GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

RESEARCH ON THE AVAILABILITY OF VINS-MONO AND ORB-SLAM3 ALGORITHMS FOR AVIATION

Burak Kaan ÖZBEK

Supervisor
Assist. Prof. Dr. Metin TURAN

MASTER'S THESIS
DEPARTMENT OF COMPUTER ENGINEERING
ISTANBUL - 2021

ACCEPTANCE AND APPROVAL PAGE

On 25/02/2021 Burak Kaan Özbek successfully defended the thesis, entitled "Research on the Availability of VINS-Mono and ORB-SLAM3 Algorithms for Aviation", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury members whose signatures are listed below. This thesis accepted as a Master's Thesis by Istanbul Commerce University, Graduate School of Natural and Applied Sciences.

Approved By:

Supervisor	**Assist. Prof. Dr. Metin TURAN** Istanbul Commerce University
Jury Member	**Assist. Prof. Dr. Arzu KAKIŞIM** Istanbul Commerce University
Jury Member	**Assist. Prof. Dr. Ertuğrul ÇETİNSOY** Marmara University

Approval Date: 15.03.2021

Istanbul Commerce University, Graduate School of Natural and Applied Sciences, accordance with the 1st article of the Board of Directors Decision dated 15.03.2021 and numbered 2021/308, "Burak Kaan Özbek" (TC: 26503474360) who has determined to fulfill the course load and thesis obligation was unanimously decided to graduated.

Prof. Dr. Necip ŞİMŞEK
Head of Graduate School of Natural and Applied Sciences

DECLARATION OF ACADEMIC AND ETHIC INTEGRITY

I hereby declare that,

- I have obtained the all information and documents within the academic and ethical rules,
- I have presented all visual and written information and results in accordance with academic ethics,
- I refer to the relevant studies in case the studies of others are used,
- neither whole nor any part of this thesis is not presented in this university or any other university, previously.

15/03/2021

Burak Kaan ÖZBEK

TABLE OF CONTENTS

ABSTRACT

M.Sc. Thesis

RESEARCH ON THE AVAILABILITY OF VINS-MONO AND ORB-SLAM3 ALGORITHMS FOR AVIATION

Burak Kaan ÖZBEK

Istanbul Commerce University
Graduate School of Applied and Natural Sciences
Department of Computer Engineering

Supervisor: Assist. Prof. Dr. Metin TURAN
2021, 57 pages

Navigation is also referred to as knowledge about how to get from one point to another on the ground, which most people have an idea about. For both civil and military aviation, navigation is a field of concern. Therefore, in order to track the aircraft path, we get support from different sensors. GPS is the most commonly used sensor among them. It is a sensor that, although it has high accuracy rates, may be out of service. This study focused on the avaliability of aircraft's navigation in conditions where GPS is not in operation. In terms of efficiency, two visual-inertial navigation systems, VINS-Mono and ORB-SLAM3, which are the most well-known algorithms in the literature, have been examined and compared. In various conditions, it was found that ORB-SLAM3 outperformed the VINS-Mono system almost twice.

Keywords: Navigation, Simultaneous Localization and Mapping, Visual-Inertial Navigation, Visual Odometry.

ÖZET

Yüksek Lisans Tezi

HAVACILIK İÇİN VINS-MONO VE ORB-SLAM3 ALGORİTMALARININ KULLANILABİLİRLİĞİ ÜZERİNE ARAŞTIRMA

Burak Kaan ÖZBEK

İstanbul Ticaret Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Metin TURAN
2021, 57 sayfa

Çoğu insanın hakkında fikir sahibi olduğu seyrüsefer, genellikle karada bir noktadan diğerine nasıl gideleceği hakkında bilgi olarak sahibi olmak olarak bilinir. Seyrüsefer hem sivil hem de askeri havacılığın ilgi alanıdır. GPS aralarında en yaygın olarak kullanılan sensördür. Yüksek doğruluk oranlarına sahip olmasına rağmen kullanım dışı kalabilen bir sensördür. Bu araştırma, GPS'in kullanım dışı kaldığı ortamlarda bir uçağın seyrüseferini sürdürebilmeye odaklanmıştır. Verimlilik açısından literatürde en çok bilinen algoritmalar olan iki görsel-ataletsel navigasyon sistemi, VINS-Mono ve ORB-SLAM3 sistemleri incelenmiş ve performans açısından karşılaştırılmıştır. ORB-SLAM3'ün çeşitli durumlarda VINS-Mono'ya kıyasla iki kat daha iyi performans gösterdiği görülmüştür.

Anahtar Kelimeler: Eşzamanlı Konumlandırma ve Haritalama, Görsel-Ataletsel Seyrüsefer, Görsel Odometri, Seyrüsefer.

ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisor, Assist. Prof. Dr. Metin TURAN, who guided me tirelessly in the face of any technical and social problems I encountered during my thesis.

I would like to thank my colleagues for understanding and giving me endless support during this stressful time.

Finally, I thank my family for their love, support, and belief in me.

Burak Kaan ÖZBEK
İSTANBUL, 2021

LIST OF FIGURES

LIST OF TABLES

SYMBOLS AND ABBREVIATIONS LIST

APE	Absolute Pose Error
AR	Augmented Reality
BA	Bundle Adjustment
BRIEF	Binary Robust Independent Elementary Features
DoF	Degrees of Freedom
DoG	Differnece of Gaussian
EKF	Extended Kalman Filter
FAST	Features from Accelerated Segment Test
GPS	Global Positioning System
IMU	Inertial Measurement Unit
KLT	Kanade Lucas Tomasi
LCD	Loop Closure Detection
MAV	Micro Aerial Vehicles
NAVAIDS	Navigation Aids
ORB	Oriented FAST and Rotated BRIEF
RANSAC	Random Sample Consensus
RMSE	Root Mean Square Error
RPE	Relative Pose Error
SFM	Structure from Motion
SIFT	Scale Invariant Feature Transform
SURF	Speeded-Up Robust Features
SLAM	Simultaneous Localization and Mapping
UAV	Unmanned Aerial Vehicle
VIO	Visual-Inertial Odometry
VI-SLAM	Visual-Inertial Simultaneous Localization and Mapping
VO	Visual Odometry
V-SLAM	Visual SLAM
VR	Virtual Reality

1. INTRODUCTION

The word aviation is most widely used to refer to mechanical air transportation performed by aircraft. Aeroplanes and helicopters are the two most common types of aircraft, but most current research meanings of the term aviation includes the use of unmanned aircraft, such as drones.

1.1. Aviation Overview

There are two forms of flight in the aviation industry: civil and military. Civil aviation, to put it plainly, is all aviation that is not related to the military. This extends to all private and commercial aircraft, regardless of whether they transport passengers, cargo, or a mixture of the two.

Military aviation, on the other hand, refers to the use of aircraft in military environments. This form of air transportation is usually used to support aerial combat or surveillance missions. The majority of military aviation is affiliated with air forces, but there are also terms such as army aviation, navy aviation, and coast guard aviation (Revfine, 2021).

In recent years, we can say that unmanned aerial vehicles (UAV) have started to take an important place in military, civil aviation applications as well as aircrafts. Although we cannot reach the numerical data of recent years, we can clearly see the increase in many areas of unmanned aerial vehicle systems used in the 4-year period from 2004-2007 as shown in Table 1.1.

Table 1.1. Numbers of UAV systems in years (Everaerts, 2008)

	2004	2005	2006	2007
Civil/Commercial	33	55	47	61
Military	362	397	413	491
Dual purpose	39	44	77	117

Research	43	35	31	46
Developmental			217	269

The other reason for so demand is that, delivery companies such as Amazon, Uber, Google realized that they could use UAVs as a delivery platform. Thereupon, research and development activities on unmanned aerial vehicles were increased. Also, demand is expected to accelerate in the coming years as more companies study how UAVs can make their job safer and more cost-effective. The entire UAV market is expected to be worth $92 billion by 2030 (Daly, 2021).

In addition, there is an autopilot system that does not receive direct assistance from the pilot, usually in civil aircraft flying on a certain route. Autopilots used to be limited to maintaining a steady heading and speed, but today's autopilots can monitor any aspect of flight envelope from takeoff to landing. Working with the autopilot software, the autopilot owes his ability to its integration with the navigation system (Skybraryaero, 2021).

Considering all this, safety is a big concern in the aviation industry, and although the rate of incidents is lower than in other industries, the survival rate is also lower. Many aviation accidents are caused by human or pilot error, despite the advancement of autopilot and other technical systems and their improved reliability. Li, Baker, Grabowski, and Rebok (2001) also state that, 80% of aviation accidents are caused by pilot error (Akca, 2020). In order to cope with problems, the aviation industry must become more competitive.

In aviation, a range of technologies are used to navigate an aircraft. Each of these systems serves a distinct function and has a distinct mode of operation. Navigation is an important feature of aviation that is affected by a number of factors. As a consequence, in an environment where UAVs begin to manifest themselves in every field, where human error is tried to be minimized, countries in military aviation are trying to neutralize enemy with electromagnetic waves, undoubtedly, unmanned or manned aircraft are open to attack and error, such

as GPS, in order to fullfill their duties completely in all conditions. It is of great importance having an advanced navigation system is crucial in terms of both safety and cost (Mishra, 2019) and develop a positioning and navigation system which is more robust in situations that mentioned above.

1.2. Problem Statement

The method of determining where a mobile robot is situated on its environment is robot localization. Localization is one of the most fundamental competencies that an autonomous robot requires because of understanding the robot's own location is a vital precursor to future action decision-making. An environment map is available in a typical robot location scenario and the robot is fitted with sensors that track the environment and control its own motion.

Navigation is a method that finds a way from one position to another. In order to complete the mission, aircraft navigation is one of the most significant fields of application, whether military or civil.

In aviation, there are different methods for navigation. Although the details are not the topic of this study, it would be helpful to clarify briefly why we want to do this research.

Aviation navigation is primarily carried out using two techniques known as dead reckoning and pilotage (Figure 1.1). By reference to various visual landmarks such as rivers, towns, airports and buildings, we can describe pilotage as the process by which the pilot navigates. However, in conditions of low visibility or where the pilot is slightly off track, the reference points are often not easily identified (Houston, 2019).

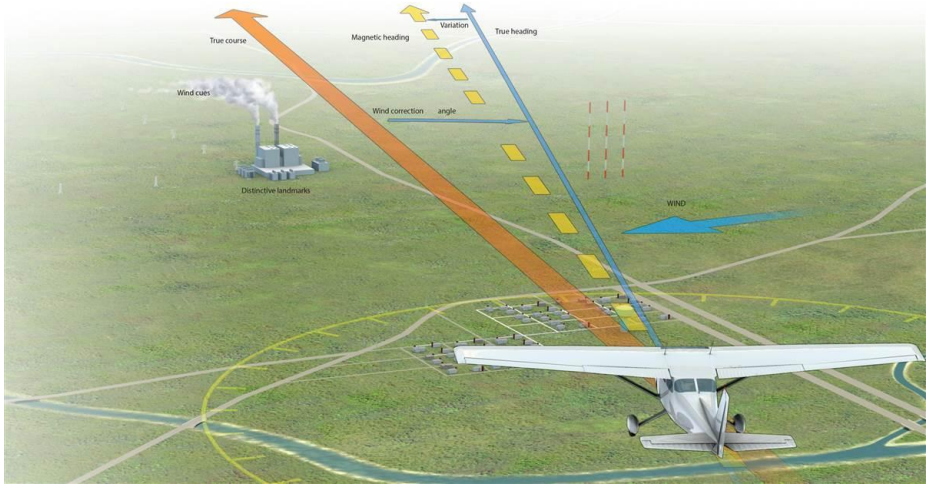


Figure 1.1. Pilotage and dead reckoning (Marsh, 2016)

The system used by the pilot while traveling overseas, woods or deserts requires more skill and experience than pilotage is dead reckoning. It is a method of navigation that relies on parameters such as time, airspeed, distance, and direction only. From one point to another, the pilot must know the distance. On the pre-flight plan diagram, the pilot will schedule his route in advance. While the pilot is flying at a constant speed, with the aid of the compass, pilot will measure how long to achieve pilot objective and will keep the plane in the right direction. However, dead reckoning is not always a reliable technique because of the shifting wind direction (Flight Literacy, 2020).

The sensors that assist in the methods we mentioned above called navigation aids (NAVAIDS). The Global Positioning System (GPS) is the most relevant of these sensors (Figure 1.2). As the most commonly used navigation aid today, GPS has proved how effective and powerful it is. GPS can provide navigation services at any time in the world under any weather conditions, while enhancing flight stages from departure, route progress and navigation on the surface of the airport (GPS, 2006).

Figure 1.2. GPS unit (Aeronautics Guide, 2017)

GPS is also prone to faulty performance, considering its accuracy and ease of use. It is unavoidable that businesses or governments would want to be more cautious against errors and assaults, considering the development of technology and significant investments in aviation. In cases where GPS is not working or producing incorrect results, there are many other navigation methods available as mentioned above. In this study, we asked the question of how to use passive sensors (cameras) to ensure navigation with high accuracy and protection from attacks. In response, we saw that we could benefit from the Simultaneous Localization and Mapping (SLAM) systems that we encounter in the field of robotics, and we tended research on this way.

SLAM methods in the literature were studied in the first section of this thesis. In the second section, we will give an overview of aviation. The fusion of the camera and the Inertial Measurement Unit (IMU) was discussed in the third section. Finally, comparative experiments were carried out on the public data related to the two most effective SLAM frameworks in the literature, VINS-Mono and ORB-SLAM3, and the problems that may be encountered in aviation mission on aircrafts scenarios were discussed.

2. LITERATURE REVIEW

In both military aviation and civil aviation, aircraft navigation knowledge has an important role. It is likely that aircraft that receive assistance from different sensors, such as GPS, Radar and Lidar, will be open to errors and attacks when delivering this information to us. The purpose of this study is to take a closer look at the SLAM approach to the navigation problem in aircraft and to present a solution plan by comparing the most robust algorithms in the literature.

While localization and mapping were first viewed as separate issues, it was agreed that they should later be concerned with together. Previous studies have been reviewed in-depth in the literature in order to support research on this topic.

Probabilistic approaches had just started to join the area of robotics and artificial intelligence when IEEE's Robotics and Automation conference took place in 1986. Peter Cheeseman, Jim Crowley, and Hugh Durrant-Whyte were among the researchers looking at adapting estimation-theoretical techniques to mapping and localization problems.

Experiments by Smith and Cheeseman (1986) have provided a basis for manipulating the associations between signs and geometrical ambiguity. An essential component of this framework was to show that there is a high degree of correlation between estimates of various landmark locations on a map and that these correlations can also increase with successive observations.

SLAM is a method according to Bailey and Durrant-Whyte (2006) by which a mobile robot can use the map to create an environment map and determine its location as well. In SLAM (Figure 2.1), both the position of the vehicle and the location of all key points are predicted on-line without the need to know the location in advance.

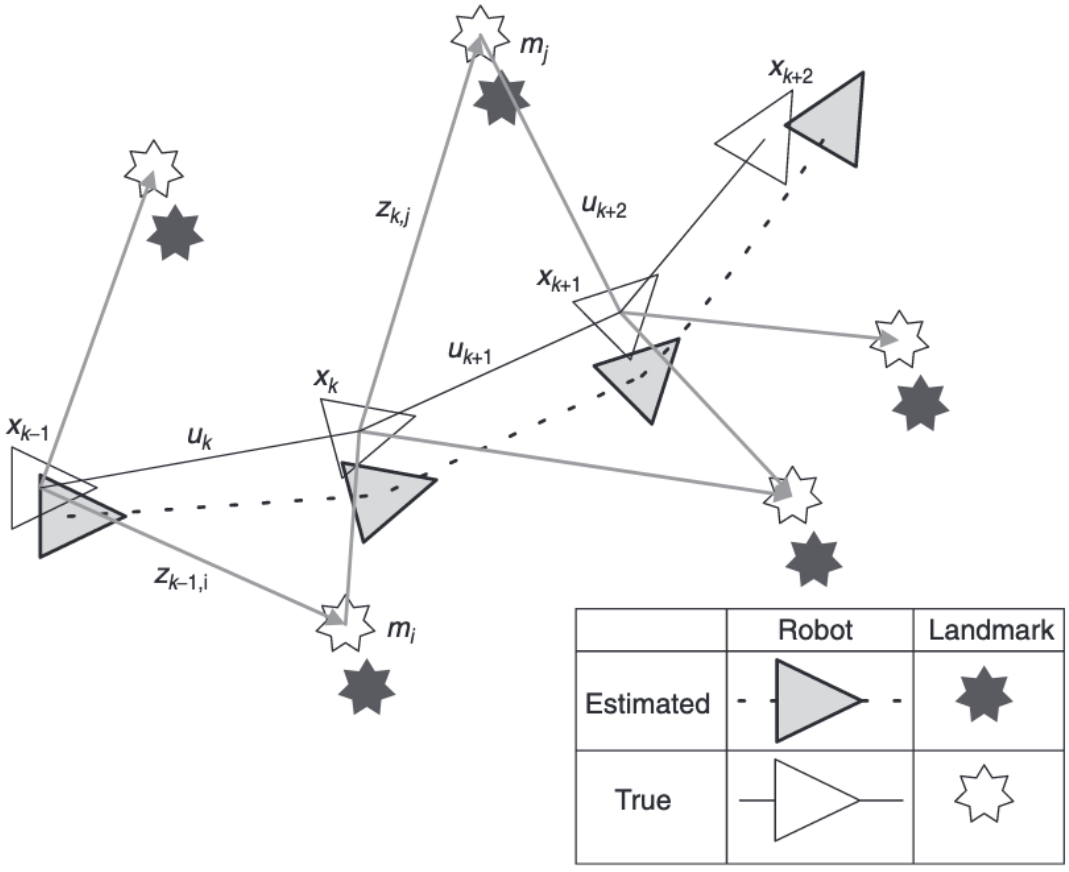


Figure 2.1. SLAM architecture (Bailey and Durrant-Whyte, 2006)

Mapping and localization were initially separately studied, and it was later realized that they relied on each other. This implies that a correct map is needed in order to be precisely located in the environment, but it is necessary to produce a good map. To be correctly positioned when the map is applied to the elements. It can be called vision-only SLAM or visual SLAM when vision is used as the only perception system (without the use of data obtained from robot odometry or inertial sensors) (Paz et al., 2016).

Davison et al. (2003) proposed one of the innovative Visual SLAM solutions (Figure 2.2). They used a single monocular camera and created a map by extracting uncommon features of the region using Shi and Tomasi (Shi and Tomasi, 1994) and comparing new features to those already found using a standardized correlation of the sum-of-squared difference.

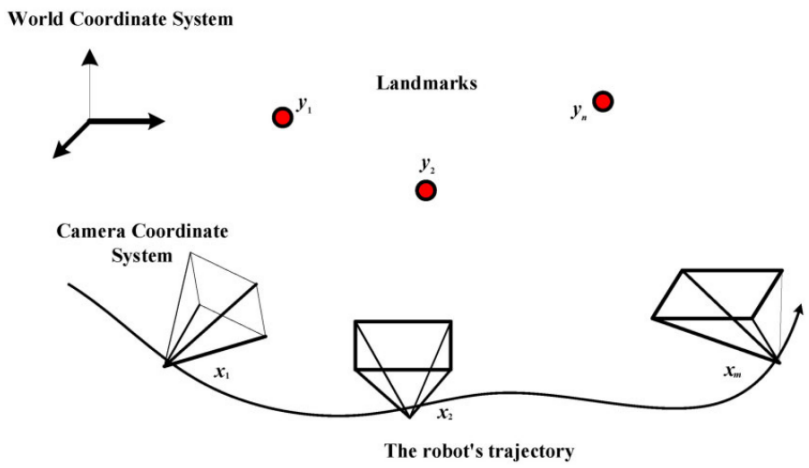


Figure 2.2. Visual SLAM representation (Yu and Shengyong, 2018)

In addition, only a small number of features were extracted and monitored to manage the computational expense of the Extended Kalman Filter (EKF) because EKF was used for state estimation. A vision-based approach to the localization and mapping of mobile robots using Scale-Invariant Feature Transform (SIFT) was proposed by Lowe (2004) for the extraction of features.

When operating under the following conditions, many Visual SLAM systems fail: in external environments, in dynamic environments, in environments with too many or too few salient features, in large-scale environments, during irregular movements of the camera, etc. The key to a good visual SLAM system is the ability to function correctly in spite of these difficulties.

With the objective of increasing accuracy and robustness, visual SLAM systems can be complemented by proprioceptive sensor information. This is referred to as Visual-Inertial SLAM by Jones and Soatto (2011).

For a variety of reasons, Visual-Inertial Simultaneous Localization and Mapping (VI-SLAM), which combines camera and IMU data for localization and environmental perception, has become increasingly popular. The technology is used in robotics, in particular in extensive studies and applications involving

autonomous micro-aerial vehicle (MAV), augmented reality (AR) and virtual reality (VR) navigation (VR).

One of the two concerns that SLAM is seeking to address is localization. One of the SLAM subtitles based on solving this issue is Visual Odometry (VO) (Figure 2.3).

The term VO was introduced in his landmark paper by (Nister, 2004). This term was used because of its similarity to the odometry of the wheel, which increasingly predicts the component's motion.

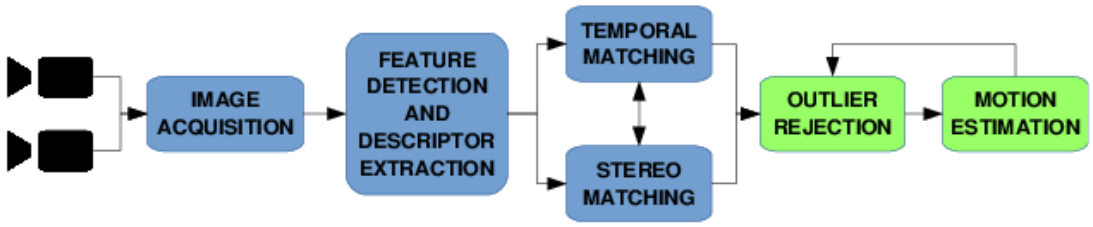


Figure 2.3. General visual odometry pipeline (Ivan and Sinisa, 2015)

According to Webster et al. (2016), visual odometry, by analyzing sequential camera images, measures the relative motion of the camera. The errors associated with the estimations obtained in visual odometry accumulate over time, similar to wheel odometry.

Maps of the environment in which a mobile robot is required to locate and navigate are not accessible in most real-world robotics applications. Therefore, in order to achieve true autonomy, one of the key competencies of autonomous vehicles is the creation of a world map.

SLAM, on the other hand, by using the following filtering approaches such as EKF-SLAM (Thrun, 2002) and particle filter-based SLAM (Lu and Milios, 1997), and smoothing approaches such as Graph-SLAM (Thrun and Montemerlo, 2006) and RGB-D SLAM (Henry, 2012), goals to solve these two problems (localization and mapping) simultaneously.

The map, modeled using Gaussian variables, is a large stacking sensor vector and a landmark state in EKF-SLAM. The EKF preserves this map, commonly referred to as the stochastic map, via the process of prediction (sensor movement) and correction (sensors observe previously mapped landmarks in the environment). Through nonlinear sparse optimization, Graph-SLAM solves the SLAM problem. They transform their intuition into a graphical representation of the problem of SLAM.

The strength of the techniques of the graphical SLAM is that they scale to many higher-dimensional maps than the EKF-SLAM. The covariance matrix, which requires quadratic space in the map size, is the primary limiting factor in the EKF-SLAM. Graphical approaches do not have those disadvantages.

Particle filters are the other main paradigm for SLAM. As a concrete guess as to what the true value of the state might be, each particle is best thought of. By collecting a number of such conjectures, particle filters catch a representative sample from the posterior distribution to obtain a collection of conjectures or a set of particles (Montemerlo et al., 2002).

Using a filter-based or optimization-based approach to fuse visual and IMU measurements is a general and effective solution to navigation errors caused by the IMU's low-frequency noise. During the fusion process, IMU and Camera are combined to create a Visual-Inertial Odometry that not only takes advantage of the versatility of the visual system and is adaptable to a wide range of scenes but also utilizes the high-precision features of the IMU in the short term. Visual and inertial sensor-based research on SLAM algorithms is therefore of great

importance and application importance, allowing vehicles to view the ambient environment in order to gain localization knowledge (Sun et al., 2018).

3. SIMULTANEOUS LOCALIZATION AND MAPPING

The problem of simultaneous localization and map building asks whether it is possible for an autonomous vehicle to start in an unknown location in an unknown environment and then gradually create a map of this environment while simultaneously using this map to measure the absolute location of the vehicle (Dissanayake et al., 2001).

A very active research topic is the use of the camera as the key source of information for environmental sensing, as the camera is lightweight, consumes less power and offers a wealth of information as well. Nowadays, digital cameras with low power consumption are inexpensive and have a lightweight form factor. Under severe conditions, they can also operate safely, since there are no moving mechanical parts. However, the job races by time in order to process a large amount of real-time information in order to create both the environmental model and the camera location in each case.

3.1. Localization

The method of determining where a mobile robot is situated on its environment is robot localization. Localization is one of the most fundamental competencies that an autonomous robot requires because of understanding the robot's own location is a vital precursor to future action decision-making. An environment map is available in a typical robot location scenario and the robot is fitted with sensors that track the environment and control its own motion.

The problem of localization then becomes estimation of the robot's position and orientation within the map using the data obtained from these sensors. Robot localization techniques need to be able to manage noisy observations and not only estimate the robot's position, but also measure the uncertainty of the estimate for the location (Webster et al., 2016).

Simple trigonometric computation is required on the robot's absolute location until a correspondence is established between the perceived features in their local frame or reference and the mapping features in the absolute frame or reference. The perceptual characteristics of the robot are the basic criterion for the applicability of specific methods. The capacity to uniquely identify landmarks, for instance, would render the issue of correspondence tirivial. In order to facilitate the recognition of a robot in the working world, industrial applications also use special artificial tags (Hahnel et al., 2004).

3.2. Mapping

There are two main types of maps used for visual localization; metric maps and topological maps.

By gathering sequences of locations, topological maps represent the world in a relative way. The locations themselves are only represented in a very rough way. Typically, topological maps are represented using graph-based structures where nodes correspond to locations and edges reflect approximate transformations between places.

A topological map consisting of visual pathways (Figure 3.1) to connect locations captured for indoor navigation purposes by wearable cameras is defined by Rivera-Rubio et al. (2014). They also illustrate that considering multiple frames (as opposed to single frames in isolation) increases the efficiency of localization across visual paths.

In guiding robots where it is difficult to obtain globally accurate metric maps, such as in mines and indoors, topological maps, often used in teaching-and-repeating navigation methods, were helpful. These methods of visual navigation have the benefit of a minimum setup time. The mapper records the visual information it observes along the road in the first traversal of the environment. The follower will be able to locate itself along the mapper's trajectory in the navigation phase and follow the path to the desired location. The system

produces a relative map of stereo images (Furgale and Barfoot, 2010) that allows the follower to locate any location along the map and navigate to any destination.

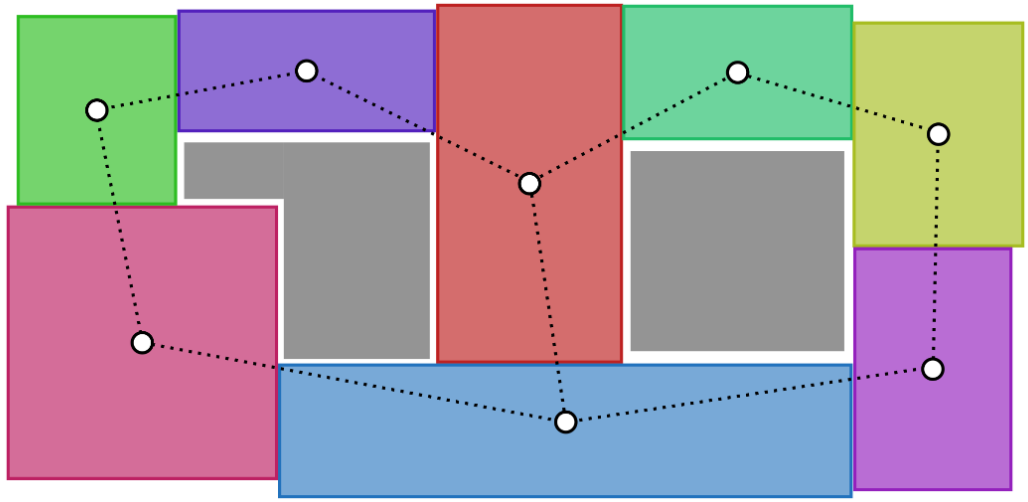


Figure 3.1. Topological graph (Blöchliger et al., 2018)

In order to enable precise location and route planning, metric maps are designed to provide precise communication between the environment and its representation. Examples of metric maps are maps in the form of 3D models or 3D point clouds. It is possible to obtain such point-cloud maps from an unordered image set using a structure from motion algorithm. The structure from motion is defined by the offline reconstruction of 3D maps from an unordered image set. Structure From Motions (SFM) algorithm operates in a greedy way on image pairs, building up a consistent 3D scene model gradually. The SFM algorithm achieves the highest possible accuracy of the model output thanks to bundle modification (i.e. non-linear optimization of the re-projection errors of the recovered dots). The intrinsic parameters of the camera do not need to be specified from the user's point of view before the reconstruction is carried out, as these parameters can be used in the optimization process. SFM algorithm has been popular indoor localization literature for this purpose as technique for building 3D models that can be used to locate mobile devices (Clark, 2017).

3.3. Visual SLAM and Visual Odometry

Visual SLAM (V-SLAM) refers to the method of determining the sensor's position and orientation in relation to its environment, while mapping the sensor's surroundings at the same time.

V-SLAM is a particular type of SLAM system that leverages 3D vision when neither the environment nor the location of the sensor is known to perform position and mapping functions. Visual SLAM technology is available in different ways, but in all visual SLAM systems, the overall architecture functions in the same way (Vision Online, 2018).

In the computer vision community, the issue of obtaining relative camera poses and three-dimensional (3D) structure from a set of camera images (calibrated or non-calibrated) is known as SFM as defined above (Figure 3.2).

A particular case of SFM is Visual Odometry (VO). SFM is more general technique and tackles with the problem of reconstruction of 3D from sequentially ordered or unordered image sets using both the structure and the camera. With offline optimization (i.e. package adjustment), the final structure and camera poses are typically refined, the measurement time of which increases with the number of images (Bailey and Durrant-Whyte, 2006). VO, on the other hand, focuses on estimating the camera's 3D motion sequentially when a new frame arrives and process it in real time. To refine the local trajectory estimation, package adjustment can be used.

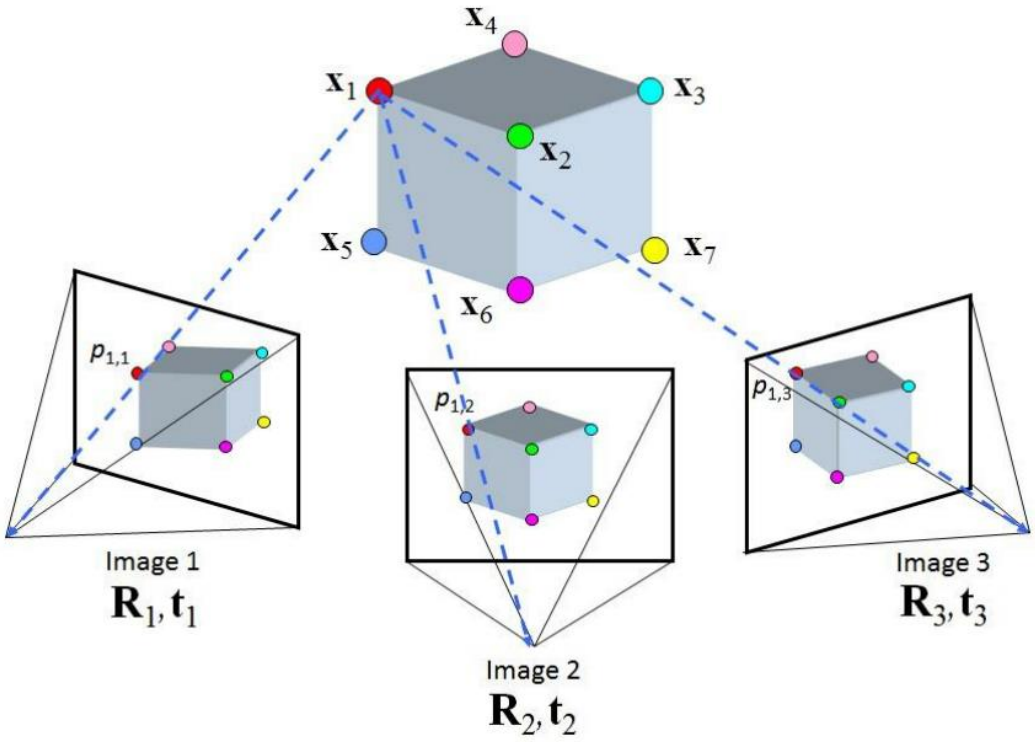


Figure 3.2. Structure from motion process (Yilmaz and Karakus, 2013)

VO aims to gradually recover the route, pose after posing, and theoretically optimize just above the last n path poses (this is also called windowed bundle adjustment). It is possible to consider this sliding window optimization to be analogous to constructing a local SLAM map. In VO, however, we are only concerned with the trajectory's local consistency, and the local map is used to obtain a more reliable local trajectory approximation (for example, in the bundle adjustment), while SLAM is concerned with the consistency of the global map (Scaramuzza and Fraundorfer, 2011).

3.3.1. Camera modelling and calibration

The mathematical model of the camera consists of conversion algorithms between the position of the points in the world of the 3D object and their existence as points on the plane of the 2D image. If the camera's intrinsic and extrinsic parameters and the observed position of the 3D object points are known, the camera model can be used to decide at which point the object ends

on the image. It can also be used in the other way around; if an image point and camera parameters are known, all possible object points from which the cloud image point originates can be measured by the camera model.

The aim of the calibration is to measure the camera device's internal and external parameters precisely. Planar checkerboard-like (Figure 3.3) patterns are the most common type. It describes how the squares on the board are placed. The consumer must take a number of pictures of the board shown in different locations and directions in order to accurately calculate the calibration parameters by making sure that the camera's field of view is as wide as possible. Using the least-square minimization procedure, intrinsic and extrinsic parameters can then be measured. The input data is 2D position of each image on the board squares' corners and their corresponding pixel coordinates.

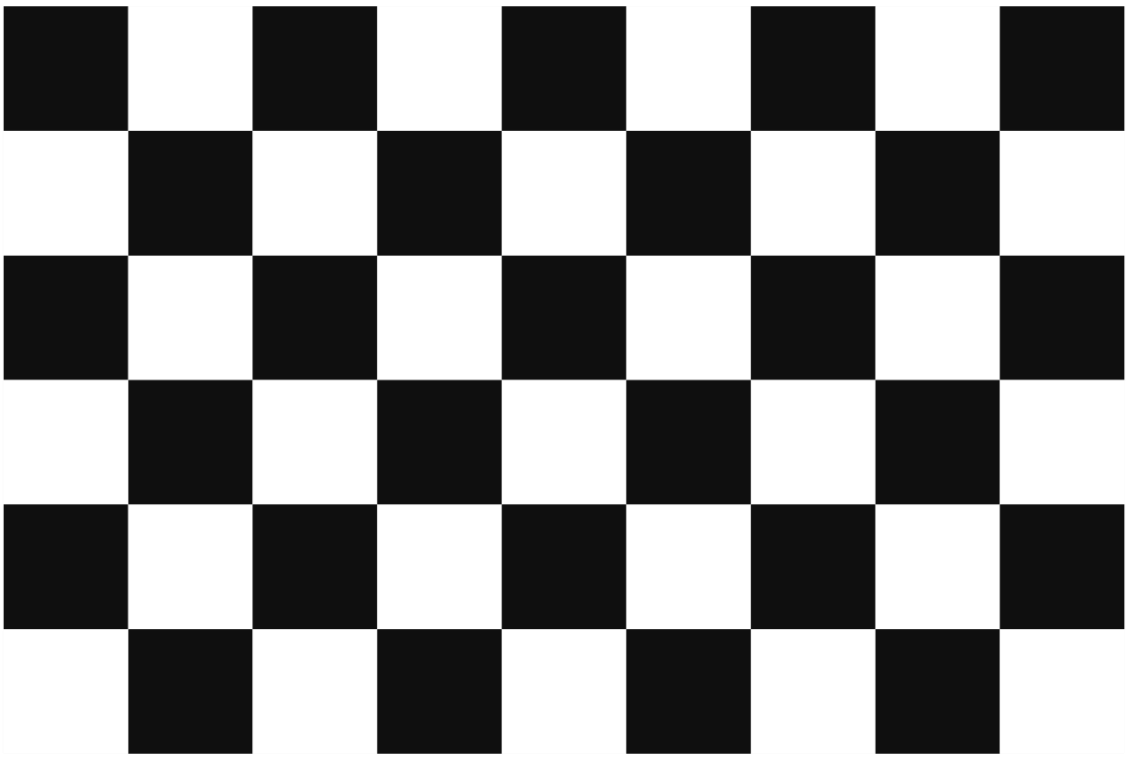


Figure 3.3. Checkerboard (8x5) (Jones, 2019)

3.3.2. Feature detection

A local feature is an image pattern that varies in terms of intensity, color and texture from its immediate surroundings. Point detectors, such as corners, are important for VO, their position in the picture must be precisely determined. Some of the state-of-the-art algorithms for the detection of features are briefly explained below.

3.3.2.1. Harris corner detector

Instead of using moving patches for every 45-degree angle, Harris corner detector (Harris and Stephens, 1988) takes the difference in angle score directly into account in relation to position and has been shown to be more effective in the distinction between edges and corners. Since then, in many algorithms, the pre-processing of images for subsequent applications has been improved and introduced.

3.3.2.2. Scale invariant feature transform

Scale Invariant Feature Transform (SIFT) has the scale invariance property, which makes it better than Harris. Harris is not scale-invariant; if the scale changes, a corner can become an edge.

The SIFT algorithm consists mainly of 4 steps. The first is the calculation of extreme scale space using the Difference of Gaussian (DoG). Second, a key point location in which the key point candidates are localized and optimized by removing the low contrast points. Third, a key point orientation assignment based on the local image gradient and, lastly, a descriptor generator to calculate the local image descriptor based on the image gradient magnitude and orientation for each key point gradient (Lowe, 2004).

3.3.2.3. Speeded-up robust features

A fast and robust algorithm for local, connected invariant representation and image comparison is the Speeded-Up Robust Feature (SURF) method (Bay et al., 2006). The main aim of the SURF method is to quickly compute operators using box filters, allowing real-time applications such as tracking and object recognition to be used.

3.3.2.4. Features from accelerated segment test

Features from Accelerated Segment Test (FAST) is a method of corner detection that could be used in a number of computer vision tasks to extract feature points and later to track and map objects (Viswanathan, 2011). FAST compare pixels only on a fixed radius circle around the point.

3.3.2.5. Shi-tomasi

The basic understanding here is that corners can be identified in all directions by looking for visible changes. We consider a small image window in this process that scans the whole image, looking for corners. If that specific window happens to be situated in the corner, moving this small window in any direction can result in a noticeable change in appearance. In either direction, there will be no change in the flat area.

This algorithm is based on a model of refined image changes and a technique for monitoring features during tracking (Shi and Tomasi, 1994). In particular, the selection maximizes the tracking quality and is therefore suitable for building, as opposed to more ad-hoc texture redness steps. Based on a measure of dissimilarity that uses affine motion as the underlying model of image change, monitoring is computationally inexpensive and helps to differentiate between good and bad characteristics.

3.3.2.6. Oriented FAST and rotated BRIEF

A fast, robust local feature detector, first proposed by Ethan Rublee et al. (2011), is an Oriented FAST and Rotated BRIEF (ORB) that can be used for computer vision tasks such as object recognition or 3D reconstruction. It is based on the FAST keypoint detector and the visual descriptor Binary Robust Independent Elementary Features (BRIEF) edited edition. Itis an alternative to the SIFT with a fast and effective way.

3.3.3. Feature tracking

To discover the features and their correspondence, there are two main approaches. The first is to recognize features in one image and, using local search techniques, such as correlation, track them in the next images. In all images, the second is to independently detect features and align them based on some metric similarity between their descriptors. When the pictures are taken from nearby points of view, the former approach is more suitable, while the latter is more suitable when a large motion or point of view is needed to move.

Their appearance may undergo major changes in the case of features that are tracked over long sequences of images, in which case the solution is to apply an affine-distortion model to each feature. The resulting tracker is also referred to as the Kanade Lucas Tomasi (KLT) tracker (Bruce and Kanade, 1981).

3.3.3.1. Kanade lucas tomasi tracker

Kanade Lucas Tomasi Tracker (KLT) is an algorithm for tracking changes. It is seeking, in its basic form, to find a change that might have taken a point of interest. The method is based on local optimization: a criterion of squared distance over a local area generally. You approximate a linear term displacement function using the Taylor series to solve this problem. This technique can also be used to address more practical changes (Scaramuzza and Fraundorfer, 2012).

3.3.4. Data association

Data association in SLAM can simply be interpreted as a problem of feature correspondence, which recognizes two features observed as being from the same physical object in the world at different locations and time points. When a robot returns to the trajectory's starting point, two common implementations of this data association are to suit two successive scenes and close a loop of a long trajectory.

Therefore, it is important to have very robust features, even under poor lighting conditions or from different points of view, in order to succeed in solving the correspondence problem. The use of vision sensors provides the opportunity to extract landmarks when considering 2D and 3D data, makes it possible to choose more robust features (Saez et al., 2006).

3.3.5. Random sample consensus

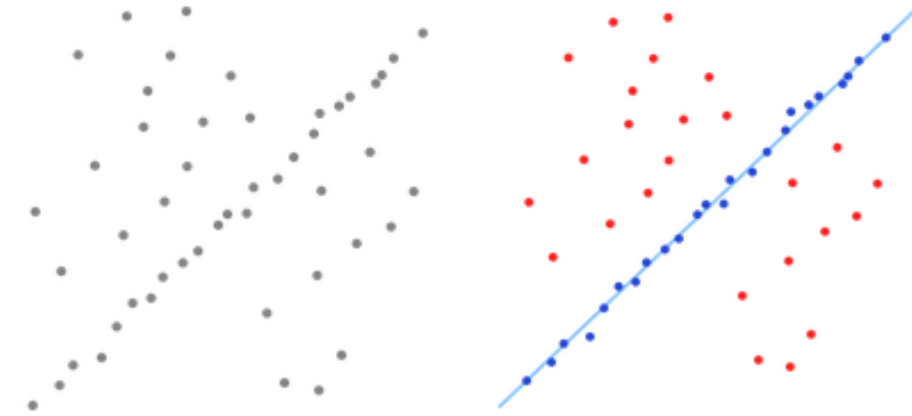
Random sample consensus (RANSAC) is an iterative method to estimate the parameters of a mathematical model from a set of outliers containing observed data (Figure 3.4) where the purpose is to prevent outliers influence the estimated values. Therefore, it can also be viewed as a method of outlier detection. It produces a logical result only with a certain probability. On the other hand, in order to obtain a higher probability more iterations are required, because of it is a non-deterministic algorithm. The algorithm was first published by Fischler and Bolles (1981). To solve the problem of position determination, they used RANSAC, where the objective is to decide the points in the space that project onto an image using a set of landmarks with known places.

The basic assumption is that the data consists of "inliers," i.e. data whose distribution can be represented by a set of parameters of the model, even though they may be subject to noise, and "outliers" that do not fit the model with the data. Outliers can occur, for example, from extreme noise values, or from incorrect measurements, or from incorrect data interpretation

assumptions. RANSAC also assumes that there is a procedure that can approximate the parameters of a model that represents or matches these results optimally, in the sense of a set of inliers.

Algorithm of RANSAC (Derpanis, 2005):

1. Select at random the minimum number of points needed for the model parameters to be calculated.
2. Resolve the model parameters.
3. Then all other data is checked against the fitted model. Those points that suit well with the approximate model are considered as part of the consensus set according to some model-specific loss function.
4. If too many points have been categorized as part of the consensus collection, the estimated model is relatively strong.
5. Afterwards, repeat steps 1-4 maximum of N times (all members of the consensus set).



A data set with many outliers for which a line has to be fitted.

Fitted line with RANSAC; outliers have no influence on the result.

Figure 3.4. Random sample consensus (Wikipedia, 2020)

3.3.6. Loop closure

Loop Closure Detection (LCD) is a key component of SLAMs that can be described as a method that tries to find a match between the current observation and the previously visited location. The robust LCD would greatly reduce the calculated trajectory and plot's drift error. The efficiency of the LCD becomes more and more important as the size of the map increases, but the time taken to find the correct loop closure candidates will become more complicated and more computational. The LCD has gained significant attention in recent years as an integral part of the SLAM issue (Wang et al., 2019).

In general loop-closure detection algorithms can be classified into three groups: map-to-map, image-to-map, and image-to-image (Andrey and Yakovlev, 2017).

The map-to-map approach is very intense in terms of results, as it deals with large amounts of each iteration when comparing sub-maps. As a consequence, it scales poorly in large environments.

The approach to image-to-map is fast and precise, but in practice it is very memory intensive since both the point-cloud map and all the image features need to be processed.

The image-to-image loop-closure scales well to large environments and, with feature-based methods, can be computed easily, but relies heavily on a vocabulary. Thus, it can be concluded that a combination of different methods is more optimum to achieve higher performance while high degree of accuracy and robustness.

4. VISUAL-INERTIAL SLAM

Due to extraneous factors such as motion blur, lack of or too much sunlight, and blocked cameras, visual measurements may be blurred or unavailable. The measurements taken from a wide variety of sensors, such as sonar, radar, lidar, encoders and touch switches, can support information used to solve the visual SLAM problem. Body acceleration and rotation speeds are such sensing modalities that is of great interest to visual SLAM. Typically, gyroscopes and accelerometers in an IMU evaluate these quantities. IMU measurements are carried out at discrete points in time, similar to visual measurements of landmark forecasts, but are measurements of continuous variables. In general, IMU measurements are taken at higher frequencies (100-1000Hz) than visual measurements (30-60Hz).

The graphical model for illustrating the visual-inertial batch problem of the SLAM is shown in figure 4.1. In the model, a new kind of constraint can be seen, linking poses directly together by integrating inertial measurements. In its tightly coupled form, this graph shows the visual-inertial SLAM problem, where visual and inertial measurements are considered at the same time and all sensor states are optimized. This is due primarily to the existence of new sensor states when inertial measurements are considered.

The second derivative of the amount of interest for optimization is determined by accelerometers: the camera's path. In order to use inertial measurements, speeds must be calculated for this purpose. Similarly, sensor biases are influenced by IMU measurements that have to be constantly estimated as they shift depending on extraneous variables such as temperature. If these parameters are not estimated, IMU measurements are restricted to their use in loosely coupled formulations, where they are used as inclination or rotation signs, but not fully incorporated with visual measurements.

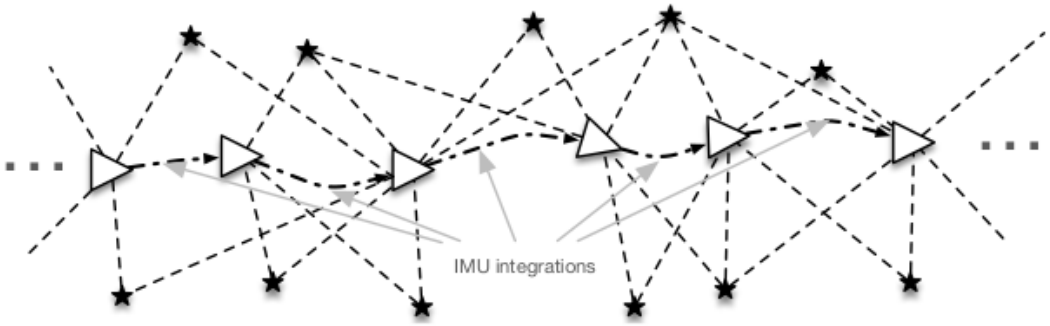


Figure 4.1. Graphical model of visual-inertial SLAM (Keivan, 2009)

4.1. Sensor Fusion

In order to provide a comprehensive and complete picture of the environment or process of interest, sensor fusion is the process of combining information from a number of different sources/sensors. In such a way the resulting information is less vague than would have been possible if these sources were used individually. In autonomous systems and mobile robotics, sensor fusion methods have a very important role. Theoretically, data fusion systems allow information to be combined in order to provide sufficient knowledge for the complexities and integrity of decision-making and autonomous execution (Khalid et al., 2015).

4.1.1. Sensor types

First of all, there are many ways for obtaining environmental measurements in SLAM applications. For this reason, lidars and cameras are most widely used sensors. The most popular cameras used for SLAM systems are monocular and stereo cameras (Figure 4.2).

In terms of the measurement properties for various use cases, there are pros and cons of each sensor type. The benefit of collecting scale data is that Lidar's and the stereo cameras assess depth in the area. In the data association procedure of any SLAM scheme, this property will be very useful. But on the

other hand, due to the type of active sensor and transmitting rays to the atmosphere in the measurement process, for example Lidar's has the downside of stopping the aircraft from being stealthy.

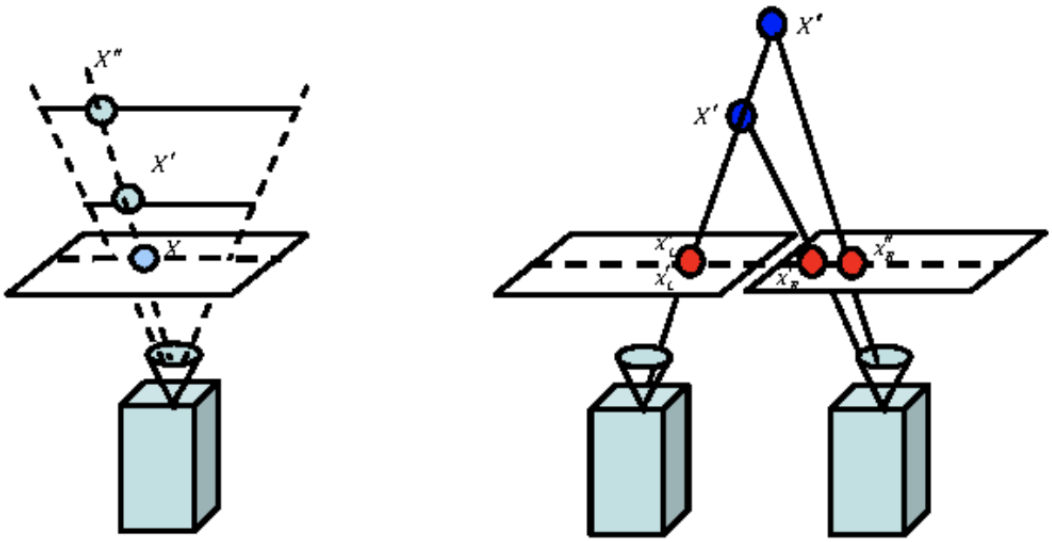


Figure 4.2. Monocular camera (left) and stereo cameras (right) (Jung et al., 2005)

Stereo and monocular cameras are other options for camera usage. Stereo cameras are useful and versatile for small-scale applications, mainly for indoor SLAM applications. They are not usable in large distance settings, since the baseline camera and the distance to the landmark ratio should not be too small to obtain accurate depth information in stereo cameras.

On the other hand, in calculating motion, the monocular camera eliminates the effect of calibration errors. One of the main benefits of using monocular cameras is that they are cheaper and simpler to deploy than stereo cameras. Although monocular cameras suffer from scale uncertainty, IMU in visual-inertial navigation helps to solve this problem.

4.1.2. Fusion methods: tightly coupled – loosely coupled

In Visual-Inertial algorithms, the design of a structure depending on whether the fashion is tightly or loosely coupled is another aspect to be calculated.

Both camera and IMU measurements are determined separately in the case of loosely coupled fusion and, in the end, fusion is applied to their calculation. As the integration of visual and inertial information is not considered at the raw data stage in a loosely coupled process, this makes the system incapable of correcting vision drifts merely by approximation.

On the other hand, to estimate the position of the platform, raw measurements of the camera and the IMU were used together in a tightly coupled process. A tightly linked approach takes more energy for computation, but it is more efficient than a loosely linked approach.

4.2. VINS-Mono

VINS-Mono is a monocular real time visual-inertial SLAM platform proposed by Aerial Robotics Group of HKUST in 2017. To provide an extremely accurate visual inertial odometer, it uses an optimized sliding window.

The VINS-Mono algorithm is briefly explained in this section. The non-linear optimization-based approach is applied to obtain visual-inertial odometry by fusing the IMU measurements and the visual camera measurements in the VINS-Mono algorithm, tightly coupled. It is time now to clarify some general concepts and concepts related specifically to VINS-Mono.

Figure 4.3 is the general structure for the visual-inertial pose estimation. Most Visual-Inertial Odometry (VIO) applications use a visual data camera and an IMU sensor. The inputs are the camera image and the IMU data containing acceleration and angular velocity measurements, in the context provided in Figure 4.3. Outputs are calculated by the 6-Degrees-of-Freedom (DoF) platform

based on these inputs. As soon as the extraction of the features and pre-integration of IMU is completed, the system starts. Inertial and visual poses for the initial estimation of the platform are combined with their pose, velocity, gyroscope bias and gravity vector. By the visual-inertial odometry algorithm, these values are modified iteratively. Finally, the 6-DoF pose of the platform can be obtained.

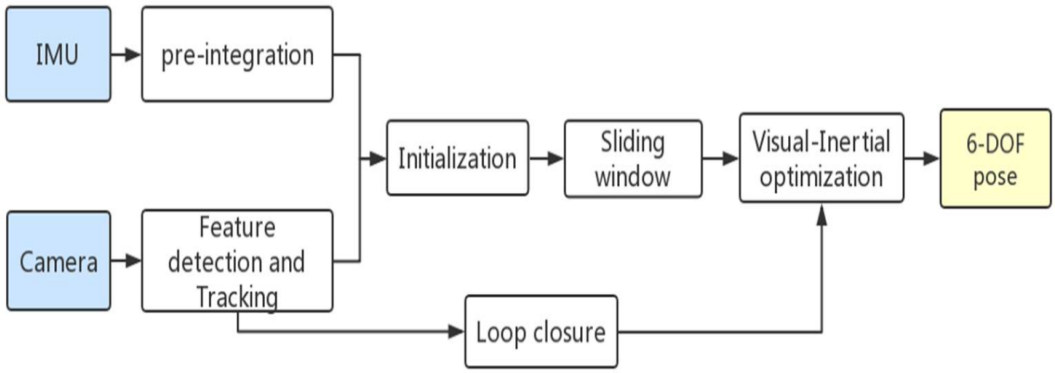


Figure 4.3. General structure of visual-inertial pose estimation (Qin et al., 2018)

For camera systems, two techniques are applied. The first uses a stereo camera to construct a system, and the other uses a monocular camera. The stereo camera approach requires a long duration for reliable results. The baseline is the difference between the stereo camera's two lenses and the depth range that can be observed and the resolution of the depth. For that purpose, for airborne applications, it requires a broad baseline that is simply not practical. The precise structure of the VINS-Mono algorithm is shown in Figure 4.4. It starts with the preprocessing stage of measurement in the VINS-Mono algorithm structure. For the popular VIO and Visual SLAM algorithms as mentioned, this section is common. A feature that extracts camera data and IMU measurements between two consecutive camera frames is included in the preprocessing measurement section. At initialization, the position values, velocity, vector, gravity, gyroscope bias and 3D position of the environmental characteristics are obtained. At a further point, pre-integrated IMU measurements and feature observations are merged into the VIO module to re-locate the system.

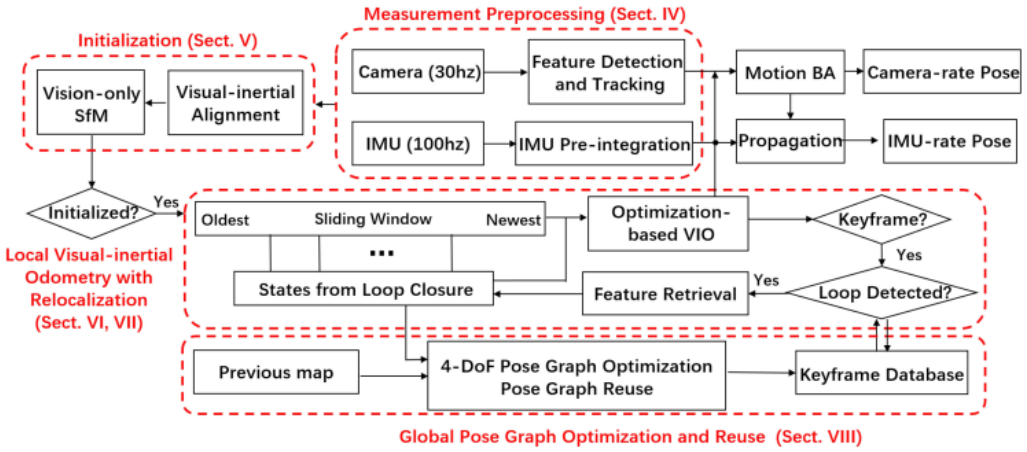


Figure 4.4. Detailed structure of the VINS-Mono (Qin et al., 2018)

Finally, to remove the drift, the pose graph optimization module is used. In the suggested framework, re-localization and graph optimization operate concurrently. Loop closures that classify the places already visited are identified in the re-location process. The entire pose graph is modified according to the correspondence between the loop closure frame and the current frame, based on the loop closures observed. On the other hand, in the graph optimization section, the residual errors of the edges between frames are minimized. The relative transformation between two frames is finally edges. The pose graph shifts and has become globally consistent as a result of this optimization. The pre-integration IMU and the IMU sample trajectory are shown in Figure 4.5. The camera is correlated with the characteristics seen in the field. Aligning the visual structure with the pre-integration of the IMU is the basic idea.

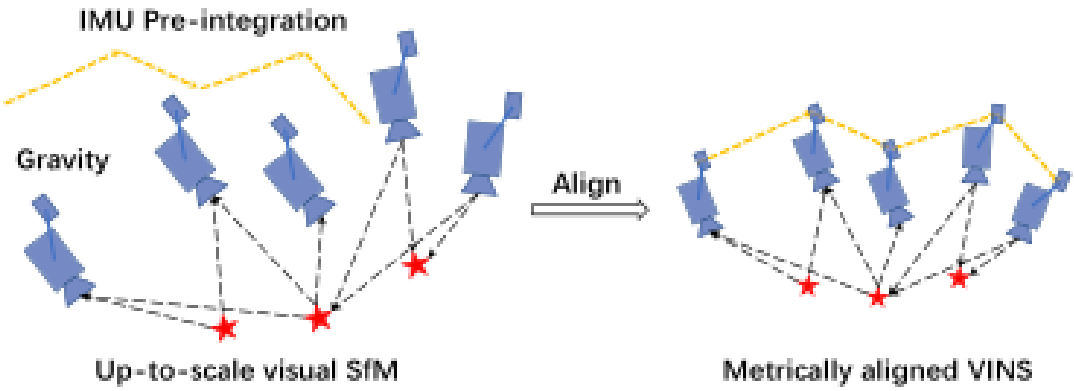


Figure 4.5. IMU pre-integration (Qin et al., 2018)

After the initialization process, Figure 4.6 shows the flow. Centered on tightly coupled monocular VIO for state estimation, this approach is called "sliding window". The strategy for marginalization is defined in Figure 4.7. In this technique, the algorithm checks whether or not a mainframe is the second last frame, then it is the oldest frame and marginalized.

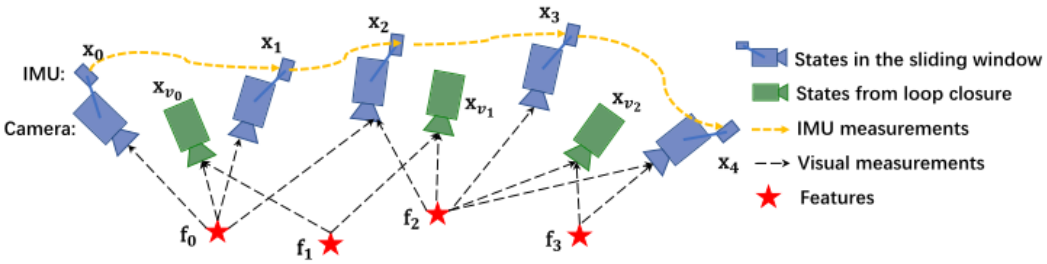


Figure 4.6. Sliding window approach (Qin et al., 2018)

As mentioned above, marginalized visual and inertial ratios are used. But if the last frame of the second is not a mainframe, the visual dimensions would be omitted. But at the IMU pre-integration stage for additional frames, inertial measurements are still retained. In the sliding window process, re-locating pose graph optimization and loop closure are shown in Figure 4.8. If the next

keyframe detects a loop, the keyframe is marginalized and re-located. All poses are optimised in another thread, according to the re-location.

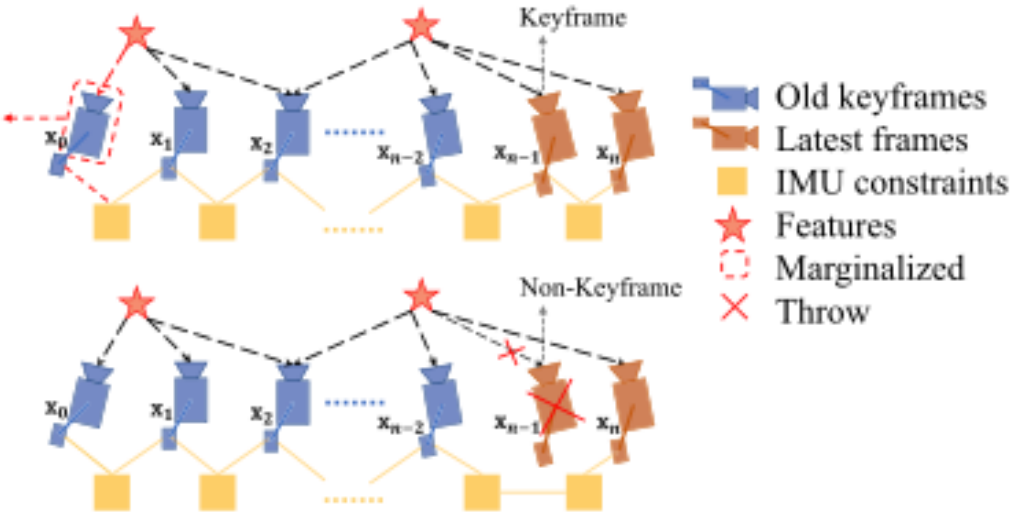


Figure 4.7. Marginalization step (Qin et al., 2018)

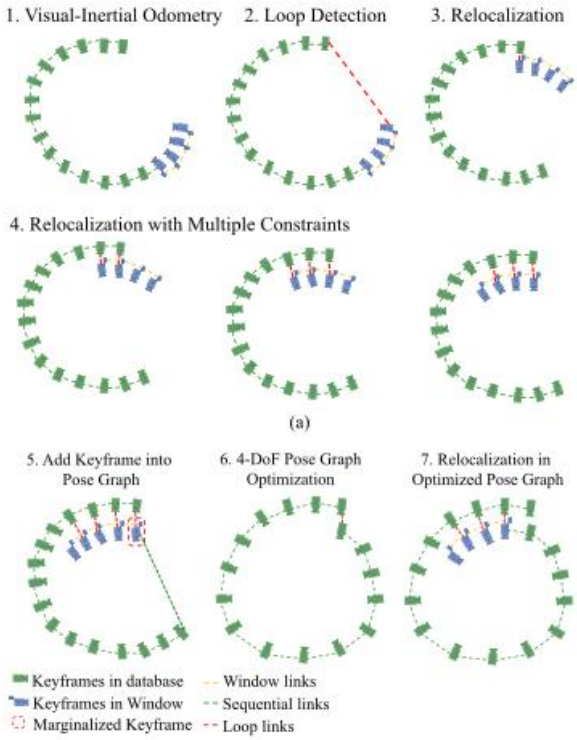


Figure 4.8. Re-localization and graph optimization (Qin et al., 2018)

4.3. ORB-Slam3

ORB-SLAM3 is a Visual-Inertial SLAM system (Campos et al., 2020) built on ORB-SLAM (Mur-Artal et al., 2015), ORB-SLAM2 (Mur-Artal and Tardos, 2017) and ORB-SLAM Visual Inertial by Mur-Artal and Tardos (2017). Figure 4.9 shows the main parts of the ORB-SLAM3. ORB-SLAM3 claims to be the best visual-inertial system in the literature.

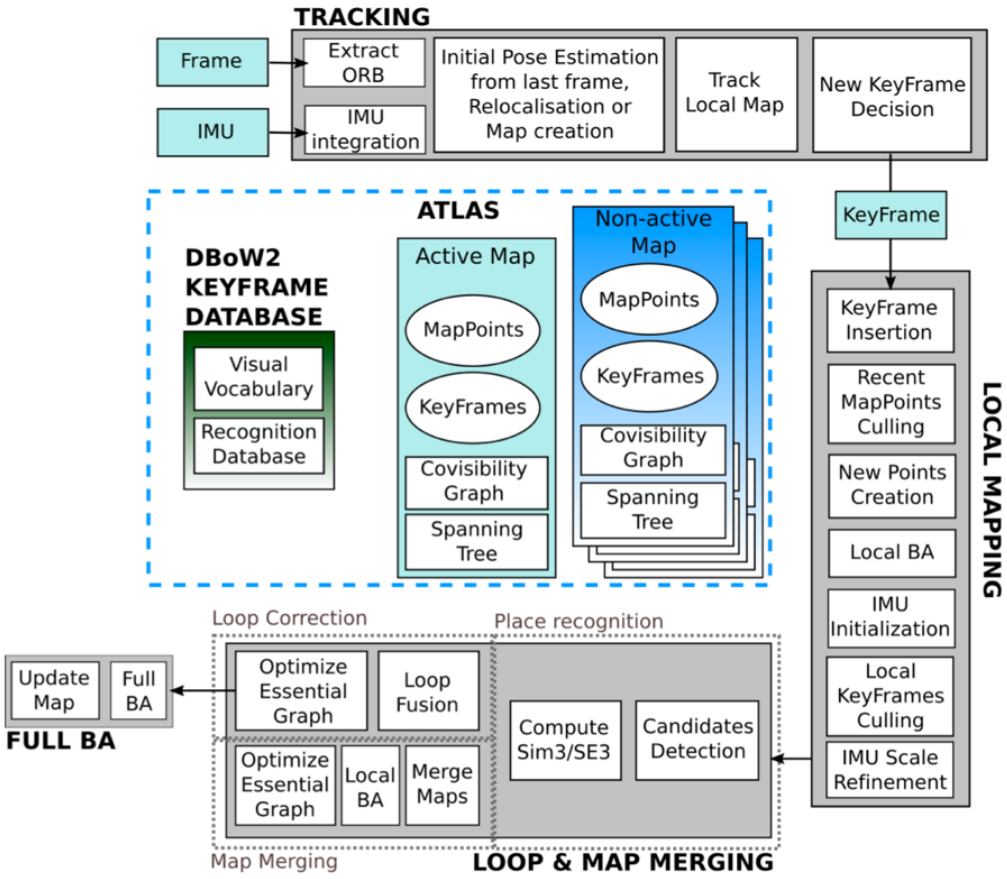


Figure 4.9. Main components of ORB-SLAM3 (Campos et al., 2020)

ORB (Rublee et al., 2011) was selected for feature extraction, as shown in Figure 4.10. Although it is invariant to the point of view, ORB is extremely quicker to compute and match. This makes it possible to comply with wide baselines, enhancing the precision of the Bundle Adjustment (BA).

A multi-map representation consisting of a variety of disconnected maps is the Atlas. There is an active map where the tracking thread locates the incoming frames, and with new keyframes, the local mapping thread constantly optimizes and expands.

The tracking thread processes the sensor information and calculates in real-time the position of the current frame with respect to the active map, thus minimizing the reprojection error of the corresponding map characteristics. It also decides whether it will become a keyframe for the current frame. Body velocity and IMU bias are determined in visual-inertial mode by using inertial residuals in optimization. The tracking thread tries to move the current frame to all Atlas maps when tracking is lost.

Using visual or visual-inertial bundle adjustments, the local mapping thread adds the keyframes and points to the active map, removes redundant ones, and refines the map, running in a local keyframe window near the current frame. For every new keyframe, Closing Loop searches for loops. If a loop is located, it measures a transformation of similarity that tells the cumulative drift in the loop. Both sides of the loop are centered at the end, and the duplicate points are joined together.

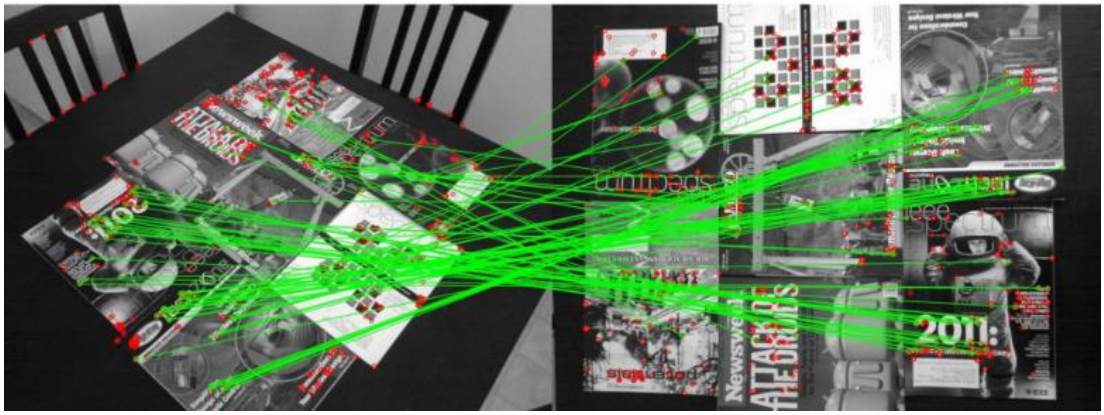


Figure 4.10. Matcing result using ORB (Rublee et al., 2011)

5. MATERIAL AND METHOD

VINS-Mono and ORB-SLAM3 were compared in terms of availability for aviation. We showed trajectories of VINS-Mono and ORB-SLAM3 initally. Eventually a numerical analysis conducted to demonstrate the accuracy of our systems by Root Mean Square Error (RMSE).

5.1. Dataset

VINS-Mono and ORB-SLAM3 were tested using a visual-inertial dataset of the EuRoc MAV (Burri et al., 2016). Two datasets were provided. The first dataset was recorded in a large machine hall and was intended to test visual-inertial motion estimation algorithms or SLAM frameworks. A 3D location was provided by a laser tracker for ground truth. On the other hand, the second dataset was recorded in the Vicon room fitted with a motion capture device with an approximate size of 8mx8.4mx4m.

5.2. Evaluation

The EuRoC MAV visual-inertial dataset provides 11 sequences. 8 of them were selected for commenting on specific conditions. Each sequence has different environment specifications as given in Table 5.1. Experiments were executed on an Intel Xeon (R) CPU E5-1620 v4, at 3.50 GHz with 32 GB memory.

The trajectories of the sequence (MH_04_difficult) with their ground-truth are presented in the Figure 5.1 in order to show that how the trajectories were examined. Figure 5.1 shows the trajectory of selected sequences with VINS-Mono (a) ORB-SLAM3 (b) and their ground truth alignments respectively (c) and (d). The Root Mean Square Errors (RMSE) of selected sequences in EuRoC datasets were evaluated by two error metrics, Absolute Pose Error (APE) (Table 5.2) and Relative Pose Error (RPE) (Table 5.3) using evo-tool[1]. The APE is well-suited to visual SLAM systems output measurements. The RPE on the other

[1] github.com/MichaelGrupp/evo

hand, is well-suited to calculate the drift of a visual odometry system, such as drift per second. ORB-SLAM3 outperformed for all cases.

APE compares the trajectory of a vehicle to the actual trajectory (ground truth), as reconstructed by an algorithm using real sensor data as its input. By comparing the absolute distance between the estimated trajectory and the ground truth, global consistency can be measured.

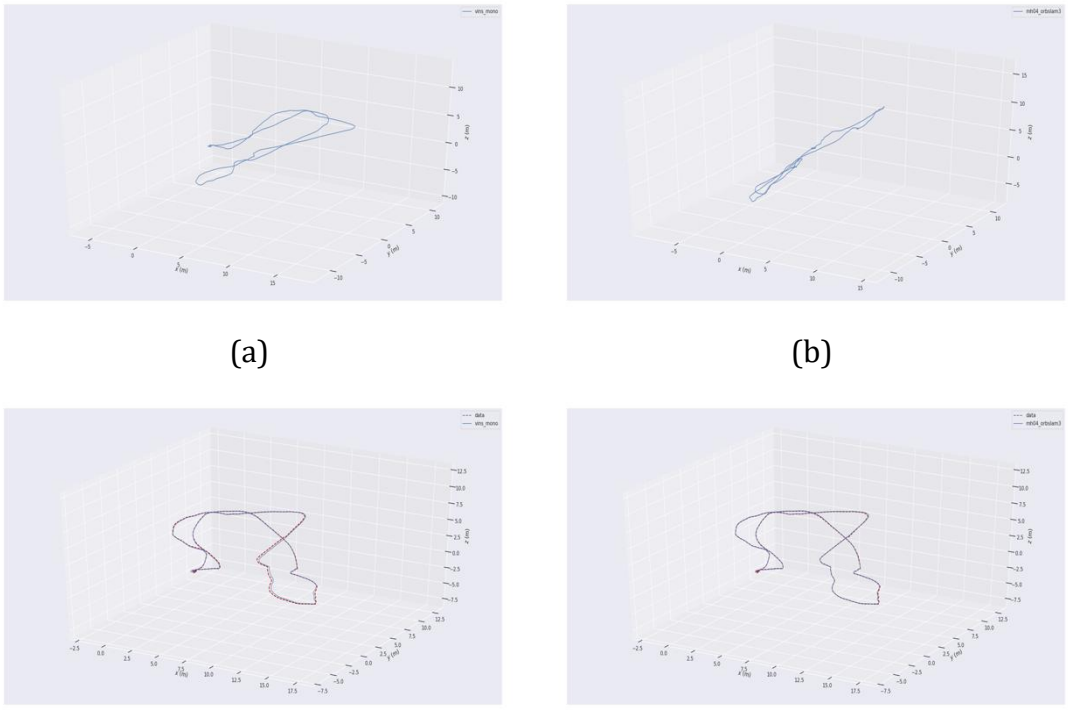
By comparing the reconstructed relative transformations between nearby poses to the actual relative transformations (ground truth), RPE tests the accuracy of a SLAM outcome, as reconstructed by an algorithm using real sensor data as its input.

Table 5.1. Environment specifications of selected datasets

Name	Distance/Duration	Average Velocity/Angular Velocity	Conditions
MH_01_easy	80.6m 182s	0.44ms^{-1} 0.22rads^{-1}	Good texture, bright scene
MH_02_easy	73.5m 150s	0.49ms^{-1} 0.21rads^{-1}	Good texture, bright scene
MH_03_medium	130.9m 132s	0.99ms^{-1} 0.29rads^{-1}	Fast motion, bright scene
MH_04_difficult	91.7m 99s	0.93ms^{-1} 0.24rads^{-1}	Fast motion, dark scene
V1_01_easy	58.6m 144s	0.41ms^{-1} 0.28rads^{-1}	Slow motion, bright scene
V1_02_medium	75.9m 83.5s	0.91ms^{-1} 0.56rads^{-1}	Fast motion, bright scene
V2_01_easy	36.5m 112s	0.33ms^{-1} 0.28rads^{-1}	Slow motion, bright scene

V2_02_medium	83.2m 115s	0.72ms^{-1} 0.59rads^{-1}	Fast motion, bright scene

When looking at RMSEs of APE, the first thing is that VINS-Mono gave the same error rate in the sequences of MH_01_easy and MH_02_easy under the same ambient conditions but with different flight durations. On the other hand, the accuracy rate of ORB-SLAM3 is slightly higher in the MH_01_easy series, where the flight takes longer. While this efficiency could be linked to better optimization of the pose on long-term flights, depending on the flight period, it will be difficult to vary accuracy rates. Testing the efficiency of ORB-SLAM3 in systems where flight missions such as airplanes and helicopters will change regularly and error rates are supposed to be close to zero will be costly for different topics, such as time and workload. When the ORB-SLAM3 or similar system is to be integrated into these aircraft, further developments and studies are needed in this respect.



(a) (b)

(c) (d)

Figure 5.1. Trajectories

How the dark environment affects the efficiency of the algorithms can not be applied to because of the change not only in the conditions but also in the distance between sequences for MH_03_medium and MH_04_difficult. The capabilities of these algorithms with the EuRoC MAV dataset in the dark environment can not be fully understood if the impact of the light level on the environment for performance is believed to be resolved at the hardware level (high sensitivity cameras). However, if different types of sensors are used in air vehicles, such as planes and helicopters, algorithms need to be compared not only routinely, but also according to hardware variations.

Table 5.2. RMSE of APE (Meters)

	VINS-Mono	ORB-SLAM3
MH_01_easy	0.182	0.016
MH_02_easy	0.182	0.065
MH_03_medium	0.404	0.041
MH_04_difficult	0.393	0.110
V1_01_easy	0.144	0.050
V1_02_medium	0.311	0.013
V2_01_easy	0.121	0.041
V2_02_medium	0.275	0.013

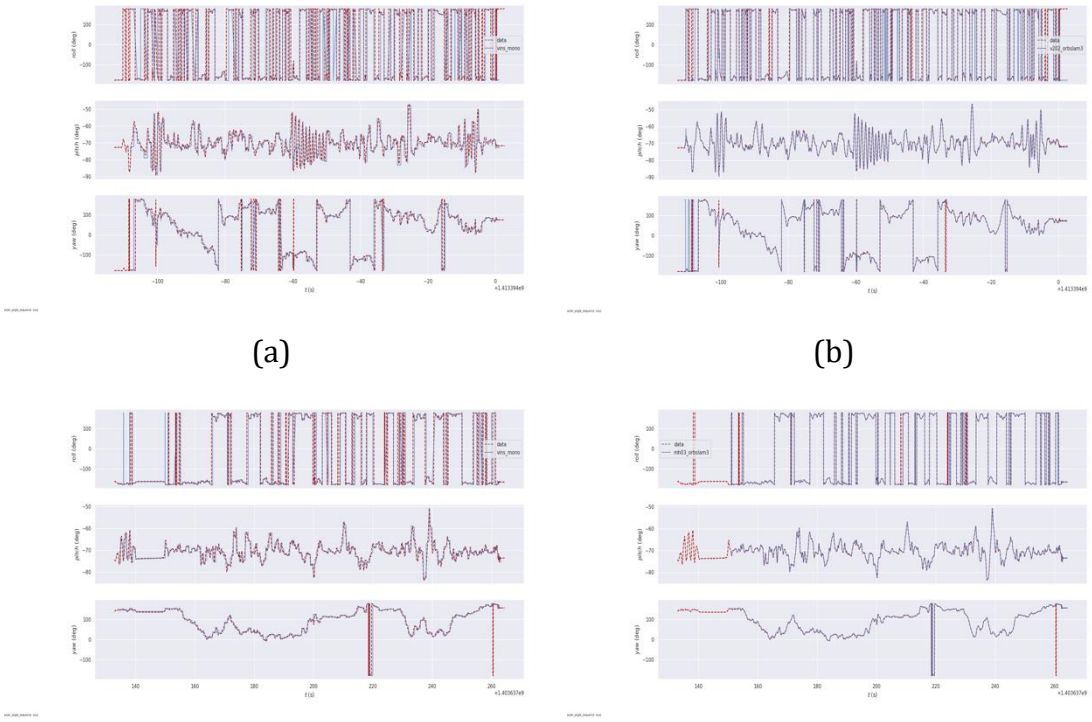
In the V2_01_easy and V2_02_medium sequences where flight times are similar to each other and only the speed of motion of the drone varies in the environment, we found that when VINS-Mono performed quick maneuvers, the error rate increased approximately 2 times. ORB-SLAM3, on the other hand, decreased the error rate even further. In the case of moving objects in the Vicon Room (V1_01_easy, V1_02_medium, V2_01_easy, V2_02_medium) sequences, it is shown that ORB-SLAM3 results in better matching and tracking features.

Table 5.3. RMSE of RPE (Meters)

	VINS-Mono	ORB-SLAM3
MH_01_easy	0.1885	0.0048
MH_02_easy	0.1985	0.0054
MH_03_medium	0.4130	0.0063
MH_04_difficult	0.4020	0.0078
V1_01_easy	0.1489	0.0038
V1_02_medium	0.3042	0.0062
V2_01_easy	0.1187	0.0044
V2_02_medium	0.2376	0.0078

We can think of the pitch as the up and down motion of the aircraft. Pitch regulation is what most precisely distinguishes an aircraft's activity in the sky from any earth-linked vehicle. This includes the act of manoeuvring an aircraft on the runway. And, in simple terms, yaw is the perpendicular movement of the plane's nose to the wings (left or right). The roll is the movement of the aircraft that rocks back and forth. In a roll, the airplane's wings shift up and down. Although the left wing is tilted up, the right inevitably points down.

When we examine the roll, pitch yaw angles given in Figure 5.2 with sequences of V2_01_medium and MH_03_medium. (a) and (b) represents VINS-Mono and ORB-SLAM3 in V2_01_medium and (c) and (d) shows the MH_03_medium in VINS-Mono and ORB-SLAM3 angles respectively. We can see that ORB-SLAM3 highly overlaps with the actual angle (ground truth) values at pitch and yaw angles. Although the accuracy of the roll angles is not as high as the others, we observe another advantage of ORB-SLAM3 with respect to the VINS-Mono algorithm.



(a) (b)

(c) (d)

Figure 5.2. Roll, pitch, yaw angles with ground truth

ORB-SLAM3 gives better performance in any experiment applied to. ORB-SLAM3 provides much higher accuracy values in RPE error values than APE error values when the error metrics considered. While ORB-SLAM3 gives approximately 2 times more precision than VINS-Mono in terms of trajectory accuracy and this rate increases even more while considering it on the basis of RPE. In evaluating visual odometry efficiency, taken into consideration of RPE, ORB-SLAM3 also performed better at localization.

6. CONCLUSION AND DISCUSSION

Perhaps it is aviation that has the least feature tolerance of any system in the world. Navigation is one of the most important subsystems when we consider many parameters such as taking off, landing and going on its path. Based on this, answer of the question how the navigation system in aircraft can be more robust is important. Experiments applied through research bring to come Simultaneous Localization and Mapping systems these have important role in the field of robotics. The fact that these systems contain a passive sensor such as a camera (safe against jamming attacks) and have proven application areas (drone, AR, VR) are very impressive. Therefore, in this study, the performances of the two most robust frameworks in the literature, VINS-Mono and ORB-SLAM3, were compared and inquired about their applicability in aviation. ORB-SLAM3 gave good results with the help of its new fast and high accurate IMU initialization technique. It was effective in the integration of the IMU by detecting the features, particularly in cases where the drone was moving quickly. But, it produces nearly 10 times higher error only in sequences where the environment is darker. However, the datasets used in research were indoor environments and generally contained spaces with a high number of features. Although some data sets that contain outdoor images found, the peformance could not be evaluated because the ground truth values were not measured correctly and were only used to run the algorithm. Since procuring our own hardware also requires a serious budget, we could not provide the environmental conditions and evaluate performance with indoor data sets as mentioned before. Considering these problems, before deciding on the correct algorithm and testing on aircraft, data sets containing outdoor environment should be created, parameters that will affect corresponding process such as less feature, more moving objects, and the parallax angle should be provided in detail. It would be better to explore and study algorithms in high speeds, high altitudes and different weather conditions such as rainy. In addition to environmental problems, it will be a problem that the cameras on the aircraft generally do not stable pose in one direction and in helicopters, usually as a

result of vibration, they are often facing the sky and mountains where the features are low.

The fact that algorithms produce different accuracy values at different conditions will cause them fall below admissible security levels for aviation. Therefore, we plan to develop and use the GPS independent navigation system, which we research in our future studies, as an auxiliary navigation tool from the moment it is cut, not assuming that GPS does not exist at all. A second solution is to consider using unmanned aerial vehicles, which hardware are closer to drones in their first use, and in low-altitude flights, we can limit the scope of our algorithm and find parts that can be improved more quickly.

Beside the problems and possible solutions discussed above, ORB-SLAM3 seems better choice at the beginning of these researches, with feature matching in fast motion, tracking capability and successful optimizations in long flights. Finally, in the light of new technologies learning-based visual-inertial systems can become widespread and need to be focused on.

REFERENCES

Aeronautics Guide, 2017. Global Positioning System (GPS) in Aviation. Retrieved: 12.01.2021. https://www.aircraftsystemstech.com/2017/05/global-positioning-system-gps.html

Akca, M., 2020. HAVACILIK KAZASI VE PİLOT HATASI KAVRAMI ÜZERİNE BİR DEĞERLENDİRME. The Journal of Social Science. 10.30520/tjsosci.682699.

Andrey, B., Yakovlev, Konstantin. 2017, Original Loop-closure Detection Algorithm for Monocular vSLAM.

Bailey, T., Durrant-Whyte, H. 2006, Simultaneous localization and mapping (SLAM): Part I, IEEE Robotics and Automation Magazine, 13(3), 108-117.

Bay, H., Tuytelaars, T., Van Gool, L., 2006, SURF: Speeded up robust features, Computer Vision-ECCV 2006, 3951, 404-417.

Blöchliger, F., Fehr, M., Dymczyk, M., Schneider, T., Siegwart, R. 2018, Topomap: Topological Mapping and Navigation Based on Visual SLAM Maps, IEEE International Conference on Robotics and Automation (ICRA), 1-9.

Bruce, L., Kanade, T. 1981, An Iterative Image Registration Technique with an Application to Stereo Vision (IJCAI).

Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., Omari, S., Achtelik, M., Siegwart, R. 2016, The EuRoC micro aerial vehicle datasets, International Journal of Robotic Research.

Campos, C., Elvira, R., Rodriguez, J. J. G., Montiel, J. M. M., Tardos, J. D. 2020, ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM, 1-15.

Clark, R. 2017, Visual-inertial odometry, mapping and re-localization through learning, PhD thesis, University of Oxford.

Daly, D., 2021. A Not-So-Short History of Unmanned Aerial Vehicles (UAV). Retrieved: 01.03.2021, https://consortiq.com/short-history-unmanned-aerial-vehicles-uavs

Davison, A. 2003, Real-time simultaneous localisation and mapping with a single camera, Ninth IEEE International Conference on Computer Vision, 1403-1410.

Derpanis, K. 2005, Overview of the RANSAC Algorithm.

Dissanayake, M. W. M. G., Newman, P., Clark, S., Durrant-Whyte, Csorba, M. 2001, A solution to the simultaneous localization and map building (SLAM) problem, 17(3), 229-241.

Hahnel, D., Burgard, W., Fox, D., Fishkin, K., Philipose, M. 2004, Mapping and localization with RFID technology, In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA).

Fischler, M., Bolles, R. 1981, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Commun, ACM 24.

Flight Literacy, 2020. Dead Reckoning. Retrieved: 09.10.2020. https://www.thebalancecareers.com/how-do-pilots-navigate-282803

Furgale, P., Barfoot, T. D. 2010, Visual teach and repeat for long-range rover autonomy, Journal of Field Robotics, 27, 534-560.

GPS, 2006. Aviation. Retrieved: 29.11.2020. https://www.gps.gov/applications/aviation/

Everaerts, J., 2008, The use of unmanned aerial vehicles (UAVs) for remote sensing and mapping, The International Archives of the Photogrammetry Remote Sensing and Spatial Information Sciences, 37.

Harris, C., Stephens, M., A Combined Corner and Edge Detector, Alvey Vision Conference, 1988.

Henry, P., Krainin, M., Herbst, E., Ren, X., Fox, D., 2012, RGB-D mapping: using kinect-style depth cameras for dense 3D modelling of indoor environments, Int. J. Robot. Res, 31(5), 647-663.

Houston, S., 2019. How Pilots Use Air Navigation to Fly. Retrieved: 29.11.2020. https://www.thebalancecareers.com/how-do-pilots-navigate-282803

Ivan, K., Sinisa, S., 2015, Improving the Egomotion Estimation by Correcting the Calibration Bias, VISAPP 2015 – 10th International Conference on Computer Vision Theory and Applications, 3, 347-356.

Jones, M. H., 2019. Calibration Checkerboard Collection. Retrieved: 12.01.2021. https://markhedleyjones.com/projects/calibration-checkerboard-collection

Jung, H. G., Lee, Y., Kim, D. S., Yoon, P. J. 2005, Stereo Vision Based Advanced Driver Assistance System.

Khalid, Y., Hadiashar, A., Reza, H. 2015, An Overview to Visual Odometry and Visual SLAM: Applications to Mobile Robotics, Intelligent Industrial Systems.

Keivan, N. 2009, Monocular Visual-Inertial SLAM and Self Calibration for Long Term Autonomy, PhD thesis, University of Queensland.

Li, G., Baker, S. P., Grabowski, J. G., & Rebok, G. W., 2001. Factors associated with pilot error in aviation crashes. Aviation, space, and environmental medicine, 72(1), 52-58.

Jones, E., Soatto, S. 2011, Visual-inertial navigation, mapping and localization: a scalable real-time casual approach, Int. J. Robot. Res, 30(4), 407-430.

Lowe, D. 2004, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis, 60(2), 91-110.

Lu, F., Milios, E. 1997, Globally Consistent Range Scan Alignment for Environment Mapping, Autonomous Robots, 4, 333-349.

Marsh, A. K. 2016. Technique-Pilotage and dead reckoning. Retrieved: 12.01.2021. https://www.aopa.org/news-and-media/all-news/2016/march/flight-training-magazine/technique-pilotage-and-dead-reckoning

Mishra, A., 2019, Navigation: Advancements & Benefits.

Montemerlo, M., Thrun, S., Koller, D., Wegbreit, Ben. 2002, FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Probleem, Proceedings of the National Conference on Artificial Intelligence.

Mur-Artal, R., Montiel, J. M., Tardos, J. D. 2015, ORB-SLAM: A Versatile and Accurate Monocular SLAM System, IEEE Transactions on Robotics, 31(5), 1147-1163.

Mur-Artal, R., Tardos, J. D. 2017, ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras, IEEE Transactions on Robotics, 33(5), 1255-1262.

Mur-Artal, R., Tardos, J. D. 2017, Visual-Inertial Monocular SLAM With Map Reuse, IEEE Robotics and Automation Letters, 2(2), 796-803.

Nister, D., Naroditsky, O., & Bergen, J. (2004). Visual odometry. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition.

Paz, L., Pinies, P., Tardos, JD., Neira, J. 2008, Large-scale 6DoF SLAM with stereo-in-hand, IEEE Trans Robot, 24(5), 946-957.

Qin, T., Li, P., Shen, S. 2018, VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator, IEEE Transactions on Robotics, 34(4), 1004-1020.

Revfine, 2021. Aviation Industry: All You Need to Know About the Aviation Sector. Retrieved: 28.02.2021. https://www.revfine.com/aviation-industry

Rivera-Rubio, J., Alexiou, I., Bharath, A., Secoli, R., Dickens, L., Lupu, E. 2014, Associating locations from wearable cameras, In Proceedings of the British Machine Vision Conference, BMVA Press.

Rublee, E., Rabaud, V., Konolige, K., Bradski, G. 2011, ORB: An efficient alternative to SIFT or SURF, International Conference on Computer Vision, Barcelone, 2564-2571.

Saez, J. M., Hogue, A., Escolano, F., Jenkin, M. 2006, Underwater 3D SLAM through entropy minimization, IEEE International Conference on Robotics and Automation. 3562-3567.

Scaramuzza, D., Fraundorfer, F. 2011, Visual Odometry Part I: The First 30 Years and Fundamentals, IEEE Robotics and Automation Magazine, 18(4), 80-92.

Scaramuzza, D., Fraundorfer, F. 2012, Visual Odometry: Part II: Matching, Robustness, Optimization, and Applications, IEEE Robotics and Automation Magazine, 19(2), 78-90.

Shi, J., Tomasi, C. 1994, Good features to track. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 593-600.

Skybraryaero, 2021. Autopilot – SKYbrary Aviation Safety. Retrieved: 01.03.2021. https://www.skybrary.aero/index.php/Autopilot

Smith, R. C., Cheeseman, P. 1986, On the Representation and Estimation of Spatial Uncertainty, The International Journal of Robotics Research, 5(4), 56-68.

Sun, K., Mohta, K., Pfrommer, Bernd., Watterson, M., Liu, Sikang., Mulgaonkar, Yash., Taylor, J. C., Kumar, V., 2018, Robust Stereo Visual Inertial Odometry for Fast Autonomous Flight, IEEE Robotics and Automation Letters, 3(2), 965-972.

Thrun, S. 2002. Probabilistic robotics. Communications of the ACM, 45(3), 52–57.

Thrun, S., Montemerlo, M. 2006. The graph SLAM algorithm with applications to large-scale mapping of urban structures, International Journal of Robotics Research, 25(5–6), 403–429.

Vision Online, 2018. What is Visual SLAM Technology and What is it Used For?. Retrieved: 15.12.2020. https://www.visiononline.org/blog-

article.cfm/What-is-Visual-SLAM-Technology-and-What-is-it-Used-For/99

Viswanathan, D., Features from Accelerated Segment Test (FAST), 2011.

Webster, J. G., Huang, S., Dissanayake, G. 2016, Robot Localization: An Introduction. Wiley Encyclopedia of Electrical and Electronics Engineering, 1-10.

Wang, Z., Shen, Y., Cai, B., Saleem, M. T. 2019, A Brief Review on Loop Closure Detection with 3D Point Cloud, IEEE International Conference on Real-time Computing and Robotics (RCAR), 929-934.

Yu, H., Shengyong, C. 2018, Advances in sensing and processing methods for three-dimensional robot vision, International Journal of Advanced Robotic Systems, 15.

Yilmaz, O., Karakus, F. 2013, Stereo and kinect fusion for continuous 3D reconstruction and visual odometry, 115-118.

Wikipedia, 2020. Random sample consensus. Retrieved: 12.01.2021. https://en.wikipedia.org/wiki/Random_sample_consensus

CURRICULUM VITAE

Name and Surname	: Burak Kaan ÖZBEK
Place and Date of Birth	: ISTANBUL, 10/04/1995
Marital Status	: Single
Foreign Languages	: English
E-mail	: bkaan.ozbek@istanbulticaret.edu.tr

Educational Status

High School	: Hayrullah Kefoglu Anatolian High School, 2013
Bachelor's Degree	: Istanbul Commerce University, Graduate School of Natural and Applied Sciences, Computer Engineering (Full Scholarship), 2017
Master's Degree	: Istanbul Commerce University, Graduate School of Natural and Applied Sciences, Computer Engineering, 2021

Work Experience

Huawei, Software Test Engineer	08.2017-12.2017
Turkish Aerospace, Software Design Engineer	12.2017-ongoing

Publications

Kaan Özbek, B., Turan, M., (2020), Research on the Availability of VINS-Mono and ORB-SLAM3 Algorithms for Aviation, WSEAS Transactions on Computers, ISSN/E-ISSN: 1109-2750, Volume 19, pp. 216-223. doi: https://doi.org/10.37394/23205.2020.19.27