\pm 16\,\mathrm{m}$	Absolute Error
Human injury alert range	$30\,\mathrm{m}$	Absolute horizontal distance

Additionally, the UAVs' GNSS navigation systems should be able to overcome the following erroneous scenarios.

Terrain errors: terrain masking of the signal, for example, by a building or mountain, blocks the antenna on the RPAS from receiving the satellite signal or create multipath components of the signal.

Atmospheric errors: errors caused from the refraction of GNSS radio signals by the ionosphere and the troposphere.

Satellite errors: errors resulting from poor or unexpected geometries related to the positions of the GNSS satellites in reference to an RPAS due to reasons such as gravitational effects of the Sun and Moon may pull the satellites from the planned orbital path.

[1]Controlled airspace is airspace of defined dimensions within which air traffic control services are provided

Geometric dilution of precision (DOP): errors occur when there is no adequate cross cut in the "fix" ,i.e., all satellites are too closely located to one other.

2.1.1 Performance Evaluation

Ensuring the safety of people and infrastructure is paramount for UAV-based delivery. According to the safety regulations presented in section 2.1, position accuracy is the only performance indicator for ensuring safe navigation. Moreover, reaching the correct destination, which is directly related to the position estimation accuracy, is also one of the top priorities for delivery applications. Therefore, while other performance metrics such as energy efficiency, computation power, and delivery times can be used to evaluate UAV navigation methods, this thesis mainly focuses on position accuracy. In the literature, mainly two performance metrics are used to evaluate position accuracy: root-mean-square error (RMSE) and percentage position drift (relative position drift) of the distance traveled [56].

2.1.1.1 Root-Mean-Square Error (RMSE)

RMSE position error is the square root of the average of squared differences between actual path coordinate values and coordinate values from the estimated trajectory. Equation 2.1 computes the RMSE position error at time t of the trajectory.

$$RMSE_t = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2)} \tag{2.1}$$

N is the number of samples taken up to time t. Δx_i, Δy_i, and Δz_i are the X, Y, and Z direction absolute position errors between the actual and estimated trajectories at each sample time.

2.1.1.2 Percentage Position Drift

Percentage position drift is the position difference between actual and estimated trajectories as a percentage of distance traveled along the trajectory to the calculated point. The percentage position drift is computed by Equation 2.2.

$$\text{Percentage position drift} = \frac{\text{Position difference}}{\text{Distance along the trajectory}} \times 100\% \tag{2.2}$$

Where, Position difference $= \sqrt{\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2}$

2.2 UAV Navigation Methods

UAV navigation solutions are divided into four main categories based on the sensors used. Satellite-based navigation [57–59], inertial navigation [60, 61], vision-based navigation [62–64], and light detection and ranging (lidar) sensor-based navigation [4, 65, 66].

Satellite-based navigation with global coverage uses GNSS receivers onboard a UAV and calculates the pose using the timing information received from orbiting satellites [57]. Inertial navigation uses IMU with sensors such as gyroscopes, accelerometers, and magnetometer sensors to estimate UAV's pose [67]. IMU provides data on linear acceleration and angular speed along three axes which can be integrated over time to estimate the pose information relative to the starting position. It is important to note that this solution drifts from the true values due to the accumulation of errors present in the sensor reading. The degree of drift is specified for the IMU class, where fiber-optic navigation grade IMUs provide the highest accuracy [68]. Vision-based navigation systems use vision sensors such as stereo cameras, RGB cameras to capture images of the surroundings [38]. Then the motion of the images is estimated by comparing the successive images. This estimated motion of the images or image features can be used to estimate the trajectory of the UAV. Similarly, in lidar-based navigation, lidar sensors capture point clouds of the surrounding environment [66]. Then spatial feature points such as edge points, planar points of point clouds are detected and tracked to estimate the trajectory of the UAV. Each of these systems has its own merits and demerits. Vision-based navigation solutions are low cost and less susceptible to aggressive motion errors, whereas low lighting may result in erroneous results. Lidar-based navigation solutions are highly accurate and immune to lighting changes. However, aggressive motion and structureless environments can affect them. Therefore, available UAV navigation systems use the above sensor technologies either stand-alone or in combination, depending on the application.

To realize these sensor combinations, different sensor fusion approaches are used. This study mainly considers three sensor fusion approaches: filter-based, optimization-based, and artificial intelligence (AI)-based-sensor fusion architectures. Moreover, sensor fusion can be further categorized into loosely-coupled and tightly-coupled methods based on the type of measurement data used [69]. For a given application, there are various factors, including the accuracy requirement and computational complexity, that govern the selection of sensor fusion mechanisms.

2.3 Types of Sensor Combinations

2.3.1 GNSS-based Navigation Methods

GNSS-based navigation methods work well in the outdoor clear sky-view environment. However, when it comes to urban environments, it encounters errors due to multi-path propagation. On the other hand, INS suffers from accumulated errors from the IMU measurements. Consequently, GNSS and IMU measurement are fused to develop GNSS/INS combined navigation systems [25, 58, 71]. However, when GNSS signals are erroneous, GNSS/INS systems only rely on INS and suffer from large accumulation errors.

2.3.2 Vision-based Navigation Methods

Vision-only navigation systems are extensively studied, and most notable methods are based on simultaneous localization and mapping (SLAM) [72]. To this end, image features, such as corners, edges or blobs, and pixel intensities are used to track the camera path and to build a map of the environment. The elementary vision-based sensor solutions are solely based on vision sensors. Among them, semi-direct monocular visual odometry (SVO) [62] has recorded the highest precision i.e., 1% position drift of the distance traveled. Other popular visual SLAM methods include large-scale direct monocular-SLAM (LSD-SLAM) [73], oriented fast and rotated brief-SLAM (ORB-SLAM) [63], direct sparse odometry (DSO) [74] and stereo odometry algorithm relying on feature tracking-SLAM (SOFT-SLAM) [64]. The most common vision-based navigation system is visual-inertial navigation, where vision sensors and IMU are integrated [37, 75–80]. Data from vision sensors and GPS can be combined to obtain improved state estimations [29, 46, 81–83]. Work presented in [48] and [84] propose navigation solution combining camera, IMU and GPS. They have managed to reduce the position error under 1 m (RMSE). However, to achieve this accuracy, readings from all the sensors must be integrated. Typically, vision-based navigation systems face challenges caused by camera sensor degradation due to the lack of features in captured images and poor lighting conditions. Further, for UAV-based delivery applications where long distances are traveled, the position estimate will drift considerably without loop closure, in-between GPS updates [85].

2.3.3 Lidar-based Navigation Methods

Lidar only navigation systems have been successfully implemented to estimate the motion of UAVs as lidar can provide accurate range measurements to the objects in space [4,65,66,86]. Many studies integrate lidar with additional sensors to mitigate the drift error accumulation. For instance, work presented in [87] and [88] integrated lidar with GPS, [89] and [90] integrated lidar with IMU, and [47, 91–93] integrated lidar with GPS and IMU. The lidar, GPS, and IMU combined system proposed in [47] has significantly improved the navigation accuracy and it only has a RMS position error of 1.1 m. However, lidar-based navigation systems are unable to handle aggressive motion due to their low-frequency update rate [2] and are prone to failures in structureless environments due to lack of features.

2.3.4 Vision, lidar Combined Navigation Methods

To mitigate the limitations of stand-alone vision and lidar sensors and achieve better accuracy in navigation, vision and lidar-based systems are fused together [94, 95]. A filter-based lidar, IMU, and camera fusion navigation system, namely LIC-Fusion, is proposed in [44, 96]. [2] proposes a highly re-configurable optimization-based pipeline that can handle lidar or camera sensor degradation scenarios. This method is known as VLOAM and has the highest accuracy, i.e, 0.22% position error of the distance traveled, among the multi-sensory navigation systems used in GPS denied environments. Lidar and vision sensors with both GPS and IMU sensors are used in a specific application of river mapping and navigation of a micro-helicopter presented in [97]. To this end, motion estimation is done using visual, inertial, and sparse GPS. The lidar measurements are only used for obstacle detection and map building. Accordingly, vision and lidar combined methods show better results for navigation and map building among the GPS denied navigation systems [43].

2.4 Sensor Fusion Approaches

In a multi-sensory system, each of the sensors has its advantages and disadvantages. Sensor fusion allows to aggregate information from different sensors together and obtains more accurate navigation as presented in section 2.3. The performance of the same sensor combination can vary depending on the sensor fusion approach. The available sensor fusion approaches can be categorized into three groups:

1. Filter-based approaches,
2. Optimization-based approaches, and
3. Artificial intelligence (AI)-based/ learning-based approaches

2.4.1 Filter-based Approaches

Filter-based approaches follow a recursive probabilistic formulation that uses a motion model and a measurement model [44, 96]. Using the motion model, the filter-based approaches predict the current pose of the UAV along with the associated uncertainty. Then the current measurements are predicted using the measurement models, which are compared with the actual measurements to update the predicted pose and the associated uncertainty calculation. Filter-based approaches typically solves the navigation problem using variants of Kalman filters including extended Kalman filter (EKF) [98], unscented Kalman Filter (UKF) [99] or particle filter [100]. To this end, they iterate at one time-step or iterate over several time-steps but do not have to consider measurements over the entire trajectory. This enables them to have a lower computation complexity and operate in real-time. However, as the entire trajectory is not considered, typically, the navigation accuracy of the filter-based approaches is lower than its counterpart, optimization-based approaches.

2.4.2 Optimization-based Approaches

Optimization-based approaches formulate navigation as a constrained non-linear optimization problem. These approaches maintain the past state information and solve a full trajectory optimization or a sliding window with a subset of states, making it more accurate. However, solving a large dimensional non-linear optimization problem at each time step is computationally expensive. Therefore, in the past, using optimization-based approaches for real-time navigation was problematic [101]. However, with the improvement of computation power of onboard hardware and optimization techniques which updates only a subset of variables, optimization-based approaches are now used as online state estimators [2, 80, 82].

2.4.3 AI-based Approaches

AI-based approaches use AI techniques to model and train a system for safe and accurate navigation using sensor data under various conditions [102–105]. Work presented in [106, 107] introduce intelligent navigation using fuzzy logic and neural networks. A common learning-based approach experimented by many research works is using deep neural network (DNN) for navigation [105,108,109]. This is due to DNNs having higher degrees of freedom to represent data and provide better results than shallow neural networks [110, 111]. However, DNNs' requirement of large data sets makes it time-consuming to train a network for navigation. Deep reinforcement learning (deep-RL), which is an adaptive system that learns from real-world experiences, has also been used for UAV navigation [112]. To this end, a reward function for navigation is evaluated from a trial and error approach based on the UAVs' navigation decisions.

The aforementioned learning-based sensor fusion approaches exhibit better performance in modeling the measurement noise and providing more accurate estimates. Hence, learning-based systems are expected to have an improvement in accuracy and reduced processing time after sufficient training. In addition to state estimation, AI-based methods are used in navigation pipeline for tasks such as semantic segmentation and mode identification [113].

Apart from the classification summarized above, sensor fusion approaches can be classified into two groups based on the type of data and the inter-dependencies between them. These two groups are referred to as loosely-coupled and tightly-coupled systems [69]. Loosely-coupled systems treat outputs of each sensor as an independent information and sequentially combine them to estimate UAV states [114, 115]. In contrast, tightly-coupled systems use raw measurements of sensors and consider the correlation between measurements [116–118]. The next chapter discusses tightly-coupled and loosely-coupled architectures in detail. For learning-based methods, this tight coupling of measurement is achieved using end-to-end deep learning architectures. Although tightly-coupled implementations require better synchronization, calibration between sensor sources, and a marginal increase in computational requirements, they generate more accurate estimation compared to the loosely-coupled systems. Since safe navigation is key to the success of UAV-based delivery and with the recent improvement in onboard processing and memory capabilities of UAVs, tightly-coupled optimization-based approaches are preferred for UAV's state estimation.

Table 2.2: Type of sensor combinations and their performance

Combination Experiment	Sensor Fusion Approach	Reference	Estimation Position Error (Best method)	Orientation Error
GPS & IMU	Filter-Based : Unscented Kalman Filter (UKF)	[25]	1.12 m (RMSE)	0.2196° (RMSE)
lidar, IMU & GPS	Optimization-based sliding window/Pose Graph Optimization	[47]	1.1 m (RMSE)	0.166° (RMSE)
Camera, IMU & GPS	Optimization-based sliding window/Pose Graph Optimization	[48]	0.40 m (RMSE)	-
Camera only	Graph optimization (SOFT-SLAM, stereo camera)	[64]	0.65% of the distance travelled	0.0014°/m
Lidar only	Nonlinear Optimization (LOAM)	[4]	0.88% of the distance travelled	-
Camera & Lidar	Nonlinear Optimization (DEMO)	[41]	1.14% of the distance travelled	0.0049°/m
KITTI Dataset				
Camera & IMU	Nonlinear optimization/Pose Graph Optimization(VINS-Mono)	[70]	0.88% of the distance travelled	-
Lidar & IMU	Nonlinear Optimization (LOAM with IMU)	[2, 4]	0.39% of the distance travelled	0.0013°/m
Lidar, Camera & IMU	Nonlinear Optimization (VLOAM pipeline)	[2]	0.22% of the distance travelled	0.0013°/m

Chapter 3

Numerical Study of Navigation Methods for UAV-based Parcel Delivery

3.1 Introduction

This chapter presents the preliminary study conducted to evaluate the suitability of the proposed method and compare it with state-of-the-art navigation systems. As mentioned earlier, the regulatory compliance of existing navigation solutions is challenged during GPS loss, multipath signals, spoofing events, and other sensor degradation scenarios. This study investigates the suitability of GPS-aided VLOAM navigation system for UAV delivery applications. To this end, a simulation and numerical evaluation were carried out, confirming that the state-of-the-art multi-sensory navigation solutions violate the UAV navigation regulations, while the proposed combined GPS/VLOAM system comply with the regulations.

3.2 GPS/INS Based Navigation Methods

The most widely used sensor technologies for UAV navigation are GPS receivers and IMUs with accelerometers, gyroscopes, and on occasion, magnetometers [46, 58, 119, 120]. GPS can provide precise location information. However, the accuracy of GPS positioning can be degraded by the multipath effect, electromagnetic interference, and other interruptions [26, 27]. Especially in urban environments, the multipath effect severely reduces the location accuracy of GPS. Moreover, GPS can only provide 3D position measurements without 3D orientation. Therefore, GPS sensors alone are not sufficient for 6-DoF (Degrees of Freedom) state estimation.

IMUs can provide accurate motion information for a brief time period by integrating the acceleration and angular velocity measurements gathered by the sensor package. However, the accumulation of the error in IMU measurements over a period of time causes significant inaccuracies in position estimation [121]. Consequently, the combination of GPS and IMU complements the limitations of both systems where IMU addresses the intermittent availability of GPS data.

GPS/INS integration has been extensively studied, and many different research work exists for the integration [25, 102, 122, 123]. The integrating methods vary according to the application based on the complexity of the integration, cost, and accuracy requirement. The level of GPS/INS integration can be classified into two categories, namely,

1. loosely-coupled GPS/INS integration, and
2. tightly-coupled GPS/INS integration.

GPS derived position and velocity measurements are used in the loosely coupled integration [115], as shown in Figure 3.1. A tightly coupled GPS/INS navigation, as shown in Figure 3.2, integrates GPS pseudorange measurements and inertial measurements for motion estimates [116, 117, 124–128]. The pseudorange measurement is the difference between the time of reception and the time of transmission of a GPS satellite signal. The Mechanization equations are set of equations used to convert the acceleration and angular velocity measurements obtained from an IMU into position, velocity and attitude information.

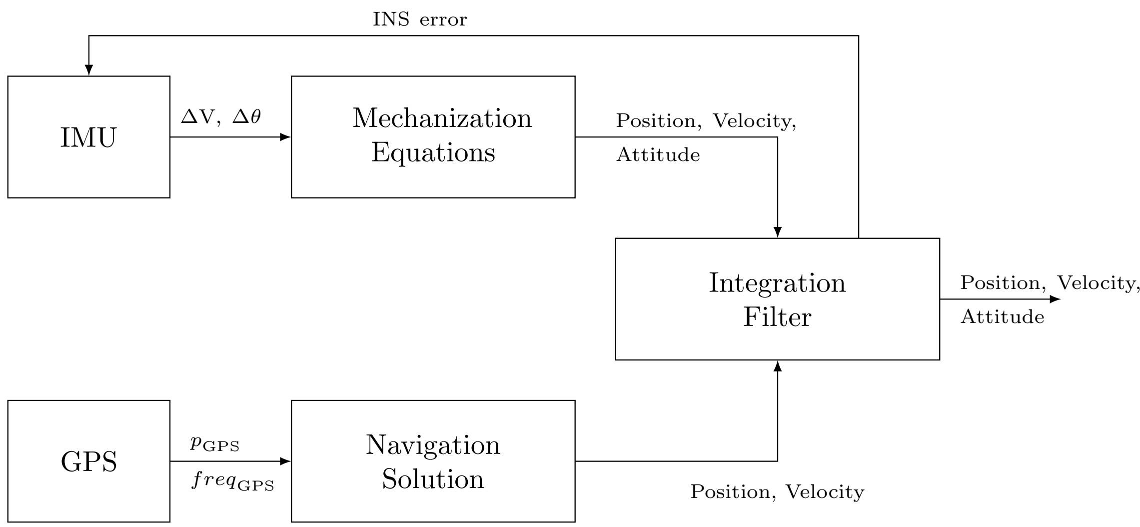


Figure 3.1: Loosely-coupled GPS/INS integration [1]

The combined GPS/INS system can achieve centimeter-level position accuracy when differential GPS (DGPS) is used, and accuracy in the order of several meters otherwise [123]. DGPS requires a ground-based reference station at a known GPS location, and the accuracy of a DGPS system degrades at an approximate rate of 0.22 m for each 100 km distance from the broadcast site [129]. Generally, the more complex the receiver design and the more expensive the GPS device, the higher the accuracy. A typical GPS/INS system has a position accuracy of approximately 1.12 m [25]. Therefore, it is safe to assume an accuracy within 1-2 meters for long-distance UAV-based delivery, and it is within the regulations of Transport Canada given in Table

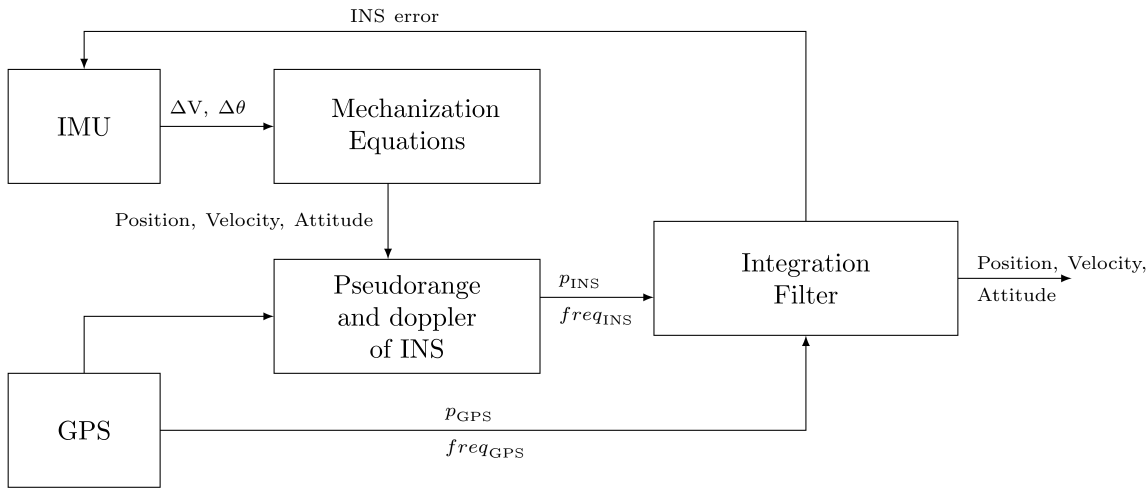


Figure 3.2: Tightly-coupled GPS/INS integration [1]

2.1.

During GPS loss, GPS/INS navigation systems solely rely on onboard IMU sensor, causing noticeable drift in the position estimation. For instance, using a MEMS tactical-grade IMU, C-MIGITS (BEI), a 3 minute GPS loss can cause a position error of approximately 20 m [26]. Consequently, relying only on INS in the absence of GPS presents reliability issues, and the system may not be able to maintain the accuracy within the regulated values in Table 2.1. Additionally, it is impossible to create an obstacle map of the navigation space by utilizing the GPS/INS sensor data. As a result, it is challenging to implement a collision detection module and maintain a safe distance for stationary or moving objects, including humans, as required by Transport Canada regulations. A prior map of the environment can be added to avoid collisions with stationary objects and infrastructure. However, a prior map cannot be used to avoid dynamic obstacles and unmapped obstacles. Therefore, a collision-avoidance system with an additional sensor suite needs to be added to protect infrastructure and people from UAV interference and reduce potential harm.

3.3 GPS Aided VLOAM Navigation

VLOAM fuses camera, lidar, and IMU data to provide accurate navigation within a short distance but could lead to erroneous pose estimations due to unbounded drift. In contrast, the GPS/INS system provides drift-free navigation for long distances through intermittent GPS updates but suffers noticeable estimation drift within a short distance when GPS loss occurs. The unique complementary characteristics of VLOAM and GPS/INS can be integrated to address the limitations of each system, resulting in improved navigation solutions for UAVs.

3.3.1 VLOAM Navigation

VLOAM provides a data processing pipeline for online navigation and builds a map of the traversed environment, utilizing data from a 3D lidar scanner, a camera, and an IMU [2]. A complete VLOAM pipeline has achieved an accuracy of 0.22% of the distance traveled, which is the highest level of accuracy achieved using onboard sensor kit without external aids like GPS to the best of our knowledge at the time of writing. This implementation used an optimization-based method to estimate UAV's pose.

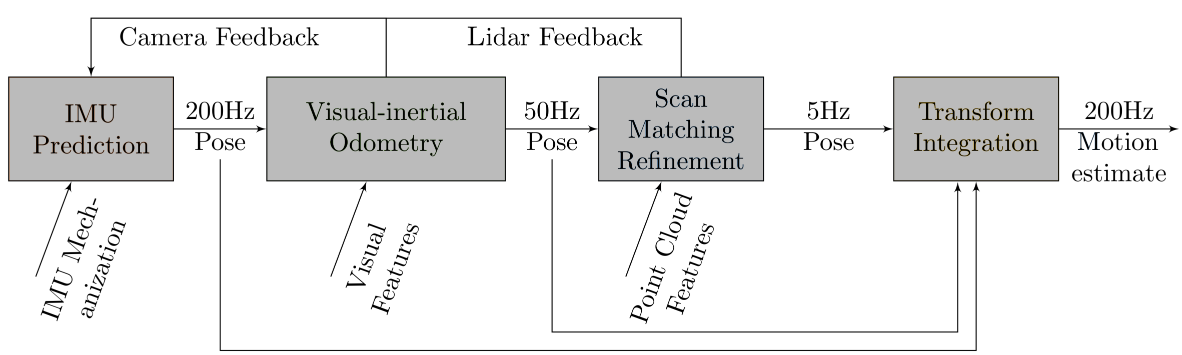


Figure 3.3: Overview of the VLOAM pipeline [2]

VLOAM is developed by integrating real-time depth enhanced monocular odometry [41] and lidar odometry and mapping in real-time (LOAM) [4]. Overview of the

VLOAM pipeline is shown in Figure 3.3. This pipeline begins with IMU mechanization for pose prediction, and then the visual-inertial combined method estimates the motion. Finally, a scan matching method further refines the motion estimate and builds a map of the traversed environment. The modules are arranged from left to right such that high-frequency modules at the beginning handle the aggressive motion, whereas low-frequency modules correct the drift from previous modules. Feedback from both the visual-inertial module and scan matching module are used for correcting the velocity drift and biases of the IMU.

Additionally, the modularized data processing pipeline enables the system to handle sensor degradation effectively. If the camera is futile due to poor lighting conditions or texture-less environments, or if the lidar is futile due to structure-less environments or weather conditions, the system can bypass the corresponding module and estimate the motion reliably using the rest of the pipeline.

IMU prediction subsystem of the VLOAM pipeline utilizes angular rates and acceleration in the camera frame as measurement inputs. This subsystem obtains a short-term prediction of the orientation by integrating gyro measurements. Then, with the help of calculated orientation, acceleration is integrated over time twice to obtain the translation. Next, the visual-inertial odometry (VIO) subsystem combines vision and IMU sensors. To this end, it uses pose constraints from IMU and camera to solve an optimization problem to estimate the incremental motion. Camera constraints are formulated by matching unique features observed across a sequence of images. Based on depth association, three types of features are used:

(a) features with depth associated from lidar range measurements,

(b) features with depth associated from triangulation, and

(c) features without depth.

VIO module is followed by the scan matching subsystem, which further refines the motion estimates received from previous modules. First, lidar points of the current lidar sweep are registered to a local point cloud using the previous odometry estimate. Then, edge and planar geometric features are detected from the point cloud and matched with the existing map. Lidar constraints are formulated to minimize the distance between detected features to the map features. Further, pose constraints from the previous VIO estimate are also used for solving the optimization problem. The map is then updated by merging the current point cloud at the end of the lidar scan sweep. Finally, the transform integration module integrates the motion estimates from three modules. Note that each module updates at different rates, the IMU prediction module runs at 200 Hz, the VIO module runs at 50 Hz, and the scan matching module runs at 5 Hz, to generate accurate high-frequency motion estimate. High accuracy of only 0.22% drift of the distance traveled of the system is useful for last-mile goods delivery in urban settings. Further, the increased reliability by compensating for sensor degradation and robustness to the aggressive motion of the VLOAM pipeline is accommodating for the UAV-based delivery application, which requires higher safety standards with the presence of humans and other structures. Moreover, the sensor suite of lidar and camera can be used to enhance safety by assisting obstacle detection and avoidance.

However, when we evaluate the system performance with the Transport Canada regulations, if used alone, a UAV with the VLOAM pipeline will only be permitted to fly under 5 km distances as there will be no loop closure in a straight path. Beyond 5 km, the position drift can be more than 10 m which violates the Transport Canada safety regulations. Moreover, the system might fail in weather conditions, for instance, foggy, rainy, or snow conditions, which can deteriorate lidar and vision sensor measurements. Therefore, VLOAM needs a global sensor that is not affected by weather conditions and also, which can correct the drift intermittently to keep the drift in check.

3.3.2 GPS and VLOAM Integration

For GPS/VLOAM integration, an optimization-based framework will perform better than filter-based methods as they are more accurate [130]. Also, filter-based methods are extremely challenging due to time synchronization and complexities associated with time-delayed measurement updates [130]. A general optimization-based framework to fuse local states with global sensors is proposed in [48]. To this end, they use a secondary pose graph optimization to fuse the local and global sensors as shown in Figure 3.4.

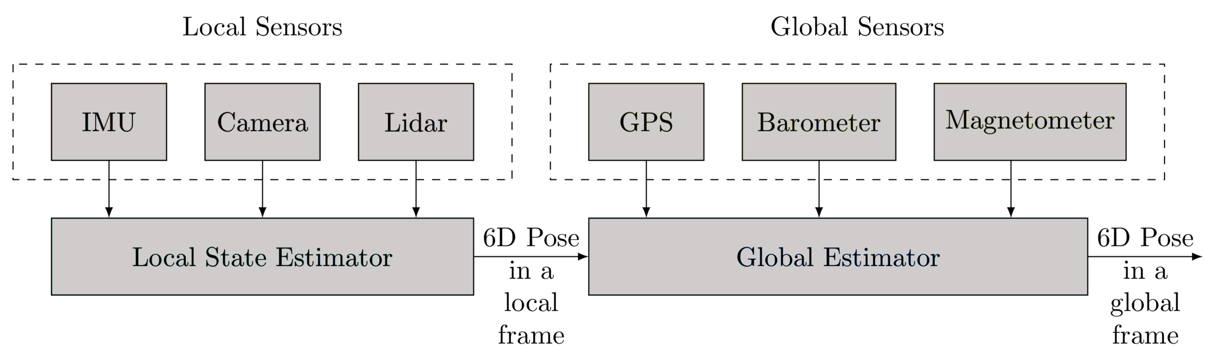


Figure 3.4: Overview of framework to fuse the local and global sensors

In this framework, sensors that are not globally referenced, such as camera, lidar, and IMU, are considered as local sensors. For navigation using local sensors, the initial pose of the UAV is taken as the origin, and UAV motion is incremented relative to that. These local sensor information is fused in local state estimator provide 6D pose relative to the origin. For pose graph optimization, any local sensor navigation framework which can provide 6-DoF poses can be used [48]. On the other hand, global sensors such as GPS, magnetometer, barometer provide globally referenced measurements and fused in global estimator. Global sensor measurements are considered general factors in pose graph optimization. The global pose graph structure of this work is shown in Figure 3.5.

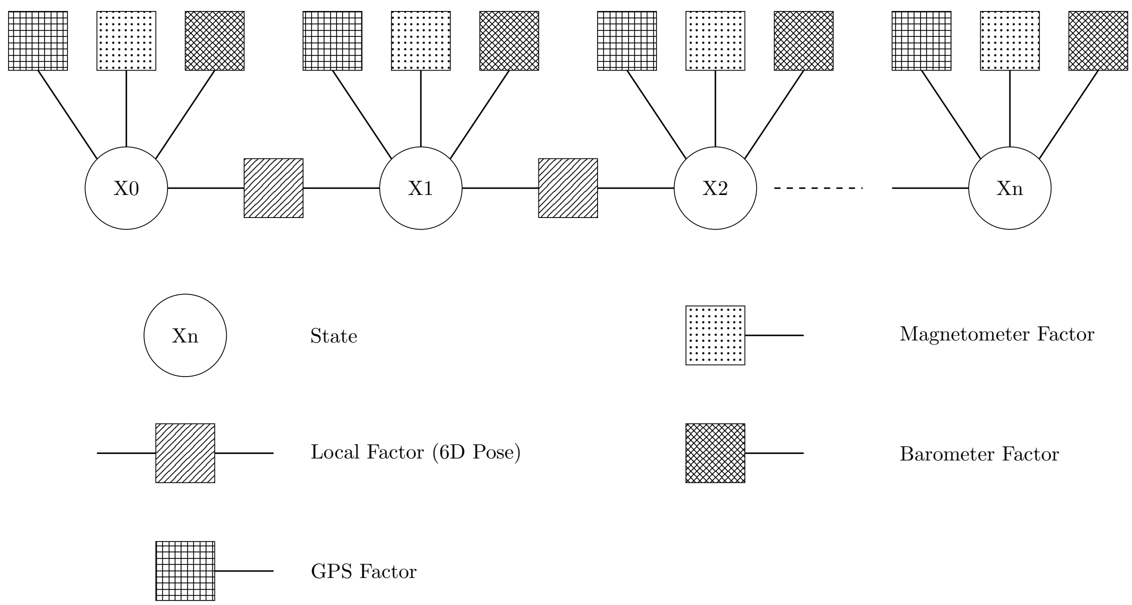


Figure 3.5: An illustration of the global pose graph structure with local and global factors

In the global pose graph, each node represents the position and orientation of the UAV in a globally referenced frame. Local factors, i.e., local constraints, are obtained by the relative pose between two frames of the local state estimator, connecting the consecutive nodes. Global measurements directly constrain the position of nodes, and they are mapped as edges in the pose graph optimization. Solving the graph involves finding the best configuration of nodes that matches all the edges to the fullest extent. To this end, the pose graph optimization is run at low frequency (1 Hz). Further, after every optimization, a transformation from local frame to global frame is estimated, enabling the real-time high-frequency global state estimation.
In their study, to obtain the experimental results, state-of-the-art VIO, namely, VINS-fusion is used as the local state estimator. GPS measurements are used as global

sensor input. Extending this work to use VLOAM as the local state estimator and GPS measurements as a global sensor can improve accuracy and reliability.

3.4 Evaluation of Navigation Methods

A simulation study was conducted using MATLAB to compare the performance of existing navigation systems with a combined GPS/VLOAM system. To this end, the 2-D trajectory of the UAV as shown in Figure 3.6 is considered. The trajectory has a total distance of 15 km, and a maximum velocity of 18 m/s which is the maximum velocity of DJI Matrice 600 UAV, is reached during the simulation. Moreover, simulations for combined sensor systems with GPS were carried out under different GPS conditions to compare the performance. This simulation study assumed that the UAV-based delivery operation is started in a space with proper GPS measurements in which multipath and other GPS errors are negligible. The sensor parameters for this simulation study were obtained from the literature and using the established datasets for UAV navigation. KITTI odometry dataset [5] has been used as the main source as it is the benchmark dataset used by state-of-the art UAV navigation methods [2, 4, 48, 131]. Moreover, to the best of authors knowledge, the KITTI dataset is the only publicly available dataset with camera, IMU, laser and GPS sensor data.

3.4.1 Sensor Parameters

The IMU sensor used in KITTI odometry dataset [5] is adopted as the IMU sensor for this simulation. To this end, OXTS RT 3003 GPS/IMU inertial navigation system is used, and IMU is simulated with gyro and accelerometer sensors with an update rate of 100 Hz. IMU parameters were obtained from VINS-fusion implementation for KITTI dataset [132] and these parameter values are given in Table 3.1.

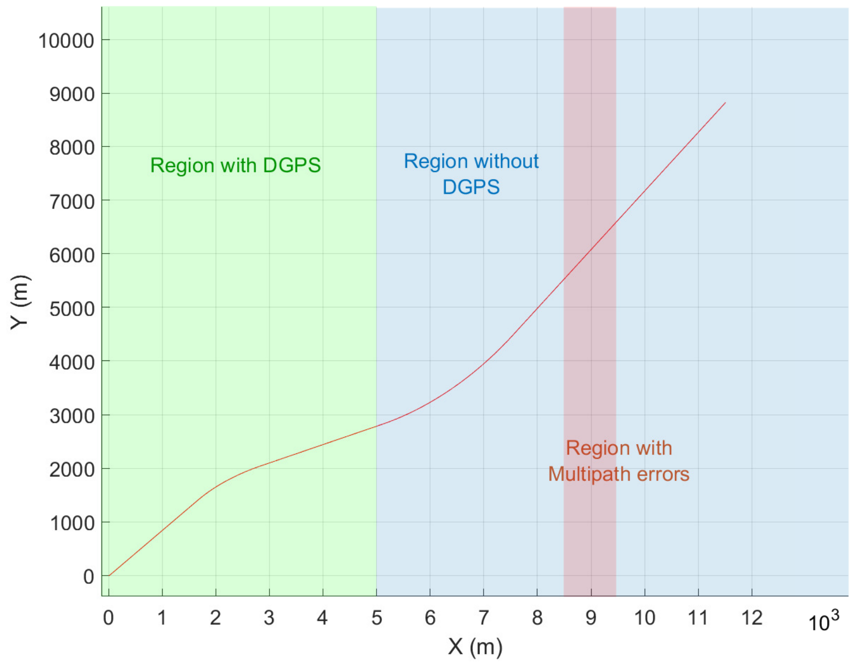


Figure 3.6: Actual Trajectory of the simulation

Table 3.1: IMU sensor parameters

Sensor Parameter	Standard deviation
Accelerometer Bias	0.001 m/s^2
Accelerometer Noise	0.1 m/s^2
Gyro Bias	0.0001 rad/s
Gyro Noise	0.01 rad/s

The VLOAM navigation system is simulated as a combination of an IMU and an odometry sensor to mimic the performance of the VLOAM system. Odometer provides velocity measurements at 50 Hz, and it is equivalent to lidar and camera correction. To maintain the consistency of the results, the same IMU parameters from the KITTI odometry dataset are used here. In general, the VLOAM system's percentage position error for the distance traveled starts with a 1.25% and ends with a 0.5% according to results submitted to KITTI odometry benchmark [5] (refer to Figure 3.7). Moreover, VLOAM system's rotation error for distance traveled starts with a 0.004°/m and ends with a 0.001°/m (refer to Figure 3.8). First, we simulated the VLOAM system as an IMU and an odometer for the 00 sequence of the KITTI odometry dataset and adjusted the standard deviation of gyro noise and standard deviation of velocity noise to match the aforementioned performance criteria. The simulation results for the KITTI 00 sequence are shown in Figure 3.9. Then, the simulation was carried out to the benchmark trajectory shown in Figure 3.6 using the tuned sensor parameters.

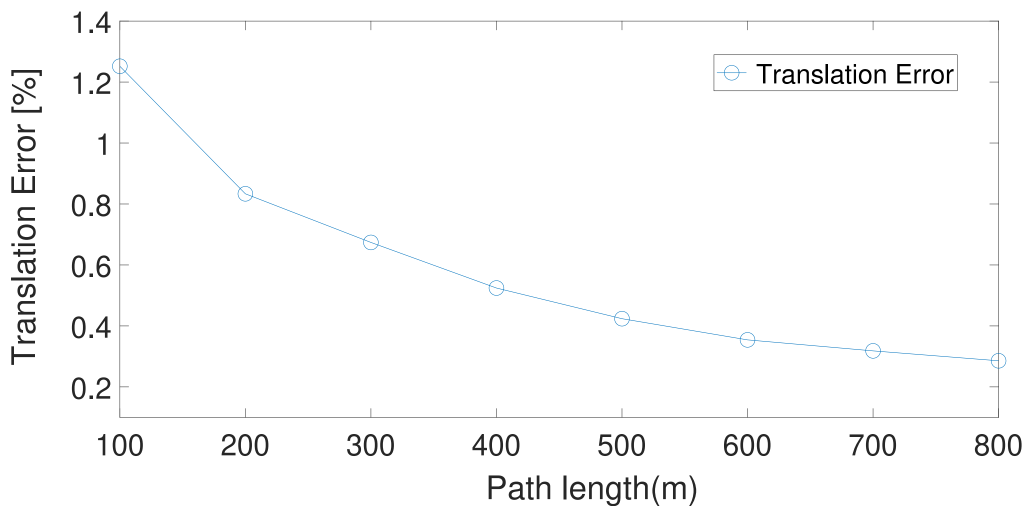


Figure 3.7: Percentage position error for the distance traveled of VLOAM system for sequence 13 of KITTI dataset [3]

In this simulation, GPS is simulated as a position sensor that provides 2-D position coordinates of the UAV at an update rate of 1 Hz. It was assumed that the GPS position noise distribution is Gaussian according to the central limit theorem [133] as there are various random noises that sum up the GPS position noise. Further, the simulation study was conducted under different GPS conditions, namely, GPS with DGPS correction, GPS without DGPS correction, and GPS with the presence of multipath errors. The first 5 km of the trajectory is considered as an area with DGPS (this corresponds to 0 s to 723 s time interval), and beyond that, GPS measurements are given without the DGPS correction (723 s to 1340 s time interval). Then, the area between 10 km and 11 km of the trajectory is considered as an area with multipath

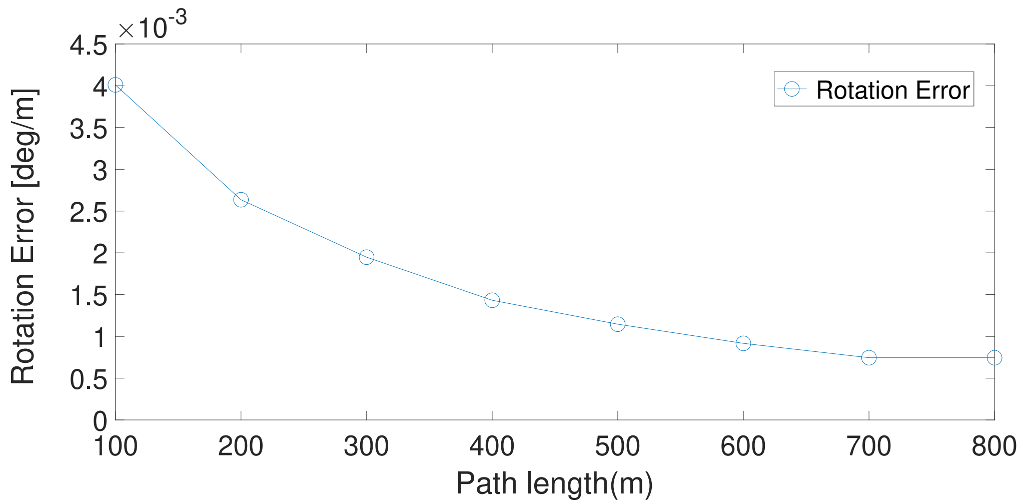


Figure 3.8: Rotation error for the distance traveled of VLOAM system for sequence 13 of KITTI dataset [3]

GPS signals (1061 s to 1118 s time interval). The parameters used for each GPS condition is acquired from literature. These parameters and other sensor parameters used for this simulation are summarized in Table 3.2.

Table 3.2: Sensor performance parameters

Combination	Reference	Position Error	Orientation Accuracy (deg/m)
IMU (100 Hz)	[134]	1.12 m (RMSE)	0.04
VLOAM (50 Hz)	[5]	1.25% of the distance traveled	0.004
GPS (1 Hz)			
• Accuracy with DGPS	[135]	3 cm	-
• Accuracy without DGPS	[25]	1.12 m	-
• Accuracy with the Presence of Multipath signals	[136]	15 m	-

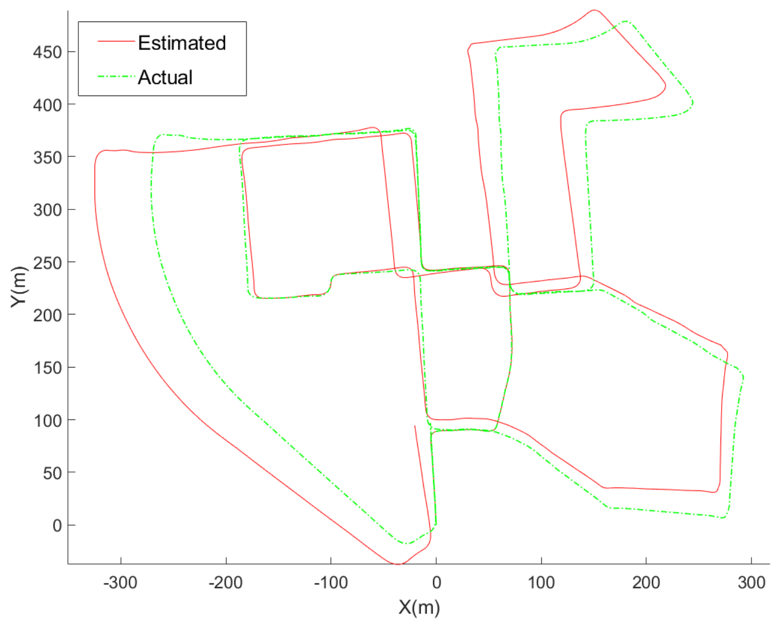


Figure 3.9: Results of the simulated VLOAM system for 00th sequence of KITTI dataset

3.4.2 Results

Figure 3.10 illustrates the absolute position error with respect to the time. As shown in the figure, the position error of both IMU only and VLOAM only navigation systems keeps increasing as a result of drift accumulation. It remains bound for the other two systems except for the area with GPS multipath signal errors. Comparatively, VLOAM drifts slower than the IMU drift. However, both the standalone VLOAM and IMU systems surpass the required safety accuracy level of 10 m position error.

In contrast, GPS/INS and GPS/VLOAM systems have managed to keep the absolute position error in centimeter-level with DGPS correction. In the absence of DGPS, GPS/INS system was able to maintain the estimation error under 2 m except for the area with multipath signals. The position error of GPS/INS exceeded the 10 m safe margin set by Transport Canada and reached 14.432 m when multipath signals were present. In contrast, even with the multipath signals, GPS/VLOAM system has kept the position error around 1 m. The maximum position error for different regions of the trajectory for each navigation system is summarized in Table 3.3. According to this simulation results, only GPS/VLOAM integrated solution manages to meet the required safety accuracy levels by maintaining the position error under 10 m throughout the entire trajectory.

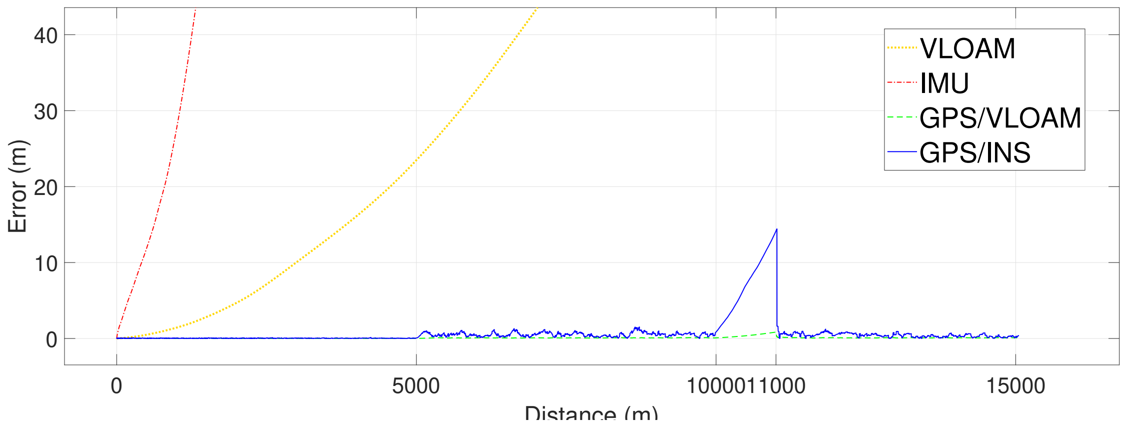


Figure 3.10: Absolute position error of IMU Only, VLOAM Only, GPS/INS and GPS/VLOAM with respect to time

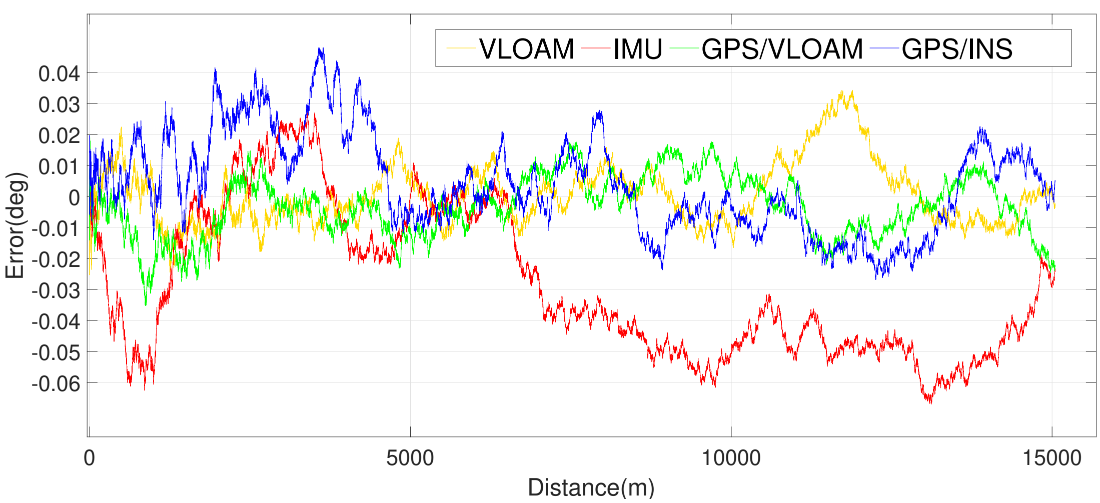


Figure 3.11: Orientation error of IMU Only, VLOAM Only, GPS/INS and GPS/VLOAM with respect to time

Figure 3.11 shows the orientation error of each system with respect to the distance.

Table 3.3: Maximum errors for each navigation solution

Method	Maximum Position Error			Maximum Orientation Error
	With DGPS	Without DGPS	With Multi-path errors	
IMU	426 m	2045 m	N/A	0.0669
VLOAM	23.52 m	158.70 m	N/A	0.0352
GPS/INS	0.125 m	1.519 m	14.432 m	0.0482
GPS/VLOAM	0.012 m	0.115 m	0.8406 m	0.0344

IMU only system has the highest orientation error, whereas GPS/VLOAM system has the lowest orientation error. Maximum orientation errors for each navigation system are summarized in Table 3.3. This simulation depicts that GPS/VLOAM navigation solution has superior performance in terms of both position and orientation accuracy.

Error states with their two standard deviations of the mean for each navigation solution are shown in Figure 3.12-3.15. Note that the Y-axis scale of each graph is different because the estimation error characteristics of each estimator are different. These results validate the consistency of proposed state estimator. As the error states are well within the two standard deviation bounds of the mean error, it is possible to confirm that the filter is working properly. Moreover, this indicates that GPS/VLOAM navigation solution does not violate the 10 m accuracy requirement in either of the X-Y directions.

3.4.3 Summary

This chapter evaluated the regulatory compliance of GPS/VLOAM integrated solution through a numerical simulation. The results demonstrated that the GPS/VLOAM integrated solution manages to solve the GPS degradation issues while adhering to recommended safety regulations. Therefore, this solution has a high potential to be used in long-distance UAV-based delivery. This chapter is only a numerical simulation to validate the proposed system. In next chapter, implementation of a novel VLOAM system is explained and Chapter 5 presents results with more exhuastive validation for the actual implementation of GPS incorporated VLOAM system.

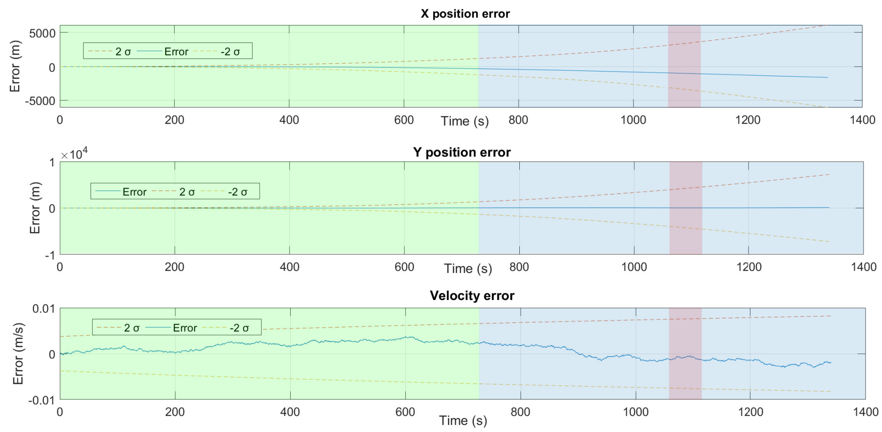


Figure 3.12: Error states with two standard deviations of the mean for IMU only navigation. Green: Region with DGPS (0s to 723s time interval or 0 to 5km), Blue: Region without DGPS correction (723s to 1340s time interval or 5km to 15km), Red: Region with multipath errors (1061s to 1118s time interval or 10km to 11km)

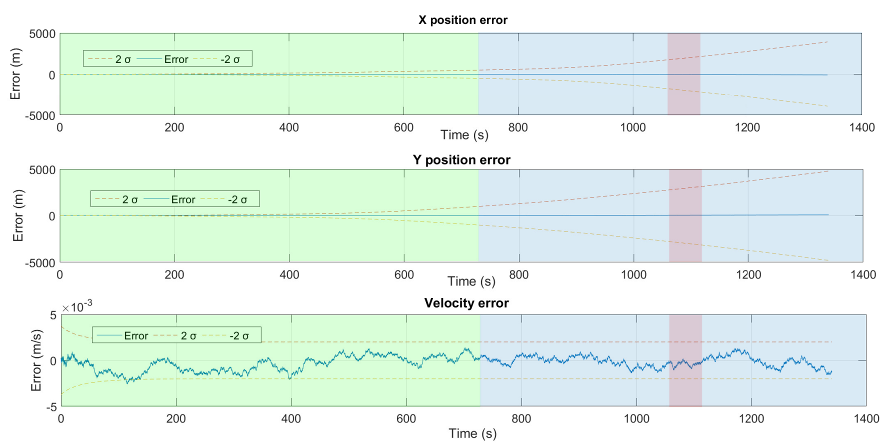


Figure 3.13: Error states with two standard deviations of the mean for VLOAM only navigation. Green: Region with DGPS (0s to 723s time interval or 0 to 5km), Blue: Region without DGPS correction (723s to 1340s time interval or 5km to 15km), Red: Region with multipath errors (1061s to 1118s time interval or 10km to 11km)

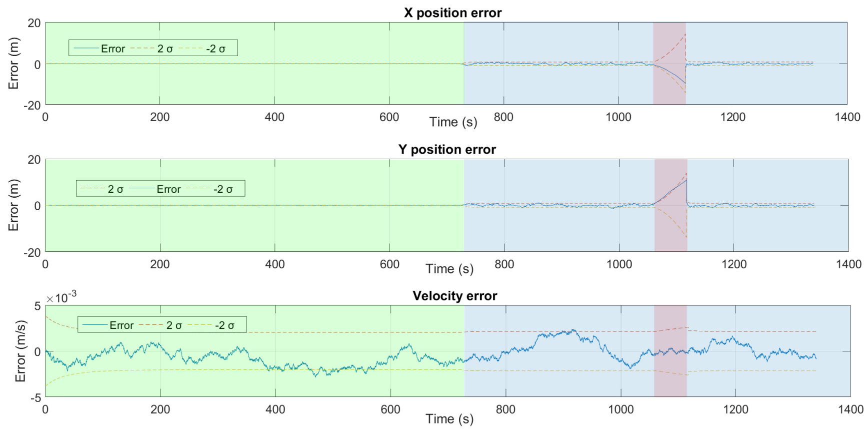


Figure 3.14: Error states with two standard deviations of the mean for GPS/INS navigation. Green: Region with DGPS (0s to 723s time interval or 0 to 5km), Blue: Region without DGPS correction (723s to 1340s time interval or 5km to 15km), Red: Region with multipath errors (1061s to 1118s time interval or 10km to 11km)

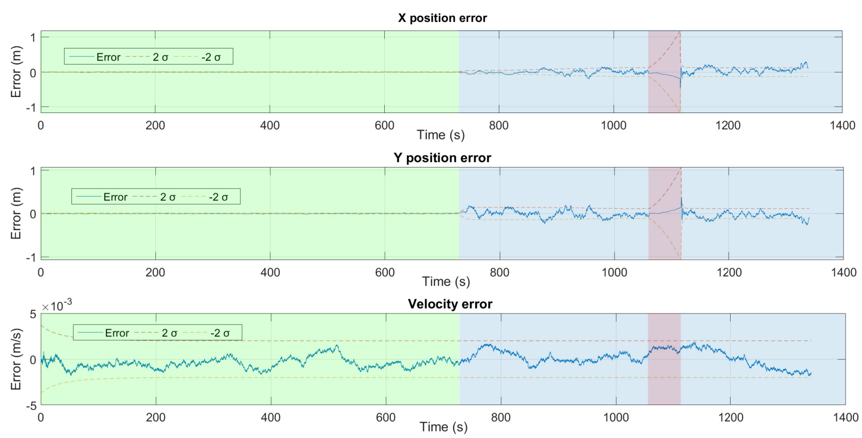


Figure 3.15: Error states with two standard deviations of the mean for GPS/VLOAM navigation. Green: Region with DGPS (0s to 723s time interval or 0 to 5km), Blue: Region without DGPS correction (723s to 1340s time interval or 5km to 15km), Red: Region with multipath errors (1061s to 1118s time interval or 10km to 11km)

Chapter 4

VI-LOAM Version 1.1

4.1 Introduction

This chapter describes version 1.1 of the Visual Inertial Lidar Odometry and Mapping (VI-LOAM) pipeline and demonstrates the performance of the system for captured in-house ground data and online benchmark data. VI-LOAM is a multi-sensory robust navigation solution that localizes a moving platform in an environment while creating a map of the traversed environment. To this end, VI-LOAM combines inertial, lidar, and visual sensor information to provide high accuracy estimation. This version of the pipeline is not aided from global sensors such as GPS. Incorporating global sensors will be addressed in Chapter 5 of this thesis.

4.2 VI-LOAM Version 1.1 Architecture

VI-LOAM version 1.1 pipeline combines visual-inertial odometry (VIO) [37] and lidar odometry (LO) and mapping [4] as shown in Figure 4.1. In this work, lidar odometry and mapping are aided by VIO pose estimation for improved results as suggested in [2]. LO module is initialized by the solution of VIO, and global pose constraints

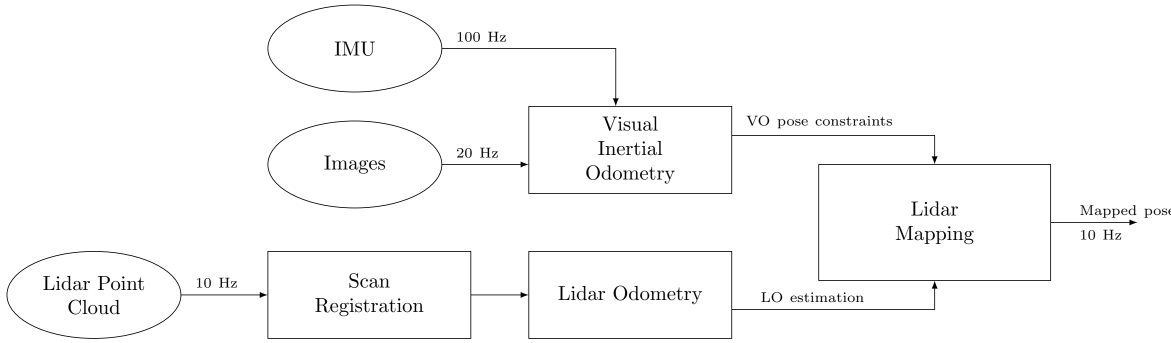


Figure 4.1: Architecture of VI-LOAM version 1.1

from the VIO are added to the optimization of the lidar mapping module. These modifications are found in sections 4.2.3.2 and 4.2.4.1. One main difference from [2] is that the proposed method uses tightly coupled visual-inertial navigation, whereas [2] uses a loosely-coupled approach to combine vision and inertial data, i.e., the IMU mechanization equations run as a separate module, and its bias is periodically updated by using the solutions given by the VIO, LO, and mapping modules. Most of the latest research work use a tightly-coupled approach for state estimation due to improved accuracy compared to the loosely-coupled systems [6, 37, 62, 76]. The functionalities of each module are described in the following sections.

4.2.1 Visual-Inertial Odometry (VIO)

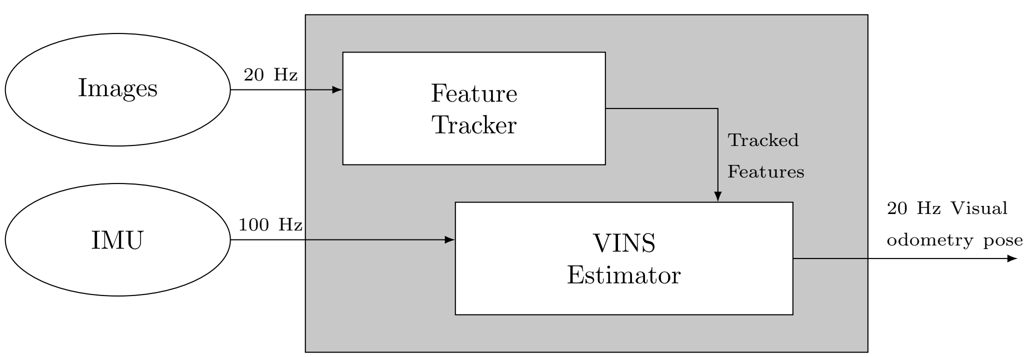


Figure 4.2: Visual inertial odometry module

Visual-inertial odometry module takes IMU and images as inputs and estimates the incremental motion of the system at the camera rate (20 Hz). This module consists of two submodules: feature tracker and odometry estimator. Feature tracker extracts corner features from images using the FAST (Features from Accelerated Segment Test) corner detector method and tracks them across image frames using the OpenCV KLT feature tracker. VIO estimator is adopted from [37] uses constraints from IMU and camera (termed the IMU pre-integration factors [101] and visual feature reprojection factors [37]). Camera constraints are formulated using the tracked features received from the feature tracker. This estimator solves a sliding window-based local optimization problem using Ceres Solver [50] to estimate the incremental motion.

4.2.1.1 Ceres Solver

Throughout the implementation of this system, Ceres Solver [50] is employed to solve optimization problems. Ceres Solver is an open source library for modeling and solving complex optimization problems. Ceres Solver can solve two types of problems:

1. Non-linear Least Squares problems with bounds constraints, and
2. General unconstrained optimization problems.

Ceres Solver solves the non-linear least squares problems of the form:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \frac{1}{2}\sum_i \rho_i\left(\left\|f_i\left(x_{i_1}, ..., x_{i_k}\right)\right\|^2\right) \\ \text{s.t.} \quad & l_j \leq x_j \leq u_j \end{aligned} \tag{4.1}$$

Where, $\rho_i\left(\left\|f_i\left(x_{i_1}, ..., x_{i_k}\right)\right\|^2\right)$ is known as a residual block. $f_i(\cdot)$ is a cost function that depends on the parameters $\{x_{i_1}, ..., x_{i_k}\}$. l_j and u_j are lower and upper bounds

on the parameter block x_j. The cost function is responsible for computing a vector of residuals and Jacobian matrices, $J_i = D_i f(x_1, ..., x_k) \quad \forall i \in \{1, \ldots, k\}$. ρ_i is a loss function that reduces the influence of outliers on the solution.
We can rewrite the the Equation 4.2 as follows;

$$\begin{gathered} \arg\min_x \frac{1}{2}\|F(x)\|^2 \ . \\ L \leq x \leq U \end{gathered} \tag{4.2}$$

Where, $F(x) = [f_1(x), ..., f_m(x)]^\top$ and L are U lower and upper bounds on the parameter vector. The general method to solve non-linear optimization problems is to solve a sequence of approximations to the original problem. For non-linear problems, an approximation can be computed by linearization, $F(x + \Delta x) \approx F(x) + J(x)\Delta x$. Then the non-linear optimization problem becomes;

$$\min_{\Delta x} \frac{1}{2}\|J(x)\Delta x + F(x)\|^2 \tag{4.3}$$

This problem is solved by iteratively updating $x \leftarrow x+\Delta x$. The algorithm convergence depend on the method to control the size of the step size Δx. To this end, the thesis used the Levenberg-Marquardt algorithm [137] given in Ceres Solver.

4.2.1.2 Sliding Window

The sliding window approach is used to increase the computational efficiency and achieve real-time performance. To this end, the optimization is carried out over a bounded-size sliding window of recent states rather than all the previous states. To estimate the system states, only the measurements inside a sliding window is utilized as shown in the Figure 4.3. When estimation is carried out for the next state, last state inside the sliding window is removed and next state with relevant measurements are added to the sliding window.

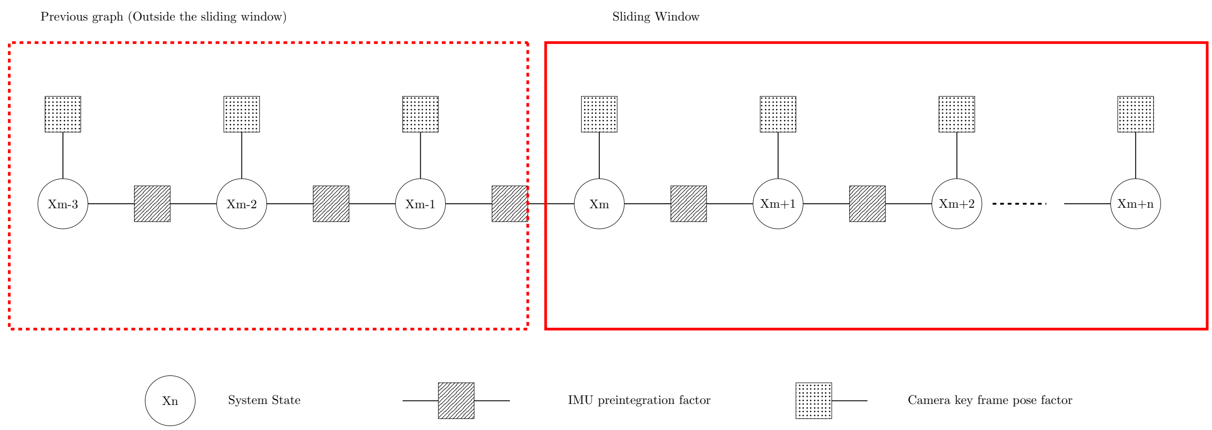


Figure 4.3: Illustration of sliding window optimization for VIO with IMU and camera factors

4.2.1.3 VIO Motion Estimation

The full state vector in the sliding window of VIO is as follows;

$$\boldsymbol{\chi} = \left[\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_n, \lambda_0, \lambda_1, ..., \lambda_m\right]$$

$$\mathbf{x}_k = \left[\mathbf{p}^w_{b_k} \quad \mathbf{v}^w_{b_k} \quad \mathbf{q}^w_{b_k} \quad \mathbf{b}_a \quad \mathbf{b}_g\right], k \in [0, n] \tag{4.4}$$

$$\mathbf{x}^b_c = \left[\mathbf{p}^b_c \quad \mathbf{q}^b_c\right]$$

where $\boldsymbol{\chi}$ is the keyframe vector with visual keyframe poses $\mathbf{x}_n$ and inverse depth of features λ_m. n is the total number of keyframes, and m is the total number of features in the sliding window. $\mathbf{x}_k$ is the IMU state at the k^{th} image time. IMU state has position, velocity, orientation of the IMU in the world frame and acceleration bias, and gyroscope bias in IMU's body frame. The extrinsic parameters between the camera and the IMU is given by $\mathbf{x}_c^b$.

The maximum posterior estimation is obtained by minimizing the sum of prior and the Mahalanobis norm [138] of all measurement residuals:

$$\min_{\boldsymbol{\chi}} \left\{ \|\mathbf{r}_p - \mathbf{H}_p \boldsymbol{\chi}\|^2 + \sum_{k \in B} \|\mathbf{r}_{B}(\mathbf{z}_{b_k+1}^{b_k}, \boldsymbol{\chi})\|^2_{\mathbf{p}_{b_k+1}^{b_k}} + \sum_{(l,j) \in C} \rho(\|\mathbf{r}_{C}(\mathbf{z}_l^{c_j}, \boldsymbol{\chi})\|^2_{\mathbf{p}_l^{c_j}}) \right\} \tag{4.5}$$

Where the Huber norm, $\rho(s)$, is the loss function; $\rho(s) = \begin{cases} 1 & s \geq 1 \\ 2\sqrt{s} - 1 & s < 1 \end{cases}$

To solve the Equation 4.5 and estimate the system states, measurements inside a sliding window is utilized as explained in section 4.2.1.2. The images inside the sliding window are between the m^{th} frame and $(m+n)^{\text{th}}$ frame. $\boldsymbol{C}$ is the set of features that have been observed two or more times in the current sliding window, and $\boldsymbol{B}$ is the set of all IMU measurements. $\mathbf{r}_{B}(\mathbf{z}_{b_k+1}^{b_k}, \boldsymbol{\chi})$ and $\mathbf{r}_{C}(\mathbf{z}_l^{c_j}, \boldsymbol{\chi})$ are residuals for IMU and visual measurements respectively. $[\mathbf{r}_p, \mathbf{H}_p]$ is the prior information from marginalization. This nonlinear problem is solved using the Ceres Solver.

The proposed system tracks around 150-200 features across images. The odometry is published with respect to the IMU frame at 20 Hz, which is the camera's image capture frequency.

4.2.2 Scan Registration

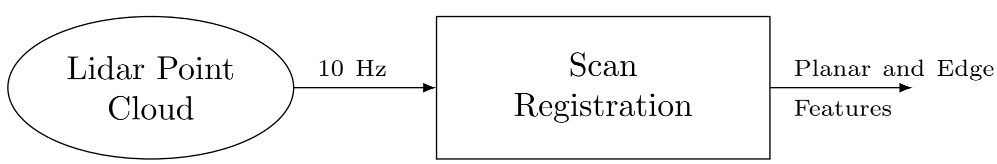


Figure 4.4: Scan registration module

The scan registration module determines the number of channels in the lidar point cloud received and arranges the points received orderly with respect to time and channel number for further processing. The number of channels can vary from 16, 32 to 64 for Velodyne lidars [139]. Further, this module extracts four different planar and edge features from the lidar point cloud and publishes them under different topics used for the lidar odometry estimation.

4.2.2.1 Planar and Edge Point Extraction

Let $\hat{\mathcal{P}}$ be the points received in the laser scan. These points are registered in the lidar frame, and the combined point cloud during sweep k is $\mathcal{P}_k$. Let i be a point in $\mathcal{P}_k$. Then, $\mathcal{S}$ is the set of consecutive points of i in the same scan. The lidar frame coordinates of a point i, $i \in \{L\}$ are denoted as $\mathbf{X}^L_{(k,i)}$. [4] has defined a parameter to evaluate the smoothness of the local surface and extract features;

$$c = \frac{1}{|\mathcal{S}|.\|\mathbf{X}^L_{(k,i)}\|} \| \sum_{\mathbf{j}\in S, \mathbf{j}\neq i} (\mathbf{X}^L_{(k,i)}) - \mathbf{X}^L_{(k,j)} \| \tag{4.6}$$

Based on the c values, the points in a scan are sorted. The points with maximum c values are classified as edge points, and the points with minimum c values are classified as planar points.

4.2.3 Lidar Odometry (LO)

The LO module calculates motion within a lidar sweep using the planar and edge features received from the scan registration module. To this end, it runs a non-linear optimization using Ceres Solver with lidar constraints. Lidar constraints are computed by calculating the distances between matched point to plane and point to edge. The LO module estimates the odometry at 10 Hz which is the lidar update rate.

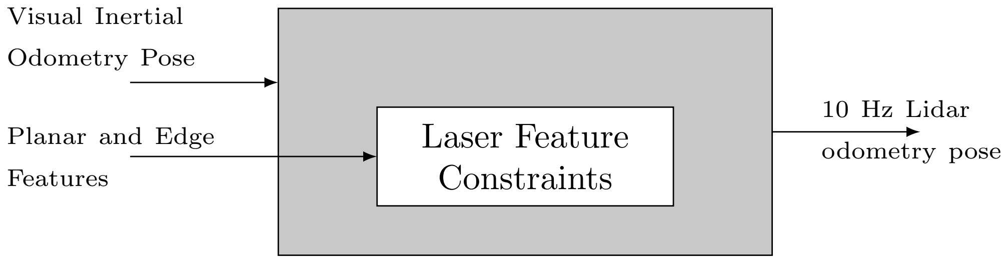


Figure 4.5: Lidar odometry module

4.2.3.1 LO Motion Estimation

Let $\mathcal{P}_k$ be the point cloud of previous k^{th} sweep and $\mathcal{P}_{k+1}$ be the point cloud of current $(k+1)^{\text{th}}$ sweep. $\mathcal{P}_k$ is reprojected to t_{k+1} timestamp and it is denoted as $\bar{\mathcal{P}}_k$. At the beginning of sweep $k+1$, $\mathcal{P}_{k+1}$ is an empty set and as more points are received, it grows during the course of the sweep. Lidar odometry recursively estimates the 6-DoF motion during the sweep. Let $\mathcal{E}_{k+1}$ and $\mathcal{H}_{k+1}$ be the sets of edge points and planar points in $\mathcal{P}_{k+1}$. For a point $i \in \mathcal{E}_{k+1}$, if (j, l) is the corresponding edge line, $\{j,\, l\} \in \bar{\mathcal{P}}_k$, the point to line distance is given by $d_{\mathcal{E}}$. For a point $i \in \mathcal{H}_{k+1}$, if (j, l, m) is the corresponding planar patch, $\{j,\, l,\, m\} \in \bar{\mathcal{P}}_k$, the point to plane distance is given by $d_{\mathcal{H}}$. Refer [4] for the derived equations for $d_{\mathcal{E}}$ and $d_{\mathcal{H}}$.

Let $\mathbf{T}^L_{k+1}$ be the lidar pose transform between t_{k+1} and t where t is the current timestamp and t_{k+1} is the starting time of sweep $k+1$. $\mathbf{T}^L_{k+1}$ consist of translation and rotation angles with respect to lidar frame, $\{L\}$, $\mathbf{T}^L_{k+1} = [t_x, t_y, t_z, \theta_x, \theta_y, \theta_z]$. Then we we can derive a geometric relationship between an edge points and planar points and the pose transform.

$$f_{\mathcal{E}}(\mathbf{X}^L_{(k+1,i)}, \mathbf{T}^L_{k+1}) = d_{\mathcal{E}}, \quad i \in \mathcal{E}_{k+1} \tag{4.7}$$

$$f_{\mathcal{H}}(\mathbf{X}^L_{(k+1,i)}, \mathbf{T}^L_{k+1}) = d_{\mathcal{H}}, \quad i \in \mathcal{H}_{k+1} \tag{4.8}$$

Finally, by stacking 4.7 and 4.8 for each matched feature point, a nonlinear function is obtained. Ceres Solver solves this nonlinear optimization problem by minimizing the distances $d_{\mathcal{E}}$ and $d_{\mathcal{H}}$ towards zero [50].

4.2.3.2 Initialization

In addition to the implementation proposed in [4], the proposed method provides the initial value for the nonlinear optimization using the odometry value computed in the VIO module. This enables optimization to converge to the optimum value faster as the initial value provided by the frame to frame motion of VIO is much closer to the optimum solution. To this end, a matching VIO update for each lidar sweep is identified.

Let visual-inertial odometry at lidar sweep timestamps t_k and t_{k+1} are ${}^{v}\mathbf{X}^{I}_{(k)}$ and ${}^{v}\mathbf{X}^{I}_{(k+1)}$ respectively. The VIO is published with respect to the IMU world frame $\{I\}$. $\mathbf{X}^{I}_{(k)} = \begin{bmatrix} \mathbf{p}^{k}_{I} & \mathbf{q}^{k}_{I} \end{bmatrix}$ where $\mathbf{p}^{k}_{I}$ is the translation vector and $\mathbf{q}^{k}_{I}$ is the rotation quaternion. Then, frame to frame motion in IMU frame is computed as;

$$ {}^{v}\mathbf{q}^{I}_{(k,k+1)} = (\mathbf{q}^{I}_{k})^{-1} \otimes {}^{v}\mathbf{q}^{I}_{k+1} \tag{4.9} $$

$$ {}^{v}\mathbf{p}^{I}_{(k,k+1)} = \mathbf{q}^{I-1}_{k} \otimes ({}^{v}\mathbf{p}^{I}_{k} - {}^{v}\mathbf{p}^{I}_{k+1}) \tag{4.10} $$

Let tranformation matrix between lidar and IMU is $\mathbf{T}^{L}_{I} = \begin{bmatrix} \mathbf{p}^{L}_{I} & \mathbf{q}^{L}_{I} \end{bmatrix}$. Using 4.9 and 4.10, VIO frame to frame motion with respect to the lidar frame $\{L\}$ can be derived as follows;

$$ {}^{v}\mathbf{q}^{L}_{(k,k+1)} = \mathbf{q}^{L}_{I} \otimes {}^{v}\mathbf{q}^{I}_{(k,k+1)} \tag{4.11} $$

$$ {}^{v}\mathbf{p}^{L}_{(k,k+1)} = (\mathbf{q}^{L}_{I} \otimes {}^{v}\mathbf{p}^{I}_{(k,k+1)}) + \mathbf{p}^{L}_{I} \tag{4.12} $$

4.2.4 Lidar Mapping

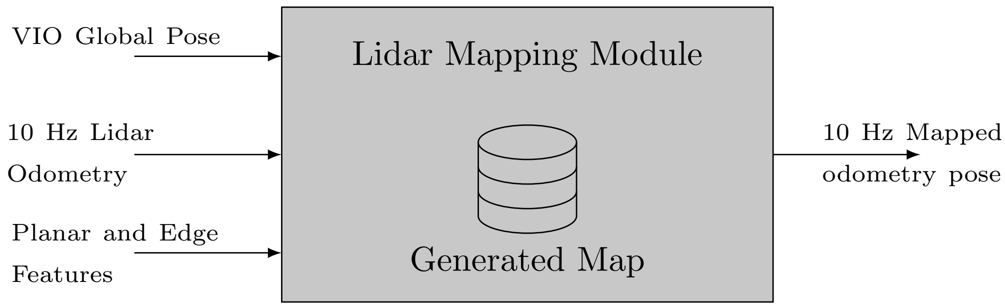


Figure 4.6: Lidar mapping module

Lidar mapping further refines the odometry by carrying out a batch optimization and updates a map of the travelled environment. The mapping algorithm is also adopted from [4] and updates once per sweep, i.e., at 10 Hz.

Lidar odometry generates undistorted point cloud $\hat{\mathcal{P}}_{k+1}$ and pose transform $\mathbf{T}^{L}_{k+1}$ which contains the lidar motion during the sweep. Let us define $\mathcal{Q}_k$ as the point cloud of the map accumulated until sweep k and $\mathbf{T}^{W}_{k}$ be the pose of the lidar on the map at the end of sweep k where $\{W\}$ represents the world coordinate frame. As shown in Figure 4.7 the mapping algorithm extends $\mathbf{T}^{W}_{k}$ for one more sweep from t_k to t_{k+1}, obtaining $\mathbf{T}^{W}_{k+1}$ while simultaneously projecting $\hat{\mathcal{P}}_{k+1}$ to the world coordinates $\{W\}$, denoted as $\hat{\mathcal{Q}}_{k+1}$. Then, the algorithm matches $\hat{\mathcal{Q}}_{k+1}$ to the existing map $\mathcal{Q}_k$ by optimizing the lidar pose $\mathbf{T}^{W}_{k+1}$.

To this end, similar to the lidar odometry process, the edge and planar geometric features of the current sweep's point cloud is matched with the existing map. However, for this optimization, ten times more features are used since map is updated after ten sweeps. The nonlinear optimization is solved using the Ceres Solver [50]. This map is incrementally built by down-sampling and registering the incoming point cloud on a voxel grid.

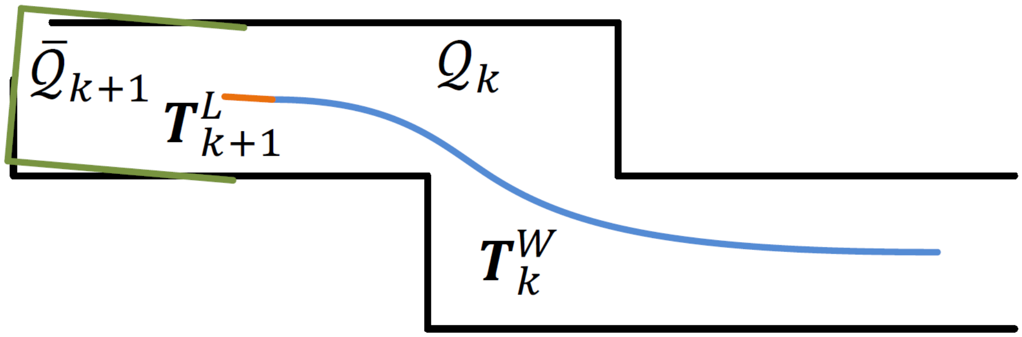


Figure 4.7: Illustration of mapping process [4]

4.2.4.1 Visual pose constraints

In this work, we have improved the mapping algorithm by using global pose constraints from the previous VIO estimate to solve the optimization problem. First, the pose from the VIO module is transformed to the world frame $\{W\}$. In our implementation, initial lidar frame $\{L\}$ is taken as the reference world frame $\{W\}$.
Let ${}^v\mathbf{X}^I_{(k+1)} = \begin{bmatrix} {}^v\mathbf{p}^I_{(k+1)} & {}^v\mathbf{q}^I_{(k+1)} \end{bmatrix}$ be the VIO pose at lidar timestamp t_{k+1}. Then, this can be transformed to world frame using the world to IMU transformation matrix, $\mathbf{T}^W_I = \begin{bmatrix} \mathbf{p}^W_I & \mathbf{q}^W_I \end{bmatrix}$.

$$ {}^v\mathbf{q}^W_{(k+1)} = \mathbf{q}^W_I \otimes {}^v\mathbf{q}^I_{(k+1)} \tag{4.13}$$

$$ {}^v\mathbf{p}^W_{(k+1)} = (\mathbf{q}^W_I \otimes {}^v\mathbf{p}^I_{(k+1)}) + \mathbf{p}^W_I \tag{4.14}$$

Let the variance of translation and rotation of VIO pose estimate be σ_t^2 and σ_R^2 respectively. $\mathbf{T}^W_{k+1} = \begin{bmatrix} \mathbf{p}^W_{(k+1)} & \mathbf{q}^W_{(k+1)} \end{bmatrix}$ is the motion to be solved. The constraints for the motion using the VIO global pose can be formulated as:

$$\begin{bmatrix} (\mathbf{p}^W_{(k+1)} - {}^v\mathbf{p}^W_{(k+1)})/\sigma_t^2 \\ ({\mathbf{q}^W_{(k+1)}}^{-1} \otimes {}^v\mathbf{q}^W_{(k+1)})/\sigma_R^2 \end{bmatrix} = 0 \tag{4.15}$$

These constraints are added to the optimization problem together with lidar edge and planar feature constraints and solved using Ceres Solver. The mapping and the odometry refinement are carried out at 1 Hz. However, previous lidar odometry is integrated on top of that to provide 10 Hz mapped odometry update.

4.3 Results

The VI-LOAM version 1.1 implementation was tested and evaluated with two datasets. The KITTI online benchmark dataset [5] and AI4L payload datasets which were captured locally in St John‘s, NL.

4.3.1 Datasets

4.3.1.1 KITTI Dataset

The KITTI dataset [5] consists of around 6 hours of data captured from driving in city of Karlsruhe, Germany. It includes grayscale and color camera images, laser point clouds, GPS measurements, and IMU data. The sensors they have used are as follows:

- Two PointGray Flea2 grayscale cameras, 1.4 Megapixels at 10 Hz
- Two PointGray Flea2 color cameras, 1.4 Megapixels at 10 Hz
- Velodyne HDL-64E rotating 3D laser scanner at 10 Hz
- OXTS RT3003 inertial and GPS navigation system at 100 Hz

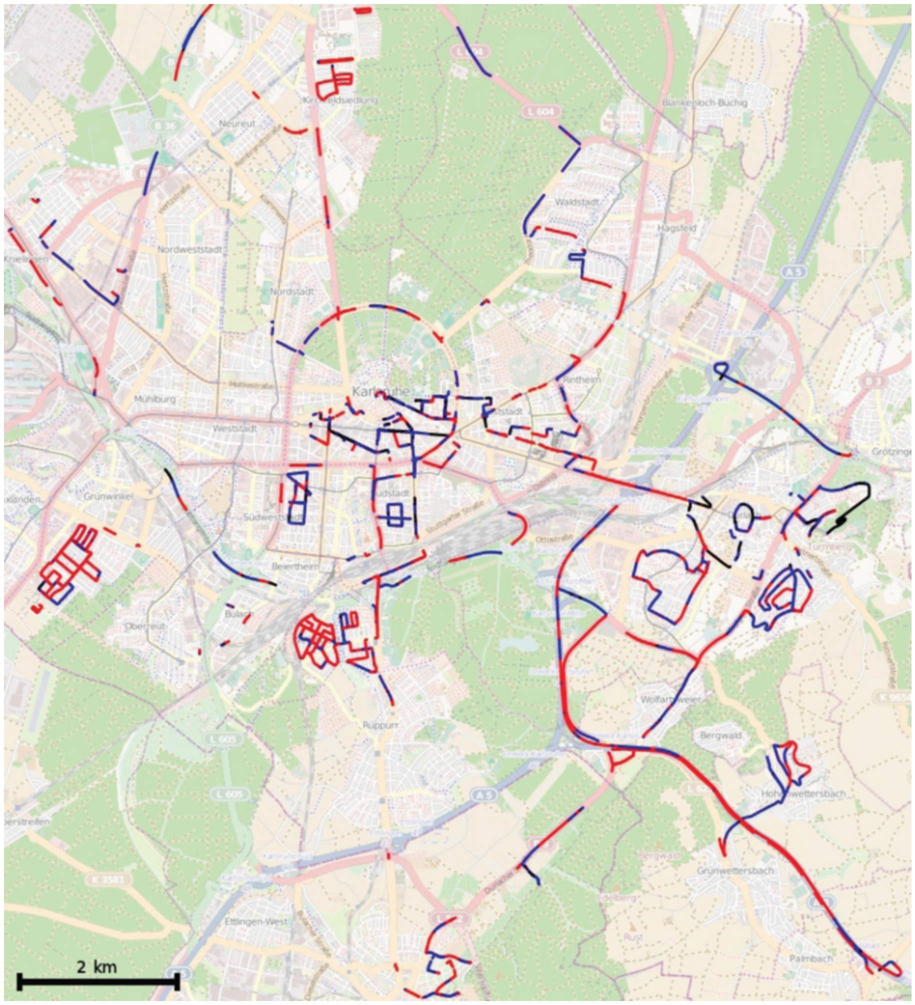


Figure 4.8: Recording zones of the KITTI dataset. This figure shows the GPS traces of the recordings [5]

The images are post-processed and cropped to a size of 1382 x 512-pixel resolution. One limitation of this system is that, even though the IMU rate is 100 Hz, the synced IMU data with other sensors are only available at 10 Hz. The sensor setup with their coordinate frames is shown in Figure 4.9. The sensor calibrations, IMU to lidar, and lidar to camera, can be found here: http://www.cvlibs.net/datasets/kitti/rawdata.php

Figure 4.9: KITT dataset sensor setup with their coordinate frames [5]

4.3.1.2 AI4L Payload Data

AI4L Payload is a custom sensor payload designed, implemented and tested by researchers in Intelligent Systems Laboratory (ISLab), Memorial University of Newfoundland, collaborating with NRC Flight Research Lab, Ottawa, Canada. The payload consists of four sensors:

- FLIR backfly camera,1.6 Megapixels at 20 Hz (Resolution: 1440 x 1080 pixels)
- Velodyne VLP-16 rotating 3D laser scanner at 10 Hz
- Ublox F9P GPS receiver running on RTK mode with a reach RS2 base station at 10 Hz
- Xsense MTI 30 IMU at 200 Hz

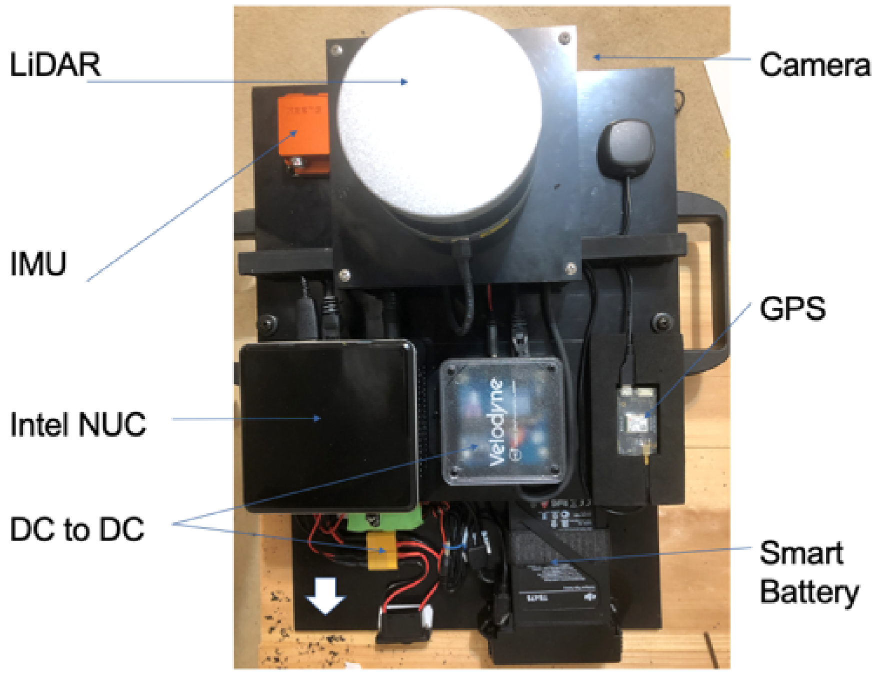


Figure 4.10: AI4L Payload sensor setup

The AI4L payload sensor setup is shown in Figure 4.10. The calibrations parameters of sensors used, IMU to camera and lidar to the camera, are mentioned below. The IMU to camera calibration was achieved using the Kalibr [140] toolbox, and the camera to laser calibration was found using the Matlab visual Lidar calibration tool.

$$
{}^{C}T_{I} = \begin{bmatrix} -0.0009792 & 0.00685726 & 0.99997601 & 0.18648448 \\ 0.99931723 & -0.03692628 & 0.00123178 & -0.04199414 \\ 0.03693384 & 0.99929446 & -0.00681642 & -0.03693199 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} R \end{bmatrix}_{3\times 3} & \begin{bmatrix} t \end{bmatrix}_{3\times 1} \\ \begin{bmatrix} 0 \end{bmatrix}_{1\times 3} & 1 \end{bmatrix}
$$

The standard t block of the homogeneous transformation matrix provides the translation between the camera and the IMU, and the standard R block corresponds to the rotation of the camera with respect to the IMU. Similarly, following homogeneous transformation matrix provides translation and rotation between the camera and the lidar.

$$
{}^{C}T_{L} = \begin{bmatrix} -0.0062 & -0.0009 & 1.000 & 0.0923 \\ -0.9992 & 0.0391 & -0.0061 & 0.0388 \\ -0.0391 & -0.9992 & -0.0012 & -0.0740 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

We captured several datasets while driving around the Memorial University of Newfoundland, Canada. GPS traces of one of them is shown in Figure 4.11, and it has a path length of around 2.3 km. The datasets include color camera images, laser point clouds, GPS measurements, and IMU data.

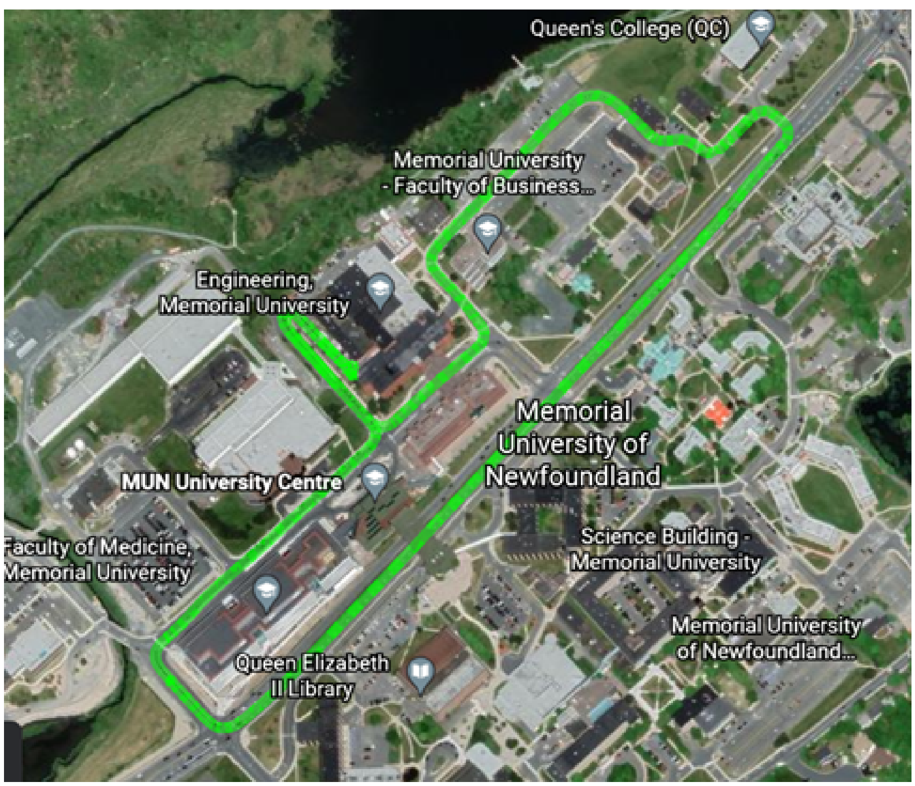
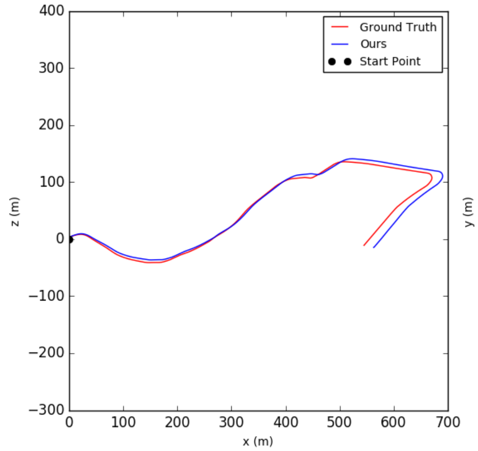


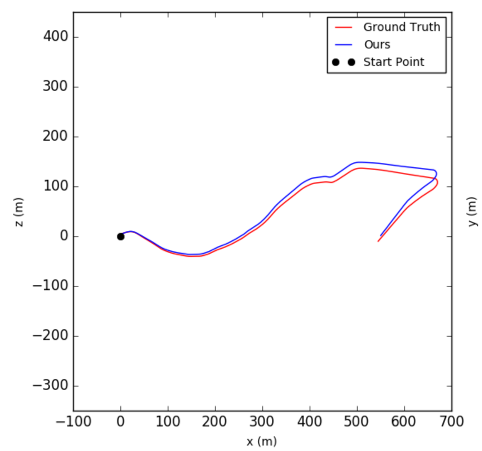
Figure 4.11: GPS traces of one of the datasets captured around the Memorial university of Newfoundland, Canada

4.3.2 KITTI Dataset Results

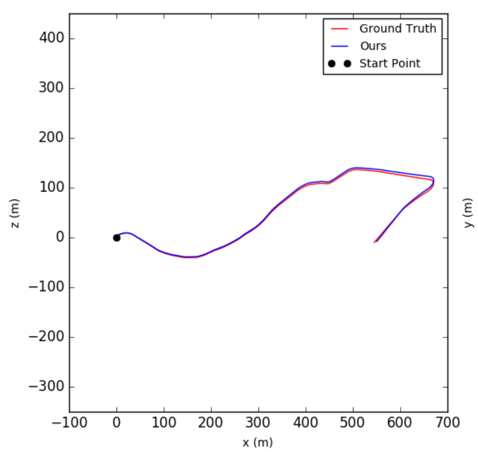
The proposed VI-LOAM version 1.1 was tested with three sequences of the KITTI odometry benchmark dataset which include sequence ’04’, ’06’ and ’10’. To this end, stereo visual-inertial odometry was used instead of monocular visual-inertial odometry because the IMU update rate of KITTI data set is not sufficient for generating accurate monocular VIO. These results were used to verify if adding lidar improves the estimation performance. The resultant estimated path with the ground truth for each navigation system for one of the sequences is presented in Figure 4.12. It indicates that VI-LOAM version 1.1 estimation is closer to the ground truth path than the other state-of-the-art systems.



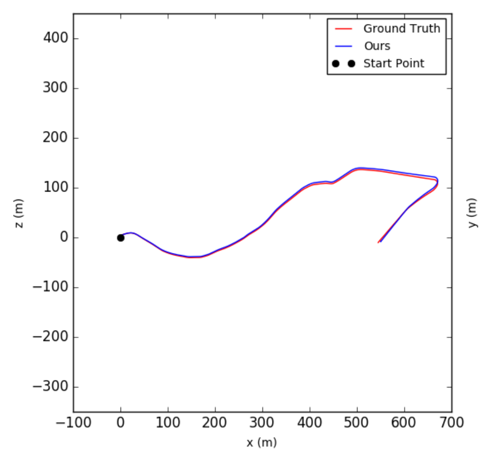
(a) Visual-inertial Odometry (VIO)



(b) Lidar Odometry (LO)



(c) VIO initialized LO



(d) VI-LOAM version 1.1

Figure 4.12: KITTI dataset sequence 10 results for each navigation solution

A summary of position and orientation estimation accuracy for each navigation method for KITTI datasets are presented in Table 4.1. According to that, VI-LOAM version 1.1 system has managed to achieve close to 1% translation accuracy and it outperforms all the other state-of-the-art navigation methods (VIO [37] and LOAM [4]) presented here for the selected KITTI data sequences. Moreover, rotation errors are also improved when the VI-LOAM system is introduced. (Note: VIO initialized LO is an intermediate architecture that only uses VIO as initialization to the LO.)

Table 4.1: Performance evaluation of each navigation method for KITTI benchmark online dataset

KITTI Sequence	Error Type	VIO	LO	VIO Initialized LO	VI-LOAM
04	Translation Error %	4.5345	1.2690	1.2655	**1.1977**
	Rotation Error (deg/m)	0.0062	0.0047	0.0046	**0.0045**
06	Translation Error %	3.1876	1.6127	1.6995	**1.4243**
	Rotation Error (deg/m)	0.0062	0.0071	0.0071	**0.0065**
10	Translation Error %	3.032	1.7721	1.5530	**1.4339**
	Rotation Error (deg/m)	0.0079	0.0068	**0.0053**	**0.0053**

4.3.3 AI4L Payload Data Results

The proposed VI-LOAM version 1.1 was evaluated on two data sets that were captured around the Memorial University of Newfoundland utilizing the AI4L sensor payload. The resultant estimated path with the generated map of the environment for each dataset is presented in Figure 4.13 and 4.14. The paths for the stand-alone VIO and stand-alone LO systems were also generated and compared with the VI-LOAM version 1.1 implementation.

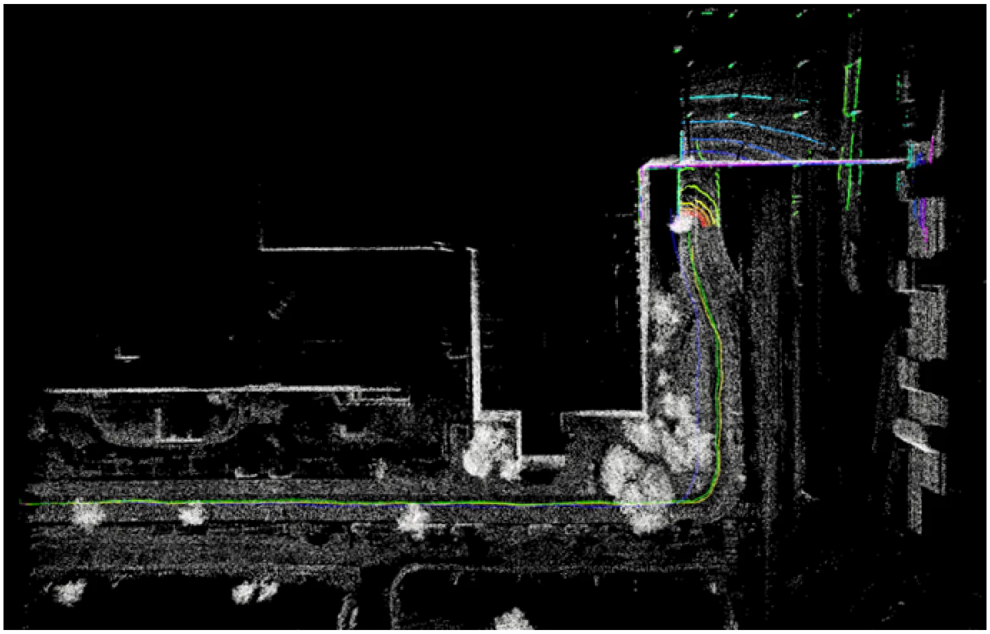


Figure 4.13: Estimated paths and ground truth with the generated map for the AI4L payload dataset #1. Colors illustrate different navigation methods and ground truth. Yellow track is for ground truth. Blue track is for standalone VIO and Green track is for VI-LOAM version 1.1 estimation.

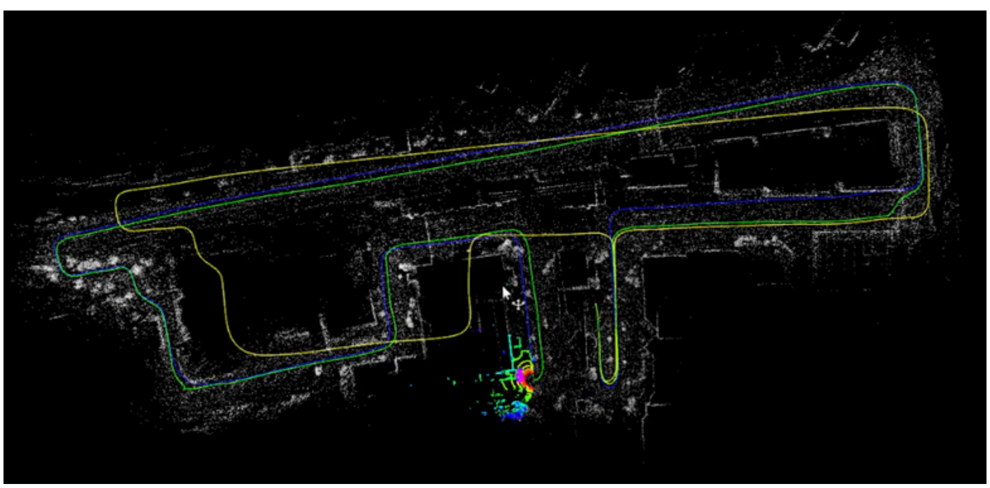
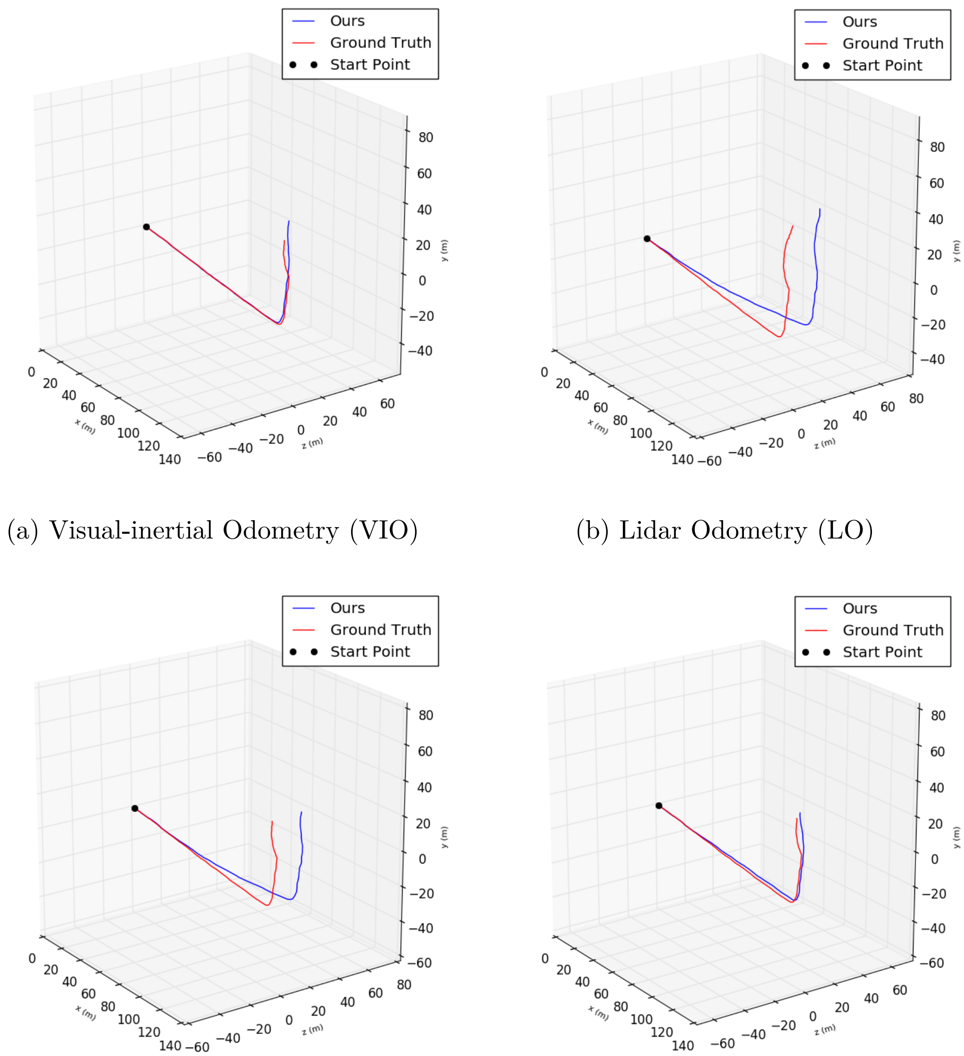


Figure 4.14: Estimated paths and ground truth with the generated map for the AI4L payload dataset #2. Colors illustrate different navigation methods and ground truth. Yellow track is for ground truth. Blue track is for standalone VIO and Green track is for VI-LOAM version 1.1 estimation.

4.3.3.1 AI4L Payload Dataset #1



(a) Visual-inertial Odometry (VIO)

(b) Lidar Odometry (LO)

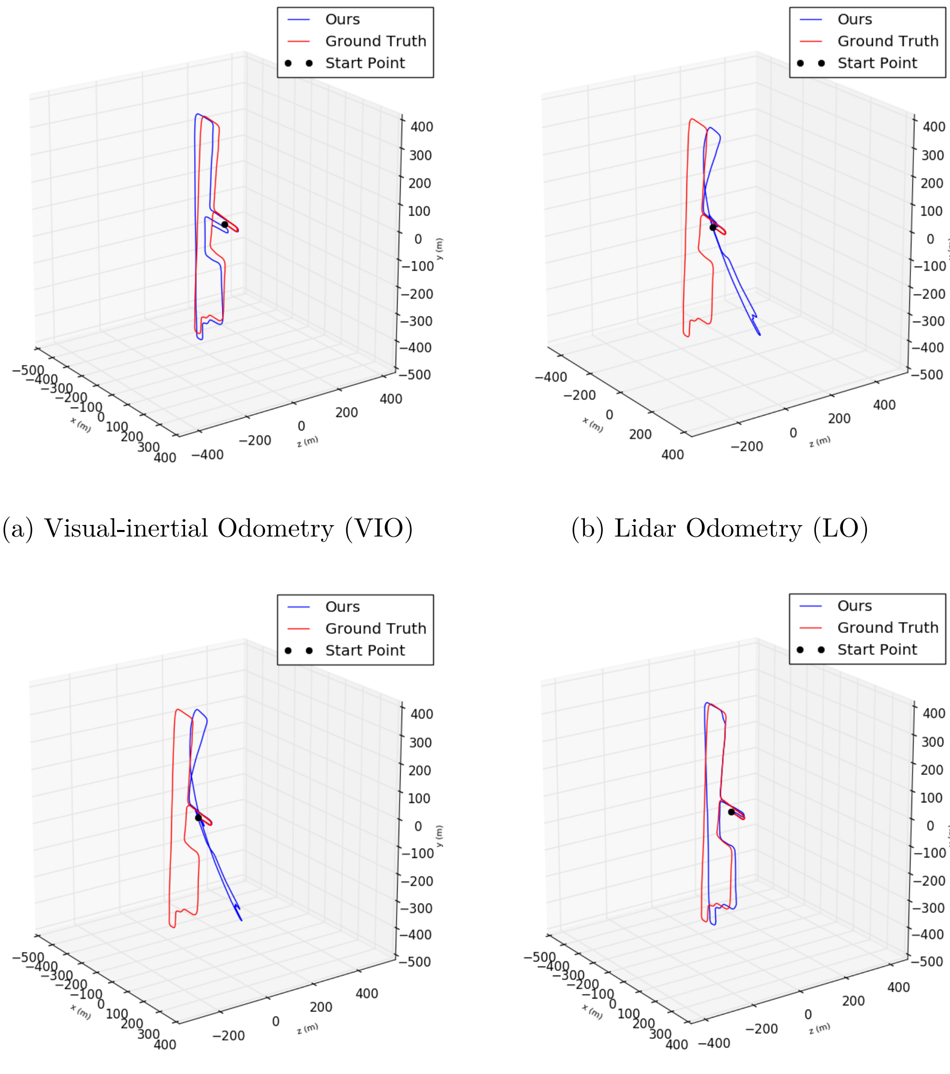
(c) VIO initialized LO

(d) VI-LOAM version 1.1

Figure 4.15: Results of each navigation solution for AI4L payload dataset #1.

The estimated path with the ground truth for each navigation system for the AI4L payload dataset #1 is illustrated in Figure 4.15. According to that, stand-alone LO had a significant variation in the z-direction, whereas stand-alone VIO had some drift in the x and y-direction. VI-LOAM version 1.1 estimation managed to minimize the variation in z-direction while reducing the drift in x and y directions as well.

4.3.3.2 AI4L Payload Dataset #2



(a) Visual-inertial Odometry (VIO)

(b) Lidar Odometry (LO)

(c) VIO initialized LO

(d) VI-LOAM version 1.1

Figure 4.16: Results of each navigation solution for AI4L payload dataset #2.

The estimated path with the ground truth for each navigation system for the AI4L payload dataset #2 is shown in Figure 4.16. According to that, stand-alone LO and stand-alone VIO had similar performance to dataset #1, in which LO had large variations in the z-direction and VIO had drift in other directions as shown in Figure 4.16. Additionally, for that dataset, VIO was unable to estimate the rotations correctly at the corners introducing a more erroneous final pose. Even though z-direction variations were corrected and drift errors were minimized with VI-LOAM version 1.1 estimation, this rotation error was unable to correct with the VI-LOAM version 1.1 system as well. However, these results were generated without the loop closure algorithm, which is useful to identify revisiting the same place and correct the path accordingly. So, this should be evaluated with a loop closure module for better performance. Moreover, these results are only preliminary results, and there are specific improvements, such as improving the calibration of the camera, which can further improve the estimation.

Table 4.2 summarizes the performance of VI-LOAM version 1.1 system for AI4L payload datasets. In this evaluation, GPS position information was used as the ground truth. Since the GPS position data does not provide any orientation information, the results presented in Table 4.2 do not include rotational error values. According to the results, VI-LOAM version 1.1 achieves the best accuracy compared to the other state-of-the-art navigation methods (VIO [37] and LOAM [4]). However, the translation drift is between 2.5% and 9% which is as not as good as the accuracy values for the KITTI dataset. This is due to availability of stereo visual odometry and more accurate sensor calibrations for KITTI odometry data. Moreover, the ground truth GPS used for the payload evaluation is not accurate as ground truth provided by KITTI dataset. Additionally, errors have occurred due to the inability to estimate the yaw angle accurately. Fixing these issues will improve the accuracy of our proposed navigation system.

Table 4.2: Performance evaluation of each navigation method for Payload dataset sequences

Payload Sequence	Error Type	VIO	LO	VIO Initialized LO	VI-LOAM
01	Translation Error %	6.3208	16.9296	11.8107	**2.7096**
	RMSE (m)	10.4	20.17	16.22	**3.5**
02	Translation Error %	12.1860	16.3915	15.2838	**8.5199**
	RMSE (m)	28.14	34.08	21.26	**20.15**

4.3.4 Execution Time Results

The execution time of the current VI-LOAM version 1.1 implementation was evaluated for two different devices: (i) a laptop computer which has an Intel Core i7-8750H CPU @ 2.20GHz with 16 GB RAM and 8 GB VGA (NVIDIA GeForce 1070) and (ii) a NVIDIA Jetson Xavier with 8-core ARM CPU, 32 GB RAM and 512-core Volta GPU. To achieve the real-time performance, odometry needs to updated at 10 Hz which is the update rate of the lidar sensor. Therefore, feature tracker, visual odometry, and lidar odometry needs to complete under 100 ms.

Execution times for the laptop computer is presented in Table 4.3. According to the results, the total odometry mean time is 113.3 ms (Total mean time of feature tracker, visual odometry, and lidar odometry nodes) which is marginally higher than the expected 100 ms.

Table 4.3: Execution time statistics for the laptop computer (in seconds)

Statistic	Feature Tracker	Visual Odometry	Scan Registration	Lidar Odometry	Lidar Mapping (Visual aided)
Mean	40.8	57.9	3.7	14.6	135.9
Std. Deviation	31	20.4	1.2	4.4	80.25
Max	130	147	13.5	35.4	350.5

NVIDIA Jetson Xavier module execution times are presented in Table 4.4. The total odometry mean time is 94.7 ms which is under 100 ms, indicating the ability to perform the navigation in real-time. However, these results are limited to a few datasets, and more validation needs to be done, which will be completed as part of the improvements described in chapter 6.

Table 4.4: Execution time statistics for the NVIDIA Jetson Xavier module (in seconds)

Statistic	Feature Tracker	Visual Odometry	Scan Registration	Lidar Odometry	Lidar Mapping (Visual aided)
Mean	8.5	62.2	5.3	24.0	119.4
Std. Deviation	3.4	18.0	1.9	6.2	36.2
Max	15.7	91.3	18.5	54.9	120.5

The execution time statistics of each module of the VI-LOAM version 1.1 system for each device are presented below in box and whisker plots.

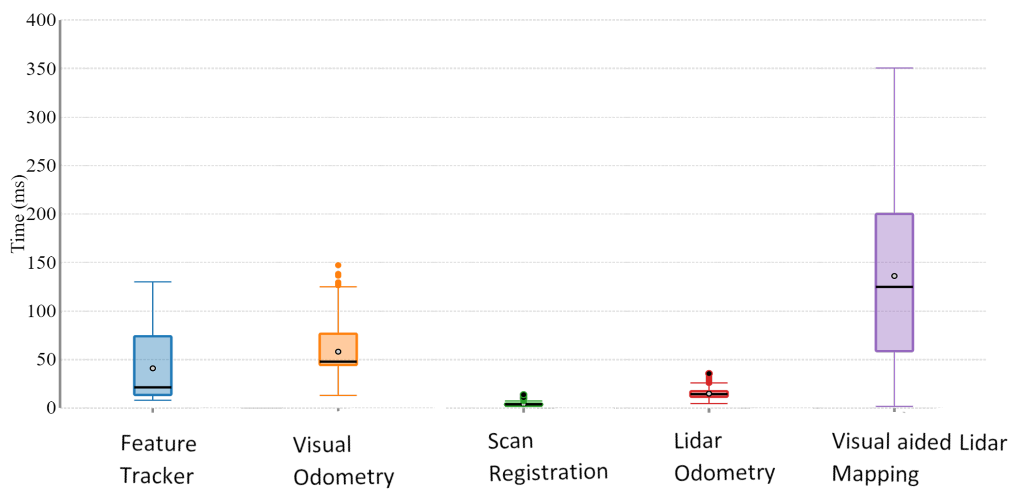


Figure 4.17: Box and whisker plot of execution times for the laptop computer

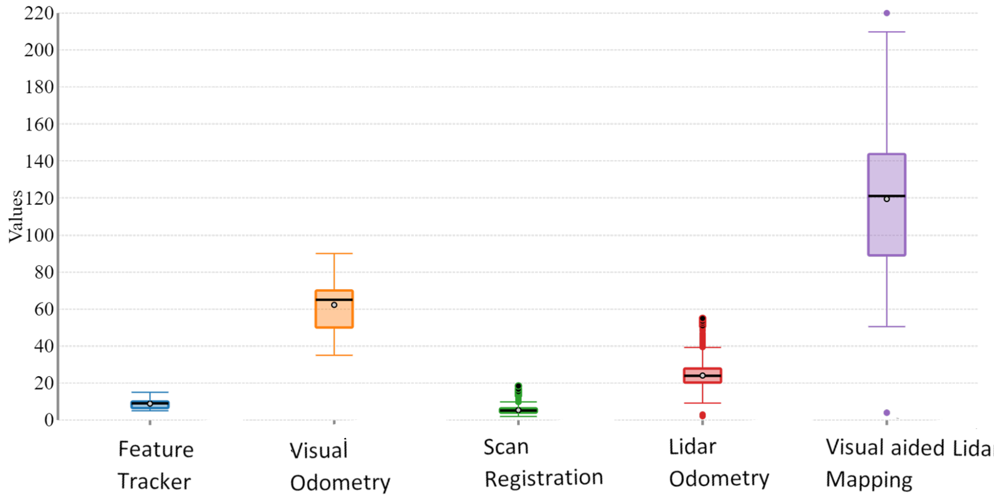


Figure 4.18: Box and whisker plot of execution times for the NVIDIA Jetson Xavier module

According to the box and whisker plot, execution times of feature tracker and visual odometry modules have improved in the NVIDIA Jetson Xavier implementation compared to the laptop computer. This is due to the GPU implementation of the visual odometry algorithm and Jetson Xavier manages to complete the process under 100 ms.

4.4 Summary

The implementation of the proposed novel VI-LOAM version 1.1 architecture and comparison with state-of-the-art navigation methods were presented in this chapter. According to the results, the proposed VI-LOAM version 1.1 has better pose accuracy than the state-of-the-art methods. However, some improvements, such as correcting orientation errors in yaw angle, are yet to be carried out. The next chapter discusses the VI-LOAM version 1.2 with the improvements. Additionally, the next chapter incorporates GPS with the novel VI-LOAM navigation system and delivers results for different GPS scenarios.

Chapter 5

VI-LOAM Version 1.2

5.1 Introduction

This chapter describes improvements in version 1.2 of the VI-LOAM pipeline and the results of this system. The main addition is the global measurements like GPS in the pipeline. Additionally, the architecture performs several algorithmic improvements to the architecture in version 1.1.

5.2 VI-LOAM Version 1.2 Architecture

Similar to VI-LOAM version 1.1, VI-LOAM version 1.2 consists of visual-inertial odometry, lidar odometry, and lidar mapping modules. Additionally, the mapped odometry is further corrected using GPS signals in the global fusion node, as shown in Figure 5.1. Furthermore, the following changes were carried out:

- incorporate depth information from lidar to visual features;
- include VIO frame to frame motion constraints in the lidar odometry optimization;

- include only roll and pitch global orientation constraints in lidar mapping optimization in contrast to including 6-DoF pose constraints in version 1.1, and
- update the VI-LOAM odometry estimation using GPS signals as and when they become available.

These revisions are discussed in sections 5.2.1, 5.2.2, 5.2.3, and 5.2.4, respectively.

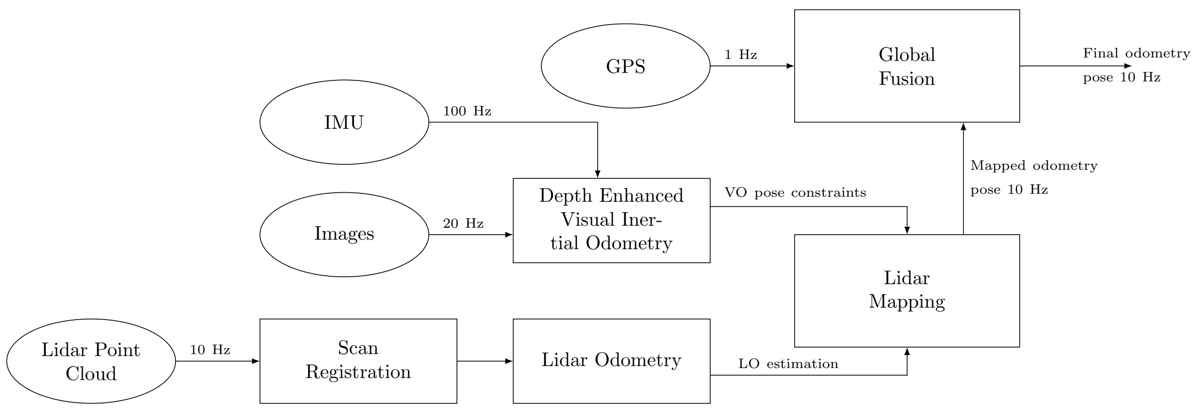


Figure 5.1: Architecture of VI-LOAM version 1.2

5.2.1 Depth-Enhanced Visual Inertial Odometry

Previous visual inertial odometry module is adopted from [37] and derive two types of camera constraints using the visual features:

- features without depth, and
- features with depth from triangulation.

In this implementation, these constraints are extended by adding depth measurements from lidar to image features. As lidar depth measurement is more accurate, depth association from lidar take precedence over triangulation. This method is adopted from [6] and the new constraint is features with depth associated from lidar.

To this end, lidar frames are registered to the camera frame using the estimated visual-inertial odometry. Multiple lidar scans are stacked to obtain a dense depth map. First, visual features and lidar depth points are projected to a unit sphere that is centered at the camera. Depth points are then downsampled and stored in a 2D K-D tree based on the two angular coordinates. Next, we search for the nearest three depth points on the sphere for a visual feature by searching the 2D K-D tree using the coordinates of the visual feature. An illustration of this process is shown in Figure 5.2.

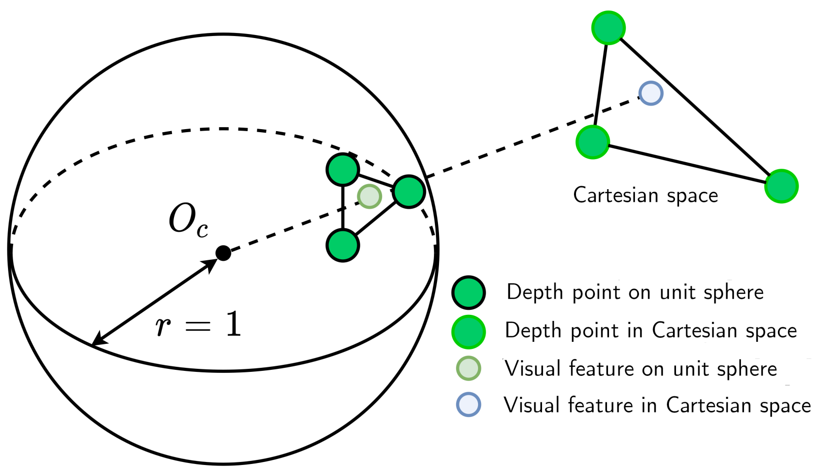


Figure 5.2: Depth association to visual features using lidar depth map [6]

Let $\hat{\mathbf{X}}_j^k, j \in \{1,\, 2,\, 3\}$ be the coordinates of the three points in the unit sphere centered at the camera and let $\mathbf{X}_i^k$ be the coordinates of feature i. Then, depth can be computed as follows:

$$(\mathbf{X}_i^k - \hat{\mathbf{X}}_1^k)((\hat{\mathbf{X}}_1^k - \hat{\mathbf{X}}_2^k) \times (\hat{\mathbf{X}}_1^k - \hat{\mathbf{X}}_3^k)) = 0 \tag{5.1}$$

5.2.2 VIO Frame to Frame Constraints

The lidar odometry module follows an implementation similar to version 1.1. However, considering the observability of VIO, frame to frame motion of the VIO is incorporated as constraints in the optimization problem. VIO has observable velocities, and frame to frame motion for small time steps can be considered as velocity measurements. Recall equations 4.11 and 4.12 for frame to frame motion of VIO transformed to lidar frame $\{L\}$. Let the variance of frame to frame translation and rotation of VIO estimate be $\sigma^2_{t,ff}$ and $\sigma^2_{R,ff}$ respectively. $\mathbf{T}^L_{k,k+1} = \begin{bmatrix}\mathbf{p}^L_{(k,k+1)} & \mathbf{q}^L_{(k,k+1)}\end{bmatrix}$ is the motion to be solved. The constraints for the motion using the VIO frame to frame pose can be formulated as:

$$\begin{bmatrix} (\mathbf{p}^L_{(k,k+1)} - {}^v\mathbf{p}^L_{(k,k+1)})/\sigma^2_{t,ff} \\ ({\mathbf{q}^L_{(k,k+1)}}^{-1} \otimes {}^v\mathbf{q}^L_{(k,k+1)})/\sigma^2_{R,ff} \end{bmatrix} = 0 \tag{5.2}$$

These constraints are added to the optimization problem together with lidar edge and planar feature constraints and solved using Ceres solver to estimate the motion during the lidar sweep.

5.2.3 VIO Global Orientation Constraints

Similarly, lidar mapping constraints are adjusted according to the observability of VIO motion. In VI-LOAM version 1.1, the global 6-DoF VIO pose, both position and orientation, was incorporated as constraints. However, only the roll and pitch orientations are the observable measurements from the VIO global state. Therefore, constraints were adjusted only to incorporate roll and pitch orientations.

Recall equation 4.13 for the quaternion, ${}^v\mathbf{q}^W_{(k+1)}$, representing the global VIO orientation in lidar world frame $\{L\}$. Recall that, for this study, the lidar frame $\{L\}$ overlaps with the world frame $\{W\}$, i.e., $\{L\} \equiv \{W\}$. Let the variance of orientation estimate to be σ^2_R and motion to be solved be $\mathbf{T}^W_{k+1} = \begin{bmatrix}\mathbf{p}^W_{(k+1)} & \mathbf{q}^W_{(k+1)}\end{bmatrix}$. In order to compute the difference between roll and pitch orientation difference, a new quaternion is derived by combining the yaw of the motion to be solved and VIO roll and pitch. This derivation is initialized by converting the existing quaternions into their respective Euler angle representation. Let $[\psi_v, \theta_v, \phi_v]$ be Euler angles for the global VIO orientation quaternion ${}^{v}\mathbf{q}^{W}_{(k+1)}$ and let $[\psi, \theta, \phi]$ be Euler angles for the quaternion of the motion to be solved. Therefore, Euler angle representation for the new combined orientation is $[\psi, \theta_v, \phi_v]$. The quaternion for these Euler angles can be computed as follows:

$$
{}^{Comb}\mathbf{q}^{W}_{(k+1)} = \begin{bmatrix} \cos\psi/2 \\ 0 \\ 0 \\ \sin\psi/2 \end{bmatrix} \begin{bmatrix} \cos\theta_v/2 \\ 0 \\ \sin\theta_v/2 \\ 0 \end{bmatrix} \begin{bmatrix} \cos\phi_v/2 \\ \sin\phi_v/2 \\ 0 \\ 0 \end{bmatrix} \tag{5.3}
$$

Consequently, the constraint for the motion using the VIO roll and pitch can be formulated as:

$$
\left[(\mathbf{q}^{W}_{(k+1)}{}^{-1}) \otimes ({}^{Comb}\mathbf{q}^{W}_{(k+1)}) / \sigma^2_{R} \right] = 0 \tag{5.4}
$$

5.2.4 Global Fusion

The global fusion module momentarily corrects the odometry from the VI-LOAM version 1.2 system when GPS signals are available. GPS provides absolute position information without any drift errors. This ensures that the drift of the VI-LOAM estimation does not accumulate and provides large erroneous results. To this end, a non-linear optimization problem is solved using GPS absolute measurement constraints and VI-LOAM version 1.2 estimation constraints. The factor graph corresponds to this optimization problem is shown in Figure 5.3.

GPS receives absolute longitude, latitude, and altitude measurements at 1 Hz. The longitude, latitude, and altitude can be converted to x, y, and z coordinates with respect to the earth frame, $\{G\}$. We can denote these coordinates as ${}^{GPS}\mathbf{p}^{G}_{(k+1)}$. Let

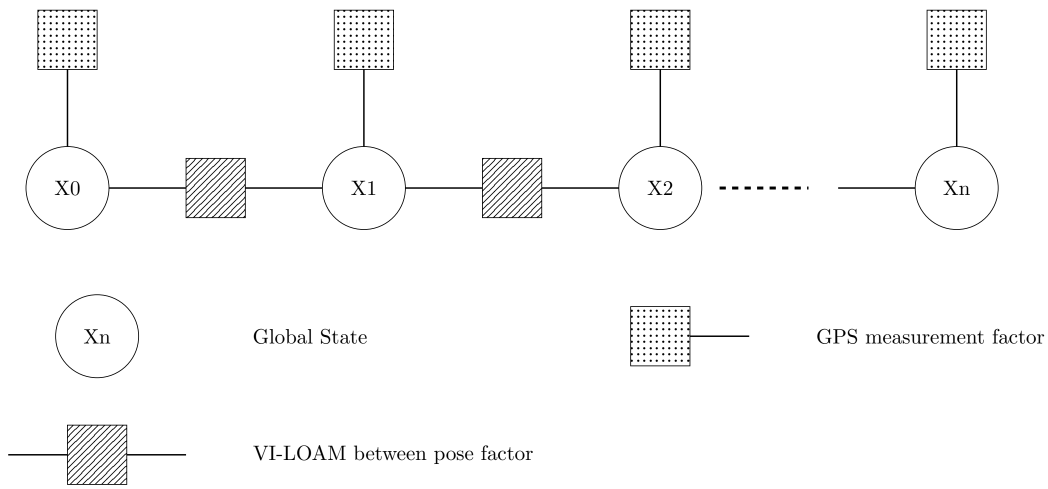


Figure 5.3: Factor graph of the global fusion optimization

the pose to be solved be $\mathbf{T}^G_{k+1} = \begin{bmatrix} \mathbf{p}^G_{(k+1)} & \mathbf{q}^G_{(k+1)} \end{bmatrix}$. Let the variance of the position measurement of GPS be ${}^{GPS}\sigma^2_t$. Then the constraints from GPS is formulated as:

$$(\mathbf{p}^G_{(k+1)} - {}^{GPS}\mathbf{p}^G_{(k+1)})/{}^{GPS}\sigma^2_t = 0. \tag{5.5}$$

On the other hand, odometry from VI-LOAM version 1.2 is formulated as a pose difference between two updates at 1 Hz. We can derive the following equations for the VI-LOAM pose difference between updates using the equation 4.9 and equation 4.10.

$$\mathbf{q}^W_{(k,k+1)} = (\mathbf{q}^W_k)^{-1} \otimes \mathbf{q}^W_{k+1} \tag{5.6}$$

$$\mathbf{p}^W_{(k,k+1)} = \mathbf{q}^{W-1}_k \otimes (\mathbf{p}^W_k - \mathbf{p}^W_{k+1}) \tag{5.7}$$

Let already solved previous pose of the global fusion be $\mathbf{T}^G_k$. Let the variance of the translation and rotation between updates for the VI-LOAM estimate be $\sigma^2_{t,\delta}$ and $\sigma^2_{R,\delta}$, respectively. Then constraints from the VI-LOAM is given by 5.8.

$$\begin{bmatrix} (\mathbf{p}^G_{(k+1)} - \mathbf{p}^G_{(k)}) - (\mathbf{p}^W_{(k,k+1)})/\sigma^2_{t,\delta} \\ (\mathbf{q}^G_{(k+1)}{}^{-1} \otimes \mathbf{q}^G_{(k)})^{-1} \otimes \mathbf{q}^W_{(k+1,k)})^{-1}/\sigma^2_{R,\delta} \end{bmatrix} = 0 \tag{5.8}$$

These constraints are formulated for all the measurements within a sliding window to jointly estimate global pose. Note that GPS constraints are only for the position. Therefore, orientation of the global pose degenerates to the orientation of the VI-LOAM version 1.2 system.

To ensure robust performance, GPS degradation cases are assessed. If the variance of the GPS signal is high, indicating an erroneous signal, the corresponding GPS measurement will be discarded and the GPS measurement constrain will not be applied in the optimization process. Consequently, the system is naturally degraded to VI-LOAM estimation if GPS is not available or provides erroneous measurements.

5.3 Results

To evaluate the proposed VI-LOAM version 1.2 implementation, it is required to have a dataset with different GPS availability regions. Neither two datasets capture using in-house AI4L sensor payload nor publicly available KITTI benchmark dataset satisfy this condition. In contrast, LVI-SAM dataset [6] contains different regions of GPS availability which is ideal for evaluating the proposed GPS aided implementation and its performance for different GPS scenarios.

5.3.1 LVI-SAM Dataset

The LVI-SAM dataset [6] consists of three datasets, with each one having a duration of more than 30 minutes. It includes grayscale camera images, laser point clouds, GPS measurements, and IMU data. The sensors they have used are as follows:

- Velodyne VLP-16 lidar updates at 10 Hz
- FLIR BFS-U3-04S2M-CS camera,1.4 Megapixels updates at 10 Hz
- MicroStrain 3DM-GX5-25 IMU updates at 100 Hz
- Reach RS+ GPS updates at 5 Hz

We have considered two LVI-SAM datasets named Handheld and Jackal for our evaluation. The Handheld dataset is gathered by carrying the sensor payload around in an open field, whereas the Jackal dataset is gathered by mounting the sensor payload on a Clearpath Jackal unmanned ground vehicle (UGV). Both datasets begin and end at the same position.

5.3.2 LVI-SAM Dataset Results

The proposed GPS aided VI-LOAM system was evaluated with two sequences of the LVI-SAM dataset. This dataset provides positioning information using a reach RTK GPS with 5 Hz update rate. In that, there are regions with GPS loss with over 15 m GPS position error. Accordingly, the path of an LVI-SAM dataset can be divided into three regions based on the GPS availability: regions without GPS error, regions with random GPS error spikes, and regions with continuous GPS loss. The comparison results of the GPS aided VI-LOAM path and GPS position for LVI-SAM dataset 1 and 2 are illustrated in Figure 5.4 and 5.5. According to them, GPS aided VI-LOAM has corrected the drift errors of stand-alone VI-LOAM and follows the ground truth path.

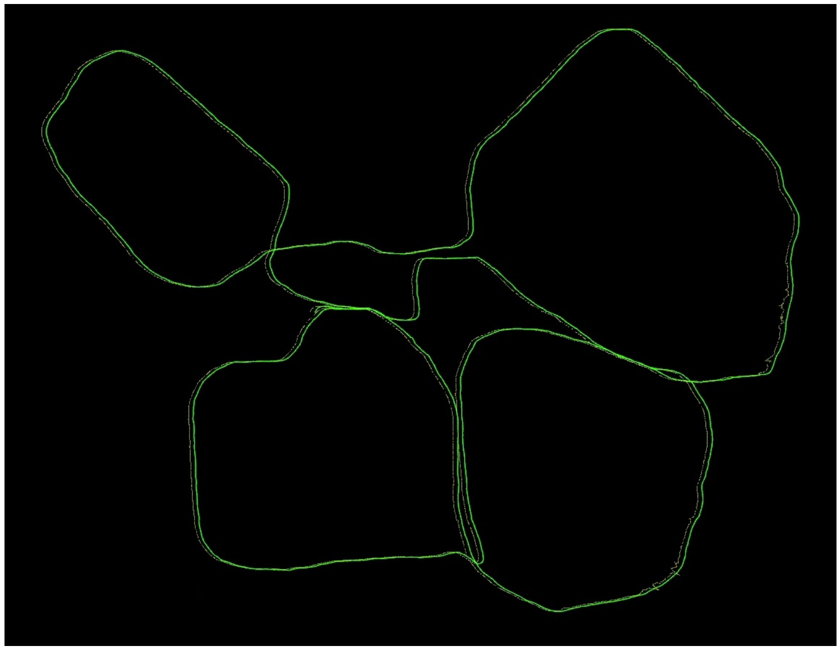


Figure 5.4: Estimated path and GPS position for the LVI-SAM dataset #1. Yellow track is for GPS position and Green track is for GPS aided VI-LOAM estimation.

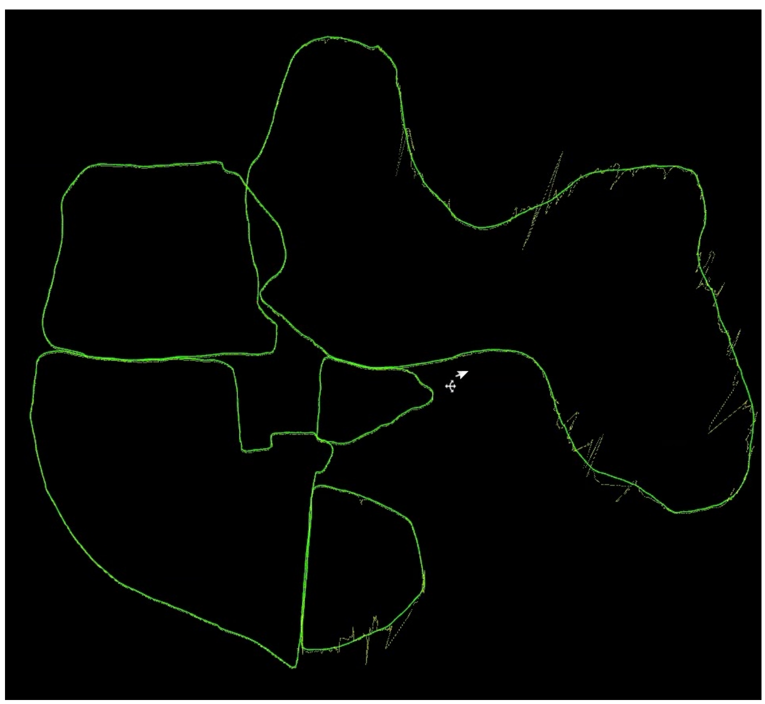


Figure 5.5: Estimated path and GPS position for the LVI-SAM dataset #2. Yellow track is for GPS position and Green track is for GPS aided VI-LOAM estimation.

Additionally, the performance of proposed method was evaluated for all the GPS regions. In the LVI-SAM dataset, when GPS was erroneous it only affected the GPS altitude estimation. Since the data were captured on a 2D terrain, the nominal altitude at each location was taken as the ground truth to compute the error. Figure 5.6 illustrates the estimated path of GPS aided VI-LOAM when random GPS errors occur. The estimated path is unaffected by these momentary GPS errors. On the other hand, when the GPS signal provides continuous errors, as shown in Figure 5.7, the estimated path has drifted in that direction. However, it has managed to keep the error to a lower value than the GPS.

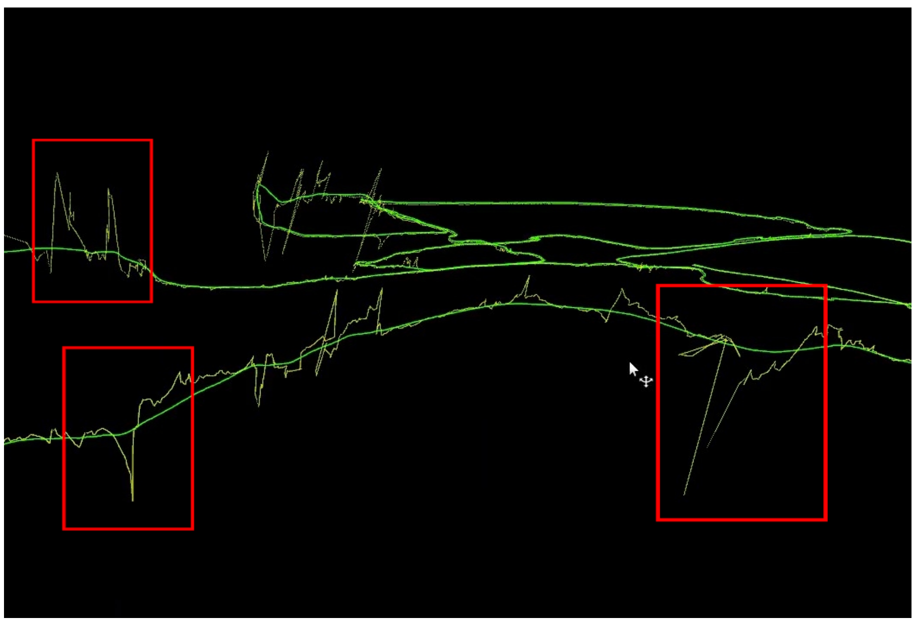


Figure 5.6: Estimated path when random GPS error spikes occur.Yellow track is for GPS position and Green track is for GPS aided VI-LOAM estimation.

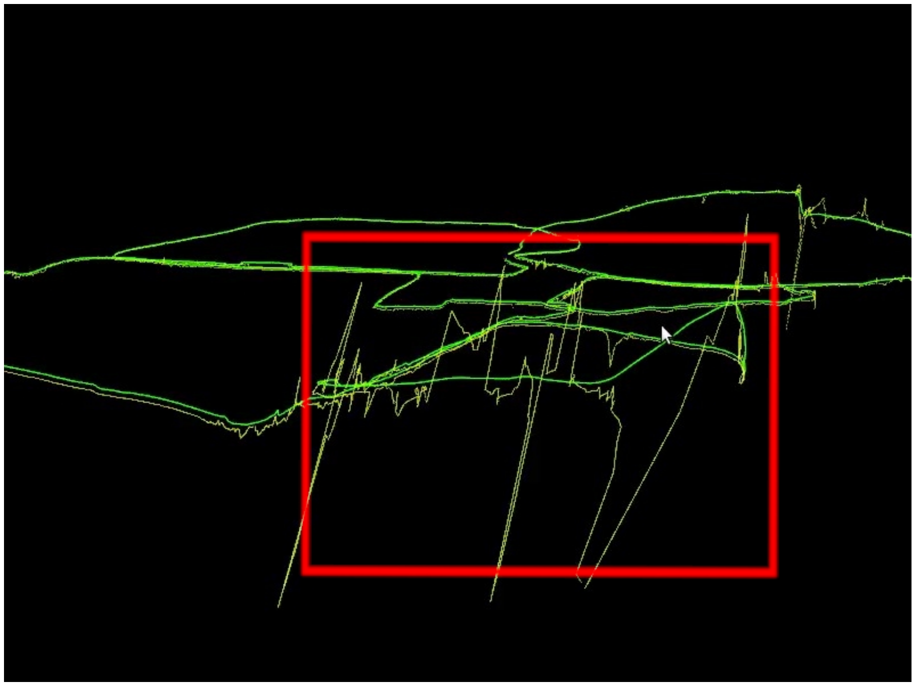


Figure 5.7: Estimated path in the region of continuous GPS errors.Yellow track is for GPS position and Green track is for GPS aided VI-LOAM estimation.

Maximum RMS position error for different GPS region is summarized in Table 5.1. According to them, VI-LOAM version 1.2 has kept the position error under 2.5 m when GPS corrections are available. Also, the system only has around 3 m position errors when sudden GPS losses occur, resulting in GPS spikes. Even when GPS is lost for a continuous period, the maximum error has gone only up to 9.7 m. Therefore, the GPS aided VI-LOAM version 1.2 system meets the required safety accuracy levels by maintaining the position error under 10 m.

5.4 Summary

This chapter introduced the improvements carried out to develop VI-LOAM version 1.2 and the integration of GPS sensor information with that system. The performance of the proposed GPS aided VI-LOAM system was evaluated with the LVI-SAM dataset

Table 5.1: Performance of GPS aided VI-LOAM version 1.2 for different regions of GPS availability

Dataset	GPS signal error	Maximum RMS Error		
		No GPS errors	Random GPS error spikes	Continuous GPS loss
LVI Dataset 1-Handheld	5-10 m	2.35 m	3 m	5 m
LVI Dataset 2-Jackal	15 m	1.3 m	3 m	9.7 m

for different GPS scenarios. According to the results, the proposed GPS aided VI-LOAM manages to correct the drift errors in the stand-alone VI-LOAM system and rectify the GPS degradation situations such as GPS loss and multipath errors. Moreover, the GPS aided VI-LOAM version 1.2 system meets the UAV safety regulations imposed by Transport Canada. The next chapter discusses the accomplishment of the objectives of this thesis work and future directives addressing some of the limitations of the proposed system.

Chapter 6

Conclusion and Future Work

The focus of this research study was to develop a novel visual, inertial, and lidar combined navigation system aided with GPS for UAV-based parcel delivery applications. The navigation system is developed while addressing the identified drawbacks of state-of-the-art navigation systems. This chapter summarizes the accomplishment of each objective with the conclusions. Future directions are then presented based on the observations and conclusions.

6.1 Research Summary Based on Objective I

The first objective of this study was to develop a novel robust visual, lidar and inertial integrated odometry and mapping system for UAV navigation. Chapter 4 has presented a novel multi-sensory architecture VI-LOAM 1.1 that was implemented by effectively combining already existing robust navigation packages VINS-mono and ALOAM. Improvements for this architecture was done in Chapter 5. This method uses tightly-coupled IMU preintegration and uses modern optimization library Google Ceres for the implementation. The proposed architecture was evaluated with KITTI benchmark dataset and in-house AI4L payload dataset. The results indicate that proposed method has higher accuracy than state-of-the-art vision-based and lidar-based navigation methods.

6.2 Research summary Based on Objective II

The second objective of this study was to implement a GPS aided multi-sensory navigation system for UAV navigation. The visual, lidar and inertial integrated navigation system was extended by incorporating GPS as a global pose correction in Chapter 5. To this end, GPS corrects the drift errors from the VI-LOAM estimator as and when the GPS signals are available. The proposed GPS aided VI-LOAM method was evaluated with LVI-SAM dataset with different GPS scenarios. According to the results, GPS aided VI-LOAM method rectify the errors by GPS failure cases such as multipath errors and GPS loss.

6.3 Research Summary Based on Objective III

The third objective of this study was to compare the proposed method with state-of-the-art UAV navigation methods and evaluate the system with UAV safety regulations. The proposed multi-sensory navigation system and state-of-the-art UAV navigation methods, VINS-mono and ALOAM, were tested and evaluated on KITTI odometry benchmark dataset and AI4L payload data in Chapter 4. Findings indicate that the proposed system has significant improvement for the position drift compared to the other methods. The extended multi-sensory navigation system with GPS was evaluated with LVI-SAM online dataset with different GPS regions. The results indicate that the system managed to keep the maximum RMS Error under 10 m even when GPS is erroneous for continuous period of time. Therefore, the proposed system meets the required safety regulations by maintaining the position error under 10 m.

6.4 Publications

A part of this work is presented at the IEEE Newfoundland Electrical and Computer Engineering Conference (NECEC) 2021 under the title "Numerical Study of GPS Aided Visual Lidar Odometry and Mapping (VLOAM) for Safety Regulatory Compliance." A journal paper from this study is submitted to the IEEE Transactions on Automation Science and Engineering, 2022 with the title "Review of Navigation Methods and Implementation of GPS/VLOAM Solution for UAV-Based Parcel Delivery", and it is under review at the time of submitting this thesis.

6.5 Future Directives

This work has obtained successful results for the proposed navigation system for selected datasets. However, an exhaustive validation needs to be done through systematic field trials to confirm its applicability under different scenarios and its robustness against unplanned weather conditions before real-world application.

The proposed method has some limitations that need to be addressed in future iterations. The transformations between lidar, camera, and IMU need to be calibrated by the user for each payload. Therefore, the accuracy of the proposed method relies on this sensor calibration provided by the user. Also, the proposed method is limited to navigation and does not include any obstacle detection or avoidance techniques. Additionally, using GPS as a global correction can be problematic as GPS is prone to signal errors and can be manipulated by others.

The proposed system can be extended to carry out self-calibration from approximate sensor calibration values. Therefore, over-dependence on user input can be avoided. This navigation system can be equipped with obstacle detection and avoidance techniques using the generated map [141] to tackle emergency situations and improve the safety of navigation. Additionally, detecting dynamic obstacles and updating the map without them [142] may improve the mapping accuracy. Future work will target using 3D digital elevation maps to provide global updates in addition to GPS as they are immune from signal errors and spoofing events.

Bibliography

[1] J. D. Gautier, *GPS/INS generalized evaluation tool (GIGET) for the design and testing of integrated navigation systems.* Stanford university, 2003.

[2] J. Zhang and S. Singh, "Laser–visual–inertial odometry and mapping with high robustness and low drift," *Journal of Field Robotics*, vol. 35, no. 8, pp. 1242–1264, 2018.

[3] J. Zhang, "Vision-lidar odometry and mapping [la] [v-loam]," Online, July 2020, accessed: 05-June-2021.

[4] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time." in *Robotics: Science and Systems*, vol. 2, no. 9, 2014.

[5] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[6] T. Shan, B. Englot, C. Ratti, and D. Rus, "Lvi-sam: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping," in *2021 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 2021, pp. 5692–5698.

[7] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle routing problems for drone delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 70–85, 2016.

[8] I. Amazon.com, "Amazon prime air," Online, Dec 2016, accessed: 10-Mar.-2021.

[9] J. Stewart, "Google tests drone deliveries in project wing trials," Online, Aug 2014, accessed: 10-Mar.-2021.

[10] D. I. GmbH, "Dhl express launches its first regular fully-automated and intelligent urban drone delivery service," Online, May 2019, accessed: 10-Mar.-2021.

[11] W.-C. Chiang, Y. Li, J. Shang, and T. L. Urban, "Impact of drone delivery on sustainability and cost: Realizing the uav potential through vehicle routing optimization," *Applied energy*, vol. 242, pp. 1164–1175, 2019.

[12] R. Knight, "Drone delivery canada to begin cargo delivery to georgina island," Online, Sep. 2020, accessed: 27-Mar.-2021.

[13] A. Pasztor, "To build a plane that can fly itself, start with a pilot in the cockpit," Online, Aug. 2020, accessed: 27-Mar.-2021.

[14] S. Babych, "Research project connecting remote communities with covid-19 test kits using drones," Online, Oct. 2020, accessed: 27-Mar.-2021.

[15] V. Yaskevich, "Medical drones. the future of emergency medicine," Online, Nov. 2020, accessed: 01-June-2021.

[16] sUAS News, "Ums skeldar announces first naval deployment of full atol capability of v-200 uav platform," Online, Feb. 2021, accessed: 27-Mar.-2021.

[17] K. T. San, S. J. Mun, Y. H. Choe, and Y. S. Chang, "Uav delivery monitoring system," in *MATEC Web of Conferences*, vol. 151. EDP Sciences, 2018, p. 04011.

[18] C. Stöcker, R. Bennett, F. Nex, M. Gerke, and J. Zevenbergen, "Review of the current state of uav regulations," *Remote sensing*, vol. 9, no. 5, p. 459, 2017.

[19] M. R. Haque, M. Muhammad, D. Swarnaker, and M. Arifuzzaman, "Autonomous quadcopter for product home delivery," in *2014 International Conference on Electrical Engineering and Information & Communication Technology*. IEEE, 2014, pp. 1–5.

[20] G. H. Elkaim, F. A. P. Lie, and D. Gebre-Egziabher, "Principles of guidance, navigation, and control of uavs," *Handbook of unmanned aerial vehicles*, pp. 347–380, 2015.

[21] C.-S. Y. C.-S. Yoo and I.-K. A. I.-K. Ahn, "Low cost gps/ins sensor fusion system for uav navigation," in *Digital Avionics Systems Conference, 2003. DASC'03. The 22nd*, vol. 2. IEEE, 2003, pp. 8–A.

[22] J. Wendel, O. Meister, C. Schlaile, and G. F. Trommer, "An integrated gps/mems-imu navigation system for an autonomous helicopter," *Aerospace science and technology*, vol. 10, no. 6, pp. 527–533, 2006.

[23] K. Li, J. Zhao, X. Wang, and L. Wang, "Federated ultra-tightly coupled gps/ins integrated navigation system based on vector tracking for severe jamming environment," *IET Radar, Sonar & Navigation*, vol. 10, no. 6, pp. 1030–1037, 2016.

[24] J. N. Gross, Y. Gu, and M. B. Rhudy, "Robust uav relative navigation with dgps, ins, and peer-to-peer radio ranging," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 3, pp. 935–944, 2015.

[25] J. H. Ryu, G. Gankhuyag, and K. T. Chong, "Navigation system heading and position accuracy improvement through gps and ins data fusion," *Journal of Sensors*, vol. 2016, 2016.

[26] K.-W. Chiang, T. T. Duong, and J.-K. Liao, "The performance analysis of a real-time integrated ins/gps vehicle navigation system with abnormal gps measurement elimination," *Sensors*, vol. 13, no. 8, pp. 10 599–10 622, 2013.

[27] M. Park and Y. Gao, "Error and performance analysis of mems-based inertial sensors with a low-cost gps receiver," *Sensors*, vol. 8, no. 4, pp. 2240–2261, 2008.

[28] A. Allouch, A. Koubaa, M. Khalgui, and T. Abbes, "Qualitative and Quantitative Risk Analysis and Safety Assessment of Unmanned Aerial Vehicles Missions over the Internet," *IEEE Access*, vol. 7, no. April, pp. 53 392–53 410, 2019.

[29] L. Arreola, A. M. De Oca, A. Flores, J. Sanchez, and G. Flores, "Improvement in the uav position estimation with low-cost gps, ins and vision-based system: Application to a quadrotor uav," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2018, pp. 1248–1254.

[30] G. Conte and P. Doherty, "An integrated uav navigation system based on aerial image matching," in *2008 IEEE Aerospace Conference*. IEEE, 2008, pp. 1–10.

[31] R. Ye, S. Redfield, and H. Liu, "High-precision indoor uwb localization: Technical challenges and method," in *2010 IEEE International Conference on Ultra-Wideband*, vol. 2. IEEE, 2010, pp. 1–4.

[32] J. Tiemann, F. Schweikowski, and C. Wietfeld, "Design of an uwb indoor-positioning system for uav navigation in gnss-denied environments," in *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2015, pp. 1–7.

[33] J. P. Queralta, C. M. Almansa, F. Schiano, D. Floreano, and T. Westerlund, "Uwb-based system for uav localization in gnss-denied environments: Characterization and dataset," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 4521–4528.

[34] E. B. Quist, *UAV navigation and radar odometry*. Brigham Young University, 2015.

[35] D. Vivet, P. Checchin, and R. Chapuis, "Localization and mapping using only a rotating fmcw radar sensor," *Sensors*, vol. 13, no. 4, pp. 4527–4552, 2013.

[36] A. Scannapieco, A. Renga, G. Fasano, and A. Moccia, "Ultralight radar for small and micro-uav navigation," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 42, p. 333, 2017.

[37] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[38] Y. Lu, Z. Xue, G.-S. Xia, and L. Zhang, "A survey on vision-based uav navigation," *Geo-spatial information science*, vol. 21, no. 1, pp. 21–32, 2018.

[39] F. Santoso, M. A. Garratt, and S. G. Anavatti, "Visual–inertial navigation systems for aerial robotics: Sensor fusion and technology," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 260–275, 2016.

[40] J. Lin and F. Zhang, "Loam livox: A fast, robust, high-precision lidar odometry and mapping package for lidars of small fov," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 3126–3131.

[41] J. Zhang, M. Kaess, and S. Singh, "Real-time depth enhanced monocular odometry," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 4973–4980.

[42] M. Mostafa, S. Zahran, A. Moussa, N. El-Sheimy, and A. Sesay, "Radar and visual odometry integrated system aided navigation for uavs in gnss denied environment," *Sensors*, vol. 18, no. 9, p. 2776, 2018.

[43] G. Balamurugan, J. Valarmathi, and V. Naidu, "Survey on uav navigation in gps denied environments," in *2016 International conference on signal processing, communication, power and embedded system (SCOPES)*. IEEE, 2016, pp. 198–204.

[44] X. Zuo, Y. Yang, P. Geneva, J. Lv, Y. Liu, G. Huang, and M. Pollefeys, "Lic-fusion 2.0: Lidar-inertial-camera odometry with sliding-window plane-feature tracking," *arXiv preprint arXiv:2008.07196*, 2020.

[45] J. Zhang and S. Singh, "Low-drift and real-time lidar odometry and mapping," *Autonomous Robots*, vol. 41, no. 2, pp. 401–416, 2017.

[46] J. Wang, M. Garratt, A. Lambert, J. J. Wang, S. Han, and D. Sinclair, "Integration of gps/ins/vision sensors to navigate unmanned aerial vehicles," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 37, no. part B1, pp. 963–969, 2008.

[47] L. Chang, X. Niu, T. Liu, J. Tang, and C. Qian, "Gnss/ins/lidar-slam integrated navigation system based on graph optimization," *Remote Sensing*, vol. 11, no. 9, p. 1009, 2019.

[48] T. Qin, S. Cao, J. Pan, and S. Shen, "A General Optimization-based Framework for Global Pose Estimation with Multiple Sensors," 2019. [Online]. Available: http://arxiv.org/abs/1901.03642

[49] T.-M. Nguyen, S. Yuan, M. Cao, T. H. Nguyen, and L. Xie, “Viral slam: Tightly coupled camera-imu-uwb-lidar slam,” *arXiv preprint arXiv:2105.03296*, 2021.

[50] S. Agarwal, K. Mierle, and Others, “Ceres solver,” http://ceres-solver.org.

[51] F. Dellaert, “Factor graphs and gtsam: A hands-on introduction,” Georgia Institute of Technology, Tech. Rep., 2012.

[52] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, “Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5135–5142.

[53] L. Weber, *International Civil Aviation Organization (ICAO)*. Kluwer Law International BV, 2021.

[54] F. A. Regulations, “Federal aviation administration,” *Washington, DC*, 1990.

[55] T. Canada, “Canadian aviation regulations,” Online, April 2022, accessed: 10-April-2022.

[56] N. El-Sheimy, H. Hou, and X. Niu, “Analysis and modeling of inertial sensors using allan variance,” *IEEE Transactions on instrumentation and measurement*, vol. 57, no. 1, pp. 140–149, 2007.

[57] W. Stempfhuber and M. Buchholz, “A precise, low-cost rtk gnss system for uav applications,” *Proc. of Unmanned Aerial Vehicle in Geomatics, ISPRS*, 2011.

[58] G. Zhang and L.-T. Hsu, “Intelligent gnss/ins integrated navigation system for a commercial uav flight control system,” *Aerospace Science and Technology*, vol. 80, pp. 368–380, 2018.

[59] S. Ragothaman, M. Maaref, and Z. M. Kassas, “Multipath-optimal uav trajectory planning for urban uav navigation with cellular signals,” in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*. IEEE, 2019, pp. 1–6.

[60] B. Barshan and H. F. Durrant-Whyte, “Inertial navigation systems for mobile robots,” *IEEE transactions on robotics and automation*, vol. 11, no. 3, pp. 328–342, 1995.

[61] T. T. Nwe, T. Htike, K. M. Mon, Z. M. Naing, and Y. M. Myint, “Application of an inertial navigation system to the quad-rotor uav using mems sensors,” *Engineering and Technology*, vol. 42, pp. 578–582, 2008.

[62] C. Forster, M. Pizzoli, and D. Scaramuzza, “Svo: Fast semi-direct monocular visual odometry,” in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 15–22.

[63] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[64] I. Cvišić, J. Ćesić, I. Marković, and I. Petrović, "Soft-slam: Computationally efficient stereo visual simultaneous localization and mapping for autonomous unmanned aerial vehicles," *Journal of field robotics*, vol. 35, no. 4, pp. 578–595, 2018.

[65] M. Velas, M. Spanel, and A. Herout, "Collar line segments for fast odometry estimation from velodyne point clouds," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 4486–4495.

[66] J. Lin and F. Zhang, "Loam_livox: A fast, robust, high-precision LiDAR odometry and mapping package for LiDARs of small FoV," 2019. [Online]. Available: http://arxiv.org/abs/1909.06700

[67] O. J. Woodman, "An introduction to inertial navigation," University of Cambridge, Computer Laboratory, Tech. Rep., 2007.

[68] Y. N. Korkishko, V. Fedorov, V. Prilutskiy, V. Ponomarev, I. Fedorov, S. Kostritskii, I. Morev, D. Obuhovich, S. Prilutskiy, A. Zuev *et al.*, "High-precision inertial measurement unit imu-5000," in *2018 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*. IEEE, 2018, pp. 1–4.

[69] G. Falco, M. Pini, and G. Marucco, "Loose and tight gnss/ins integrations: Comparison of performance assessed in real urban scenarios," *Sensors*, vol. 17, no. 2, p. 255, 2017.

[70] T. Qin, P. Li, and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[71] M. Barczyk and A. F. Lynch, "Invariant observer design for a helicopter uav aided inertial navigation system," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 3, pp. 791–806, 2013.

[72] T. Taketomi, H. Uchiyama, and S. Ikeda, "Visual slam algorithms: a survey from 2010 to 2016," *IPSJ Transactions on Computer Vision and Applications*, vol. 9, no. 1, pp. 1–11, 2017.

[73] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European conference on computer vision*. Springer, 2014, pp. 834–849.

[74] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2017.

[75] Z. Yang and S. Shen, "Monocular visual–inertial state estimation with online initialization and camera–imu extrinsic calibration," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 39–51, 2016.

[76] S. Shen, N. Michael, and V. Kumar, "Tightly-coupled monocular visual-inertial fusion for autonomous flight of rotorcraft mavs," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 5303–5310.

[77] S. Heo, J. Cha, and C. G. Park, "Ekf-based visual inertial navigation using sliding window nonlinear optimization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 7, pp. 2470–2479, 2018.

[78] B. Herissé, T. Hamel, R. Mahony, and F.-X. Russotto, "Landing a vtol unmanned aerial vehicle on a moving platform using optical flow," *IEEE Transactions on robotics*, vol. 28, no. 1, pp. 77–89, 2011.

[79] L. Huang, J. Song, C. Zhang, and G. Cai, "Observable modes and absolute navigation capability for landmark-based imu/vision navigation system of uav," *Optik*, vol. 202, p. 163725, 2020.

[80] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual–inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.

[81] A. Vu, A. Ramanandan, A. Chen, J. A. Farrell, and M. Barth, "Real-time computer vision/dgps-aided inertial navigation system for lane-level vehicle navigation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 899–913, 2012.

[82] M. Bryson, M. Johnson-Roberson, and S. Sukkarieh, "Airborne smoothing and mapping using vision and inertial sensors," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 2037–2042.

[83] A. R. Vetrella, G. Fasano, and D. Accardo, "Attitude estimation for cooperating uavs based on tight integration of gnss and vision measurements," *Aerospace Science and Technology*, vol. 84, pp. 966–979, 2019.

[84] S. Cao, X. Lu, and S. Shen, "Gvins: Tightly coupled gnss-visual-inertial for smooth and consistent state estimation," *arXiv preprint arXiv:2103.07899*, 2021.

[85] D. Scaramuzza, F. Fraundorfer, and M. Pollefeys, "Closing the loop in appearance-guided omnidirectional visual odometry by using vocabulary trees," *Robotics and Autonomous Systems*, vol. 58, no. 6, pp. 820–827, 2010.

[86] D. Droeschel, J. Stückler, and S. Behnke, "Local multi-resolution representation for 6d motion estimation and mapping with a continuously rotating 3d laser scanner," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 5221–5226.

[87] M. Joerger and B. Pervan, "Range-domain integration of gps and laser-scanner measurements for outdoor navigation," in *Proceedings of the 19th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2006)*, 2006, pp. 1115–1123.

[88] M. Hentschel, O. Wulf, and B. Wagner, "A gps and laser-based localization for urban and non-urban outdoor environments," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 149–154.

[89] A. Soloviev, D. Bates, and F. Van Graas, "Tight coupling of laser scanner and inertial measurements for a fully autonomous relative navigation solution," *Navigation*, vol. 54, no. 3, pp. 189–205, 2007.

[90] S. Liu, M. M. Atia, T. B. Karamat, and A. Noureldin, "A lidar-aided indoor navigation system for ugvs," *The Journal of Navigation*, vol. 68, no. 2, pp. 253–273, 2015.

[91] N. Li, L. Guan, and Y. Gao, "A seamless indoor and outdoor low-cost integrated navigation system based on lidar/gps/ins," in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*. IEEE, 2020, pp. 1–6.

[92] A. Soloviev, "Tight coupling of gps, laser scanner, and inertial measurements for navigation in urban environments," in *2008 IEEE/ION Position, Location and Navigation Symposium*. IEEE, 2008, pp. 511–525.

[93] M. Jabbour and P. Bonnifait, "Backing up gps in urban areas using a scanning laser," in *2008 IEEE/ION Position, Location and Navigation Symposium*. IEEE, 2008, pp. 505–510.

[94] J. Zhang and S. Singh, "Visual-lidar odometry and mapping: Low-drift, robust, and fast," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 2174–2181, 2015.

[95] S. Yun, Y. J. Lee, and S. Sung, "Imu/vision/lidar integrated navigation system in gnss denied environments," in *2013 IEEE Aerospace Conference*. IEEE, 2013, pp. 1–10.

[96] X. Zuo, P. Geneva, W. Lee, Y. Liu, and G. Huang, "Lic-fusion: Lidar-inertial-camera odometry," *arXiv preprint arXiv:1909.04102*, 2019.

[97] S. Scherer, J. Rehder, S. Achar, H. Cover, A. Chambers, S. Nuske, and S. Singh, "River mapping from a flying robot: state estimation, river detection, and obstacle mapping," *Autonomous Robots*, vol. 33, no. 1, pp. 189–214, 2012.

[98] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit *et al.*, "Fastslam: A factored solution to the simultaneous localization and mapping problem," *Aaai/iaai*, vol. 593598, 2002.

[99] H. B. Khamseh, S. Ghorbani, and F. Janabi-Sharifi, "Unscented kalman filter state estimation for manipulating unmanned aerial vehicles," *Aerospace Science and Technology*, vol. 92, pp. 446–463, 2019.

[100] S. Thrun, "Probabilistic robotics," *Communications of the ACM*, vol. 45, no. 3, pp. 52–57, 2002.

[101] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation." Georgia Institute of Technology, 2015.

[102] L. Semeniuk and A. Noureldin, "Bridging gps outages using neural network estimates of ins position and velocity errors," *Measurement science and technology*, vol. 17, no. 10, p. 2783, 2006.

[103] A. Noureldin, A. El-Shafie, and M. Bayoumi, "Gps/ins integration utilizing dynamic neural networks for vehicular navigation," *Information fusion*, vol. 12, no. 1, pp. 48–57, 2011.

[104] Y. Zhou, J. Wan, Z. Li, and Z. Song, "Gps/ins integrated navigation with bp neural network and kalman filter," in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2017, pp. 2515–2520.

[105] J. Fayyad, M. A. Jaradat, D. Gruyer, and H. Najjaran, "Deep learning sensor fusion for autonomous vehicle perception and localization: A review," *Sensors*, vol. 20, no. 15, p. 4220, 2020.

[106] S. Yin, X. Zhu, J. Qiu, and H. Gao, "State estimation in nonlinear system using sequential evolutionary filter," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 6, pp. 3786–3794, 2016.

[107] M. A. K. Jaradat and M. F. Abdel-Hafez, "Enhanced, delay dependent, intelligent fusion for ins/gps navigation system," *IEEE Sensors Journal*, vol. 14, no. 5, pp. 1545–1554, 2014.

[108] M. K. Al-Sharman, Y. Zweiri, M. A. K. Jaradat, R. Al-Husari, D. Gan, and L. D. Seneviratne, "Deep-learning-based neural network training for state estimation enhancement: Application to attitude estimation," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 1, pp. 24–34, 2019.

[109] M. Mérida-Floriano, F. Caballero, D. García-Morales, F. Casares, and L. Merino, "Bioinspired vision-only uav attitude rate estimation using machine learning," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2017, pp. 1476–1482.

[110] N. H. Singh and K. Thongam, "Neural network-based approaches for mobile robot navigation in static and moving obstacles environments," *Intelligent Service Robotics*, vol. 12, no. 1, pp. 55–67, 2019.

[111] V. J. Hodge, R. Hawkins, and R. Alexander, "Deep reinforcement learning for drone navigation using sensor data," *Neural Computing and Applications*, vol. 33, no. 6, pp. 2015–2033, 2021.

[112] H. Huang, Y. Yang, H. Wang, Z. Ding, H. Sari, and F. Adachi, "Deep reinforcement learning for uav navigation through massive mimo technique," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 1117–1121, 2019.

[113] S. F. Abedin, M. S. Munir, N. H. Tran, Z. Han, and C. S. Hong, "Data freshness and energy-efficient uav navigation optimization: A deep reinforcement learning approach," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[114] J.-P. Tardif, M. George, M. Laverne, A. Kelly, and A. Stentz, "A new approach to vision-aided inertial navigation," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 4161–4168.

[115] C. Hide, T. Moore, and M. Smith, "Adaptive kalman filtering algorithms for integrating gps and low cost ins," in *PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No. 04CH37556)*. IEEE, 2004, pp. 227–233.

[116] Y. Li, J. Wang, C. Rizos, P. Mumford, and W. Ding, "Low-cost tightly coupled gps/ins integration based on a nonlinear kalman filtering design," in *Proceedings of the 2006 National Technical Meeting of The Institute of Navigation*, 2006, pp. 958–966.

[117] M. G. Petovello, *Real-time integration of a tactical-grade IMU and GPS for high-accuracy positioning and navigation*. Citeseer, 2003.

[118] G. Huang, M. Kaess, and J. J. Leonard, "Towards consistent visual-inertial navigation," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 4926–4933.

[119] S. Zhao, Y. Chen, and J. A. Farrell, "High-precision vehicle navigation in urban environments using an mem's imu and single-frequency gps receiver," *IEEE transactions on intelligent transportation systems*, vol. 17, no. 10, pp. 2854–2867, 2016.

[120] J. Farrell, *Aided navigation: GPS with high rate sensors.* McGraw-Hill, Inc., 2008.

[121] A. Barrau and S. Bonnabel, "A mathematical framework for imu error propagation with applications to preintegration," in *2020 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 2020, pp. 5732–5738.

[122] A. Nemra and N. Aouf, "Robust ins/gps sensor fusion for uav localization using sdre nonlinear filtering," *IEEE Sensors Journal*, vol. 10, no. 4, pp. 789–798, 2010.

[123] M. S. Grewal, L. R. Weill, and A. P. Andrews, *Global positioning systems, inertial navigation, and integration.* John Wiley & Sons, 2007.

[124] M. George and S. Sukkarieh, "Tightly coupled ins/gps with bias estimation for uav applications," in *Proceedings of Australiasian Conference on Robotics and Automation (ACRA)*, 2005.

[125] R. Babu and J. Wang, "Improving the quality of imu-derived doppler estimates for ultra-tight gps/ins integration," *GNSS2004*, pp. 16–19, 2004.

[126] ——, "Ultra-tight gps/ins/pl integration: a system concept and performance analysis," *GPS solutions*, vol. 13, no. 1, pp. 75–82, 2009.

[127] G. Yuan and T. Zhang, "Unscented kalman filtering for ultra-tightly coupled gps/ins integration," in *2009 International Conference on Mechatronics and Automation.* IEEE, 2009, pp. 4556–4560.

[128] D.-J. Jwo, C.-F. Yang, C.-H. Chuang, and K.-C. Lin, "A novel design for the ultra-tightly coupled gps/ins navigation system," *The Journal of Navigation*, vol. 65, no. 4, pp. 717–747, 2012.

[129] L. S. Monteiro, T. Moore, and C. Hill, "What is the accuracy of dgps?" in *Proceedings of the 61st Annual Meeting of The Institute of Navigation (2005)*, 2005, pp. 805–817.

[130] C. Chen, H. Zhu, M. Li, and S. You, "A review of visual-inertial simultaneous localization and mapping from filtering-based and optimization-based perspectives," *Robotics*, vol. 7, no. 3, p. 45, 2018.

[131] C.-C. Chou and C.-F. Chou, "Efficient and accurate tightly-coupled visual-lidar slam," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[132] T. Qin, J. Pan, S. Cao, and S. Shen, "A general optimization-based framework for local odometry estimation with multiple sensors," *arXiv preprint arXiv:1901.03638*, 2019.

[133] H. Fischer, *A history of the central limit theorem: from classical to modern probability theory.* Springer Science & Business Media, 2010.

[134] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The euroc micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

[135] C. Specht, J. Pawelski, L. Smolarek, M. Specht, and P. Dabrowski, "Assessment of the positioning accuracy of dgps and egnos systems in the bay of gdansk using maritime dynamic measurements," *The Journal of Navigation*, vol. 72, no. 3, pp. 575–587, 2019.

[136] H. Chengyan, G. Ji, L. Xiaochun, and L. Jun, "Multipath performance analysis of gnss navigation signals," in *2014 IEEE Workshop on Electronics, Computer and Applications.* IEEE, 2014, pp. 379–382.

[137] K. Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quarterly of applied mathematics*, vol. 2, no. 2, pp. 164–168, 1944.

[138] R. De Maesschalck, D. Jouan-Rimbaud, and D. L. Massart, "The mahalanobis distance," *Chemometrics and intelligent laboratory systems*, vol. 50, no. 1, pp. 1–18, 2000.

[139] Velodyne, "Surround lidar," Online, accessed: 10-March-2022.

[140] L. Oth, P. Furgale, L. Kneip, and R. Siegwart, "Rolling shutter camera calibration," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1360–1367.

[141] M. Nieuwenhuisen, D. Droeschel, M. Beul, and S. Behnke, "Obstacle detection and navigation planning for autonomous micro aerial vehicles," in *2014 international conference on unmanned aircraft systems (ICUAS).* IEEE, 2014, pp. 1040–1047.

[142] F. Ferri, M. Gianni, M. Menna, and F. Pirri, "Dynamic obstacles detection and 3d map updating," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* IEEE, 2015, pp. 5694–5699.