Study of vehicle localization optimization with visual odometry trajectory tracking

Dayang Nur Salmi Dharmiza Awang Salleh

▶ **To cite this version:**

Dayang Nur Salmi Dharmiza Awang Salleh. Study of vehicle localization optimization with visual odometry trajectory tracking. Signal and Image Processing. Université Paris Saclay (COmUE), 2018. English. NNT : 2018SACLS601 . tel-03091994

HAL Id: tel-03091994

https://theses.hal.science/tel-03091994

Submitted on 1 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

université PARIS-SACLAY

NNT : 2018SACLS601

Thèse de doctorat

UNIVERSITÉ PARIS SUD
Comprendre le monde, construire l'avenir

Study of Vehicle Localization Optimization with Visual Odometry Trajectory Tracking

Thèse de doctorat de l'Université Paris-Saclay
préparée à l'Université Paris-Sud

Ecole doctorale n°580 Sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat: Traitement du signal et des images

Thèse présentée et soutenue à Orsay, le 19/12/2018, par

DAYANG NUR SALMI DHARMIZA AWANG SALLEH

Composition du Jury :

Maan El Badaoui El Najjar Professeur, Université de Lille 1	Président
Roland Chapuis Professeur, Université de Clermont Ferrand	Rapporteur
David Betaille Directeur de recherche, IFSTTAR Nantes	Rapporteur
Roger Reynaud Professeur, Université Paris Sud	Examinateur
Lydie Nouvelière Maitre de Conférences, Université d'Evry	Examinatrice
Emmanuel Seignez Maitre de Conférences, Université Paris Sud	Directeur de Thèse

Synthèse

Au sein des systèmes avancés d'aide à la conduite (Advanced Driver Assistance Systems - ADAS) pour les systèmes de transport intelligents (Intelligent Transport Systems - ITS), les systèmes de positionnement, ou de localisation, du véhicule jouent un rôle primordial. Le système GPS (Global Positioning System) largement employé ne peut donner seul un résultat précis à cause de facteurs extérieurs comme un environnement contraint ou l'affaiblissement des signaux. Ces erreurs peuvent être en partie corrigées en fusionnant les données GPS avec des informations supplémentaires provenant d'autres capteurs. La multiplication des systèmes d'aide à la conduite disponibles dans les véhicules nécessite de plus en plus de capteurs installés et augmente le volume de données utilisables. Dans ce cadre, nous nous sommes intéressés à la fusion des données provenant de capteurs bas cout pour améliorer le positionnement du véhicule.

Parmi ces sources d'information, en parallèle au GPS, nous avons considérés les caméras disponibles sur les véhicules dans le but de faire de l'odométrie visuelle (Visual Odometry - VO), couplée à une carte de l'environnement. Nous avons étudié les caractéristiques de cette trajectoire reconstituée dans le but d'améliorer la qualité du positionnement latéral et longitudinal du véhicule sur la route, et de détecter les changements de voies possibles.

Après avoir été fusionnée avec les données GPS, cette trajectoire générée est couplée avec la carte de l'environnement provenant d'Open-StreetMap (OSM). L'erreur de positionnement latérale est réduite en utilisant les informations de distribution de voie fournies par OSM. Dans ce but, nous avons proposé une méthode basée sur les facteurs de distribution (RPDF). Tandis que le positionnement longitudinal est optimisé avec une correspondance de courbes entre la trajectoire provenant de l'odométrie visuelle et les routes segmentées décrites dans OSM par un ensemble de trois nœuds connectés.

Pour vérifier la robustesse du système, la méthode a été validée avec des jeux de données KITTI en considérant des données GPS bruitées par des modèles de bruits usuels notamment les bruits blanc, un biais de mesure et une perte de signal GPS. Plusieurs méthodes d'odométrie visuelle ont été utilisées pour comparer l'influence de la méthode sur le niveau d'amélioration du résultat après fusion des données. En utilisant la technique d'appariement des courbes que nous proposons, la précision du positionnement connait une amélioration significative, en particulier pour l'erreur longitudinale. Les performances de localisation sont comparables à celles des techniques SLAM (Simultaneous Localization And Mapping), corrigeant l'erreur d'orientation initiale provenant de l'odométrie visuelle. Les différents bruits modélisés sur les données GPS n'influent pas sur les performances du résultat de la localisation et la robustesse de la méthode proposée a été éprouvée dans le cas de routes divergentes nécessitant un choix sur la voie suivie par le véhicule.

Nous avons ensuite employé la trajectoire provenant de l'odométrie visuelle dans le cadre de la détection de changement de voie. Cette indication est utile pour les systèmes de navigation des véhicules. La détection de changement de voie a été réalisée par une somme cumulative et une technique d'ajustement de courbe et obtient de très bon taux de réussite. Des perspectives de recherche sur la stratégie de détection sont proposées pour déterminer la voie initiale du véhicule en particulier dans le cas d'un changement de nombres de voie sur la même route. De plus, des statégies d'optimisation des résultas sont proposés quant à la précision des données OSM, notamment sur la position de voie additionnelle ou en détectant du marquage au sol, au besoin.

En conclusion, les résultats obtenus lors de ces travaux montrent l'intérêt de l'utilisation de la trajectoire provenant de l'odométrie visuelle comme source d'information pour la fusion de données à faible coût pour la localisation des véhicules. Cette source d'information provenant de la caméra est complémentaire aux données d'images traitées qui pourront par ailleurs être utilisées pour les différentes taches visée par les systèmes d'aides à la conduite.

Declaration of Authorship

I, Dayang Nur Salmi Dharmiza AWANG SALLEH, declare that this thesis titled, "Study of Vehicle Localization Optimization with Visual Odometry Trajectory Tracking" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- No part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor, Emmanuel Seignez, for giving me an endless support and motivation in many aspects of life, from the first day I set my foot in France. I am truly grateful for having such a great supervisor that never stops giving me encouragement during my study.

I would also like to thank my thesis reviewers, Prof. David Betaille and Prof. Ronald Chapuis, for your comments and fruitful suggestions on my work. Not to forget, my colleagues – Rokiatou, Jennifer, Hernan, Sara, Dai Duon, and many others – that always brighten up my days at lab and become my source of information whenever I have questions or doubts.

I am also debtful to my family especially my parents, who raised me up well, help me making decisions, and very supportive in my study. My siblings also deserve to be mentioned here, for their prayers for my success. Last but not least, I would like to endlessly thank my husband, who had sacrificed a lot on our journey to finish my study away fr. Words can never really describe how thankful I am to have you by my side and only God can return your kindness.

Contents

List of Figures

List of Tables

Dedicated to:

Babah,

Mayya,
and our new bundle of joy . . .

Chapter 1

Introduction on Vehicle Localization

"When Henry Ford made cheap, reliable cars people said, "Nah, what's wrong with a horse?" That was a huge bet he made, and it worked."

— Elon Musk

As quoted from the co-founder, CEO and product architect of Tesla Incorporation, it proves that with innovation and technology, the quality of life is improved, although it might seem dispensable in the beginning. Indeed, humans – whether consciously or not – have always desired for more comfortable life by putting great effort on technology development. That is the reason why technology development has never stopped emerging and researchers are working with each other in their own respective fields.

Vehicle technology has been developing immensely as it is one of the important necessities in daily life. It is the second most expensive commodity purchased by an adult after residential property. People commute either by public transport or personal vehicle from one point to another and safety is among the main concerns. Besides, with urban development and advanced road network, using the most efficient route in terms of journey time and fuel consumption is of the huge factor to be taken into consideration in our hectic life. Thus, the need for high safety vehicles with accurate positioning technology and uninterrupted routing service are constantly in high demand.

1.1 Background

Recent decade has shown massive development in intelligent vehicles and Advanced Driver Assistance Systems (ADAS) with the birth of autonomous vehicles. These vehicles are capable of 'self-driving' without human input by sensing its environment from sensors. This includes Tesla cars, Google self-driving car (Waymo) and cruise-control systems introduced by high-end car manufacturers like Mercedes and Audi. These vehicles are equipped with sensors for various techniques of sensing their surroundings, such as odometry, GPS, radar, laser sensor, and computer vision.

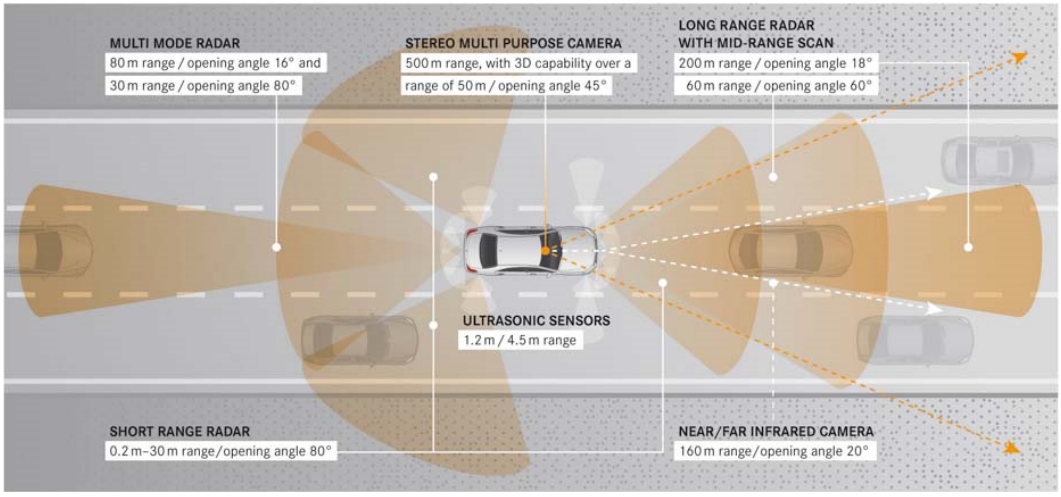


FIGURE 1.1: *Sensors equipped on a Tesla model S*

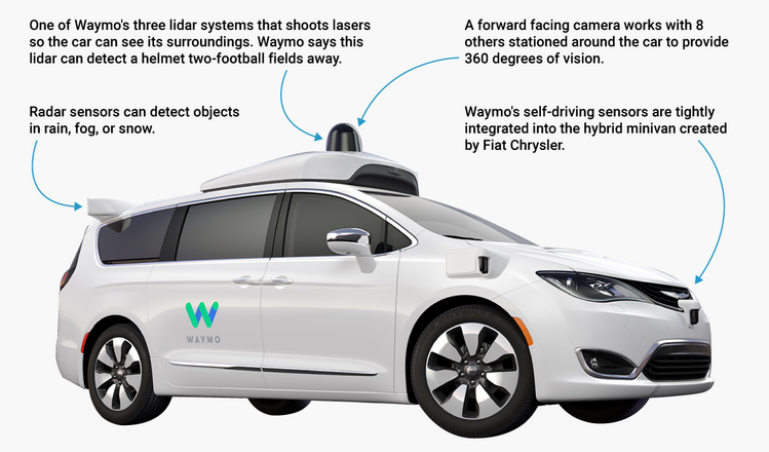


FIGURE 1.2: *Sensors equipped on a Waymo autonomous car*

The sensed surroundings are then interpreted into information for vehicle's safety assessment in terms of collision probability, speed limit, and allows the vehicle to localize its

position accurately for navigation and routing. To enable the functionality of all these assessments, it relies on the key component in an intelligent vehicle which is its navigation system. Without a correct navigation, it might affect the vehicle routing choice for the driver to arrive at certain destination. Accurate localization is also a requirement for ADAS and Location Based Services (LBS) that recently emerges to provide information or entertainment for users. Localization plays a crucial role in determining the exact position of the vehicle in the global coordinate to liaise with the navigating system and obtain accurate information for LBS.

1.1.1 GPS

Vehicle navigation system is part of the automobile controls (can also be a third-party device) used to find direction from a location to certain destination. It typically uses a satellite navigation device to obtain its position data which is then correlated to a position on a road. Then, routing can be calculated when the driver needs to know the advised direction of taken route. These days, we are able to obtain our position from the widely available Global Positioning System (GPS) devices, provided in a handheld device or equipped in the vehicle. Initially, the GPS was designed only for military purposes until in the 1980's, it was made available for public use.

The GPS communicates with satellites to estimate our coordinates on earth. It consists of 24 satellites, that circle the earth twice a day, to provide one's position, at a specific time, with velocity information. Through GPS, we are able to locate positions on the earth based on the distance measurement from the satellites. This allows us to record or find locations coordinates on the earth and with the use of a map, we can navigate from a point to another point. Therefore, GPS devices are also typically used for vehicle navigation purposes with routing services. In order to determine one's two-dimensional position (latitude and longitude), the GPS receiver must be locked with the signal of at least three satellites. With four or more satellites in view, the receiver can calculate position in three-dimensional (latitude, longitude and altitude). GPS satellites transmit at least two low-power radio signals that travel by line of sight, which means they can pass through clouds, glass and plastic. However, the signals will not go through most solid objects, such as buildings and mountains.

However, relying on GPS alone may result in localization failure due to errors in GPS signal. The factors that reduce GPS signal accuracy include:

- *Ionosphere and troposphere delays*: The ionosphere is the atmosphere layer from 50 to 500 km altitude that consists mostly of ionized air, while the troposphere is the lower layer of the atmosphere (below 13 km) that faces variations in temperature, pressure, and humidity associated according with weather conditions. Satellite signals tend to be refracted as they pass through the earth's atmosphere – causing the signals to slow down or speed up. Since the satellite signals contain correction information for

ionospheric interference, the GPS system is able to partially correct the error, and this leaves up to a 10m of horizontal error on the ground.

- *Signal multipath interference*: This is caused by the reflected signals from surfaces near the GPS receiver that can either interfere with or be mistaken for the true uninterrupted signal from a satellite.
- *Receiver clock errors*: The receiver's built-in clock may have slight timing error because it is less accurate than the atomic clocks on GPS satellites.
- *Orbital errors*: Satellite orbit ('satellite ephemeris') pertains to the altitude, position and speed of the satellite and it varies due to gravitational attraction and solar pressure fluctuations.
- *Number of visible satellites*: GPS accuracy relies on the number of satellites that are in view with the receiver (the more, the better). When a signal is obstructed, users will get position errors or possibly no position reading at all.
- *Satellite geometry*: Satellite signals are more effective when satellites are located at wide angles in relative to each other, rather than in a line or tight grouping.
- *Selective availability (S/A)*: It is the intentional degradation of the satellite signals by a time varying bias, applied by the U.S. Department of Defense before, which makes signals less accurate for non - U.S. military and government users for security reasons. The government reduced the amount of signal interference to 0m in May 2000, which improved the accuracy of civilian GPS receivers. However, they retain the ability to reactivate S/A without notice to the GPS users.

To overcome these problems, GPS error compensation techniques have been proposed. For instance, (Bétaille et al., 2015) presented positioning improvement by 3D geometrical city modelling. From the modelling, it is possible to identify line-of-sight or non-line-of-sight satellite signals, hence allowing correction of pseudo-range measurements. Then, the urban trenches were created automatically from a rover trajectory as a reference to match with the nearest road arc segment. It is an interesting approach of integrating trajectory, map, and GPS data correction. However, the modelling is based on the BD Topo map database for France, thus it is not practical for global use.

Besides, another option is to have a backup system by fusing GPS data with other additional information from sensors or map to obtain one's position correctly. Therefore, it is recommended to explore the available sensors typically used in vehicle that can provide additional information for localization to design a strategy in improving localization accuracy.

1.1.2 Sensors

Sensors are an important component of automotive electronic control systems, defined as 'devices that transform (or transduce) physical quantities such as pressure or acceleration (called measurands) into output signals (usually electrical) that serve as inputs for control systems' (Norton, 1989). In other words, it is a device that measures certain quantity and converts it into a signal that can be read by an observer, an instrument or a computer. Sensors apply physical properties of matter and laws of physics to perform and they are capable of perceiving properties or environmental attributes. These attributes include light, heat, pressure, sound, electromagnetism or motion. Generally, sensors can be classified into two categories: passive and active. A passive sensor only measures the energy or information obtained from the objects in the environment for observation. Meanwhile, an active sensor emits energy and then observes any change in the reflected energy to estimate the state of the environment.

Today, most vehicles already incorporated many types of sensors such as the Mass Air Flow Sensor (MAF), engine speed sensor, fuel temperature sensor, and voltage sensor. These sensors are mainly built into vehicle's engine to ensure that the driver can identify and prevent possible issues before they result in breakdowns that consumes expensive repairs. Besides, these engine sensors are also to ensure that the vehicle is operating efficiently. Aside from engine sensors, vehicles are more and more equipped with either built-in or autonomous sensor for environment perception purposes. This enables the system to obtain information from the surrounding environment, in terms of distance with other objects/vehicles, pedestrian or road signs detection, map building, and localization.

A. Wheel Odometry (ABS)

It is the most widely used method to estimate the position of mobile robots from the number of rotations of the wheels. Braking in vehicles today is commonly assisted with Anti-Lock Braking System (ABS) that uses angular encoders attached to the wheels. With this system, wheel rotation is translated into linear displacement relative to the ground while the encoder as shown in Figure 1.3 is utilized for rotation measurement. Since wheel odometry is a relative positioning technique, it faces positioning accuracy problems due to the drift caused by wheel slippage. This error is accumulated over time for both translational and orientational errors that worsens the positioning performance without proper correction.

B. INS

Inertial navigation system (INS) as shown in Figure 1.4 is a navigation tool that utilizes motion sensors (accelerometers), rotation sensors (gyroscopes) and embedded systems to calculate the position, orientation and velocity of a moving object continuously. It has the

FIGURE 1.3: Wheel odometry encoder (*Pololu Corporation 2016*)

advantage of being self-contained, without requiring external references. Unfortunately, INS is also prone to drift accumulation because the velocity and position change calculation is performed by mathematical integrations of acceleration. While accelerometer data has to be integrated twice to obtain the position, rate-gyro data are only integrated once for orientation tracking. Thus, resulting small errors in the acceleration measurement and angular velocity will contribute in velocity data errors. This are then compounded into larger errors in position results (Wang et al., 2014; Woodman, 2007). These errors are also increased with time and the position needs to be corrected periodically.

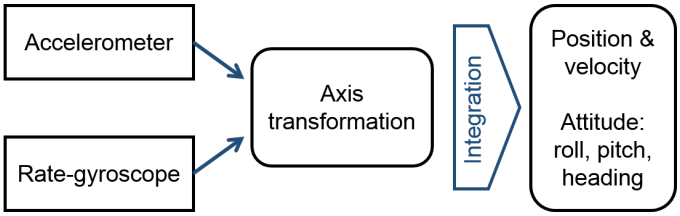


FIGURE 1.4: Block diagram of inertial navigation system (Gade, 2005)

C. Radar

In the beginning of the 20th century, an early implementation of radar technology (an acronym for Radio Detection And Ranging) was meant to detect the presence of ships in fog. Nowadays, radar is still highly valued because it can provide distance and velocity information, even in challenging environmental conditions. Radar can function even in fog, rain, wind, darkness, or blinding sun. As an active sensor, radar uses a transmitter that emits radio waves and a receiver that collects the reflected waves. Radar uses the Doppler Effect to accurately measure the object's velocity by detecting changes in the reflected wavelengths. Object detection quality highly relies on the wave reflection strength which is influenced by factors such as the distance, the object's size, its absorption characteristics and the reflection angle.

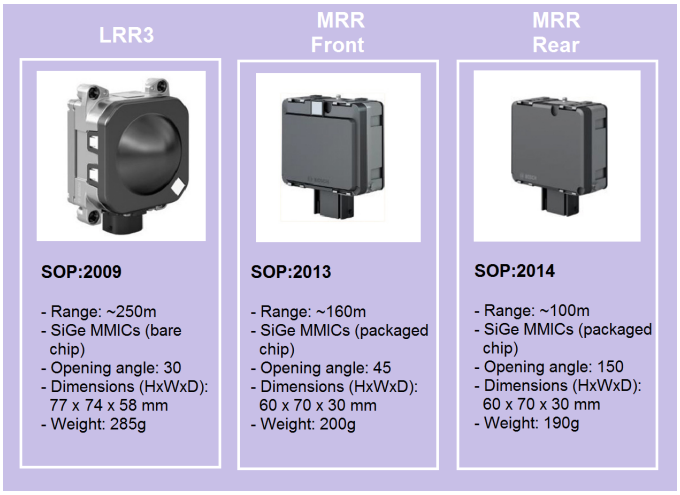


FIGURE 1.5: Examples of medium- and long-range automotive radar sensors operating up to 250 meters. (*Bosch radar*)

In automotive application of radars, challenging scenarios result in problems especially when two objects are located too close next to each other on the road. This is because it would provide a very similar reflections making it difficult to distinguish. Besides, large objects that are close to the transmitter can saturate the receiver and a radar usually cannot resolve the type of the object. Signals received from radar are not intelligent enough to resolve small details of objects which would be useful in object identification. Therefore, without this qualitative information, the application for this sensor is limited to detected object's distance and speed measurement.

There are also characteristic trade-off in radar range versus field of view as indicated in Figure 1.5. In order to identify objects at further distance (for higher-speed traffic), long-range radar systems that are capable of measuring distances exceeding 250m would have a narrow beam width of only about 15°. As a result, although it allows measurements of objects directly in front of the car up to a long distance, it will not be able to detect vehicles or other objects nearby outside the beam angle. There are mechanical and multi-beam solutions that may increase the systems' field of view, but they require larger sensors, and more complex in computation and systems.

D. Camera

While radar is good at detecting objects physically, the systems are not able to identify objects and visible messages like road signs, road markings, and traffic light colours. Video images captured from cameras on the other hand support the driver with a better representation of the environment outside the vehicle. Today, two-dimensional cameras are widely available to display images while luxury class car makers are beginning to install

cameras with virtual, three-dimensional image displays. Besides, many drivers have installed plugged in camera that records video as a proof for safety purposes. Camera also helps to overcome blind-spot problem typically faced by human drivers that contributes to the major traffic accident factors.

Aside from security measures, with computer vision, cameras are also able to automatically detect objects, classify them, and determine the distance from them, provided that the calibration parameters are carefully calculated. For example, from cameras, we are able identify pedestrians and cyclists, motor vehicles, side strips, bridge abutments, and road margins. The algorithms are also used to detect traffic signs and signals as an additional information of the road rules. These information are useful to provide knowledge on the surrounding environment to be integrated in the driver safety system.

With the technology advancement, cameras today can also be utilized for vehicle trajectory generation from visual odometry. This is achieved by the feature detection by image processing, and with vehicle movement, the same features can be matched in the preceding frames. The matched and tracked features contain the information of vehicle ego-motion, which can be translated into trajectory. This allows for enhanced navigational accuracy in vehicles on any surface for localization purposes.

Cameras utilized in vehicles can be divided into monocular and stereo vision. Monocular vision only requires a single camera for image processing while stereo vision uses two cameras, placed horizontally, in the system. With stereo camera, scene depth information can be estimated from the calculation of the object's distance from multiple captures of the same scene from different viewpoints. Although it consumes slightly higher implementation cost compared to monocular vision, the stereo vision is highly advantageous to extract information about the relative position of 3D objects in the vicinity of the system.

Since a full high-definition (HD) image equals to millions of pixels at each frame, use of cameras in ADAS produces massive amounts of data for image processing. This makes processing a computationally intense and algorithmically complex job. Despite of this, cameras are by far the cheapest and the most available sensor in automotive application. Besides, unlike radar, cameras can see colour, making them the best candidate for scene interpretation in intelligent vehicles.

E. Lidar

Lidar, an acronym for Light Detection and Ranging, is a remote sensing method that uses light in the form of a pulsed laser to measure ranges. It is originally developed over 40 years ago to track lunar and satellite distances. However, as the lidar systems become more compact, many applications have been unveiled. Lidar was first utilized on cars in the 2005 Grand DARPA challenge that makes the innovator, Stanley, won the challenge by using 5 lidar sensors mounted on the car roof.

A lidar system consists of four key components: a) a transmitter to emit laser pulses, b) a receiver to intercept pulse echoes, c) an optical analysing system to process input data, and d) a powerful computer to visualize a live, 3D image of the system's surroundings. Lidar systems emit light waves that travel outwards in all directions until making contact with an object, resulting in a reflected light wave that is redirected back to the source. The distance of that object is then calculated based on the time it took for the echo to return, in relation to the known speed of light.

These years, lidar is widely used in self-driving car, owing to its capability to generate huge 3D maps to allow the car or robot navigate within it. By using a lidar to map and navigate an environment, it allows the system to know ahead of time the bounds of a lane, or if there is a stop sign or traffic light a few hundreds of metres ahead. This kind of predictability is what a technology like self-driving cars requires, and has been a big reason for the progress over the last five years.

However, this great performance does not come without a high cost. In order to achieve a 360° horizontal field-of-view depth map around a car, current LIDAR systems such as that from Velodyne utilize a mechanical, rotating system that rotates the laser in all directions which is neither aesthetically pleasing nor aerodynamic. These 360° lidar systems are also quite expensive, with prototype systems today costing about $70,000. Besides, lidar faces a challenge with its vertical resolution where the Velodyne's lidar for instance, only has a vertical resolution of 64 pixels. To achieve adequate vertical pixel density, the field-of-view is limited to 26°, resulting in detection failure of objects directly in front of a vehicle like a curb or pothole.

1.1.3 Harvesting Information from Digital Maps

Aside from the perceptive properties obtained from the sensors equipped to a vehicle, we also have abundant information of the surrounding environment provided by digital maps. A digital map may contain information such as road connectivity, traffic signs, speed limit, public transport stations, and geographical structure. Data provided by the digital map is growing rapidly with the crowdsourcing and is expected to improve the map's information accuracy. This allows the possibility to harvest the information provided by the digital map as an additional input for vehicle localization by data fusion. Inputs obtained from sensors or GPS data could be a complimentary information for map matching which will further increase localization accuracy effectively.

Nowadays, the whole world map is digitalized and accessible from map service providers on the web. Among the widely known digital mapping services utilized in navigation system development are the Google Maps, Bing Maps, Apple Maps and OpenStreetMap. Here, each service will be described and compared in terms of the service availability and information accuracy.

A. Bing Maps

Bing Maps is a web mapping service provided as part of the Microsoft's Bing suite of search engines which was initially known as Live Search Maps. With Bing Maps, users can browse and search topographically shaded street maps for many cities worldwide. It is also possible to browse public user-created points of interest and searches cover public collections, businesses or types of business, locations, and people. Five street map views are available: Road View, Aerial View, Bird's Eye View, Street Side View, and 3D View.

Road view is the default map view that displays vector imagery of roads, buildings, and geography. The data from which the default rendered road map is licensed from Navteq. In some places, road view maps from alternative data providers are also available. For instance, in the UK, road data from the Ordonnance Survey can also be displayed. There are also 1st and 3rd party applications that can add additional functionality and content to Bing Map such as parking finder, taxi fare calculator and Facebook friends map. Bing Maps updates are being released on a monthly basis and each imagery release may contain more than 10TB of imagery data.

B. Google Maps

As recognized from its name, Google develops the Google Maps service since 2005, that currently offers satellite imagery, street maps, 360° panoramic street view, real time traffic updates and route planning. The service's front end utilizes JavaScript, XML and Ajax. In 2008, Google Map Maker was launched to allow users to contribute in mapping service update, but it was discontinued from March 2017. However, any errors such as inaccurate geographical locations or location names can be reported by users by suggesting corrections through the 'send feedback' button which will then be reviewed before accepted or declined.

Google Maps might have a lot of useful information and services, but it is not open at the level of raw map data, in order to maintain its commercial advantage. They only expose their downstream products and services from the map data. However, with the growing tendency among developer communities to explore on the innovation possibilities from the raw map data, it is not feasible on the Google's restricted platform.

C. Apple Maps

Apple Inc. launched Apple Maps in September 2012 in iOS to replace Google Maps as the default mapping service for Apple operating systems. This is due to the growing tensions between Google and Apple in late 2009 when Android version of Google Maps featured turn-by-turn navigation, which the iOS version lacked. Thus, Apple announced the initial release of Apple Maps that include turn-by-turn navigation, 3D maps, Flyovers (3D photo-realistic views of big cities and landmarks), and virtual assistance Siri. At first, Google

did not immediately launch a mapping application on its own on mobile. Not until three months after Apple Maps announcement, Google Maps was released in the App Store with new features. The largest selling point for Apple Maps' turn-by-turn navigation is the 3D maps (activated by swiping up with two fingers during a journey). While it is currently available in only certain locations around the world, Apple is constantly updating its 3D maps and adding new locations. Users can see surrounding building in 3D, which makes it easier to figure out exactly where they are in relation to the position displayed on the screen.

However, Apple Maps has also been criticized for numerous errors and poor map detalization since its launch. Since then, some errors have been corrected, but the problem of detalization is still relevant for the USA, Western Europe, and some Asian countries. For example, in Moscow, Russia, according to Apple Maps, there are no buildings, and some cities look like cross-roads in the fields. Another drawback of Apple maps is that it is not accessible for use in the offline mode. The system can display the cached data without internet connection, but it displays only those regions that a user has viewed before and will not work at a new place.

D. OpenStreetMaps

OpenStreetMap (OSM) is an open project that was aimed at providing a free detailed map of the world. The project is constantly updated and improved by users around the world where any registered user can edit and make additions to the map. This ensures high accuracy and detalization of cartographic data. OSM was initiated by Steve Coast in 2004, inspired by the success of Wikipedia and the predominance of proprietary map data. Since then, it has grown to over 4 million registered users as of November 2017, who can collect data using manual survey, GPS devices, aerial photography, and other free sources for map update. This crowdsourced data is made available under the Open Database Licence that allows everyone to access, download, store, and use the data freely without restrictions. Aside from the free access, here are several advantages of OSM in comparison with other commercial map service providers:

- Users can style their map to look how they want, while with other map providers, the map style is fixed by the provider.
- With OSM, users are in control of viewing things as wished. They can choose to emphasize cycle routes (where most of other maps do not even have cycle routes) and play down motorways or only want to label subway stops but ignore bus stops by filtering the raw map data.
- Most commercial providers just do streets but OSM do much more which includes natural features, bus routes, footpaths and cycleways, administrative boundaries, shops, rivers and canals.

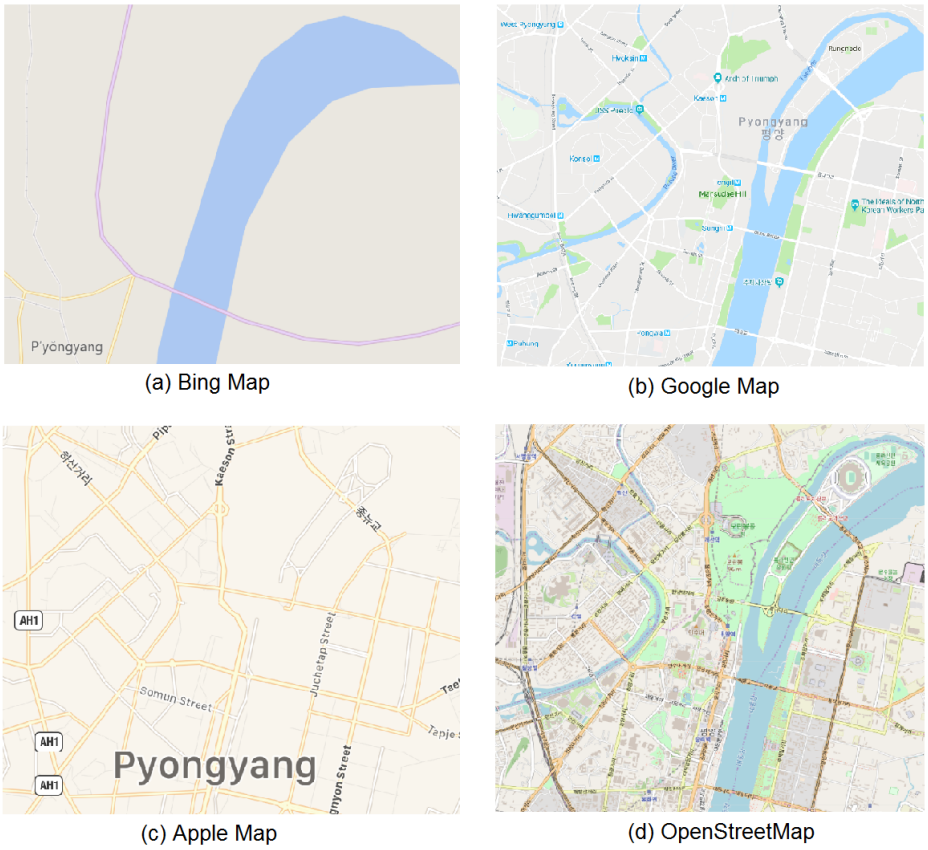
- Maps from commercial providers are updated every month based on the region, or less. New roads and buildings can be missing long after they are built without frequent update. OSM data is constantly updated, any minute or users can opt to update the map themselves.
- There is no limit to what users can do with OSM as the data keeps on growing every second and raw data is always easily available for use.
- Users can download all or some of the map for offline use, meaning we can safely use OSM information to find our way around without having to reveal our location to anybody.

In fact, more major organizations have been choosing OSM for their maps. For instance, Foursquare switched to the OSM powered Mapbox platform in February 2012 and since March 2013, Wikipedia has also started using OSM as well. Craigslist currently uses OSM for apartment searches and even Apple has used OSM data in its maps. There are also other popular platforms using OSM powered maps such as Github, Pinterest, Roadtrippers and Strava. The main reasons for this progress are OSM's style flexibility and the fact that OSM is always free to users, developers and companies.

(a) Bing Map (b) Google Map

(c) Apple Map (d) OpenStreetMap

FIGURE 1.6: Cartography differences at Digiteo building, Universite Paris-Saclay



(a) Bing Map (b) Google Map

(c) Apple Map (d) OpenStreetMap

FIGURE 1.7: Cartography differences at Pyongyang, North Korea

1.1.4 Digital Maps Comparison

In order to illustrate map accuracy and cartography data, snapshots of similar area are taken as shown in Figure 1.6 and Figure 1.7 as of 9 March 2018. The first set of images are taken to compare map updates since the area has undergone development over the past two years. Meanwhile map of Pyongyang city of North Korea is also used to observe the map detalazation in a secretive state. Apparently, Google Map manages to obtain detailed map of the city which was released in 2013 – thanks to the citizen cartographers that contributed through Google Map Maker although OSM has more information.

It is obvious that public contribution on map data is the most effective way to acquire an up-to-date information as portrayed by maps from OpenStreetMap. This information is very useful in LBS and also aiding vehicle localization if utilized correctly. Besides, the availability of information provided by the digital maps to public allows rapid development in map-based applications.

1.2 Contributions

Certainly, an autonomous car might have solved the main problems in localization and risk assessments, but it owes to the equipped sensors performance with higher computational and financial cost. The objective in our research has been to develop an accurate low-cost localization in urban environment for common vehicle. Without requiring built-in sensors installation on vehicle, we aim to develop a standalone system that is autonomous and can be widely used in ADAS. The main problems in GPS data are studied and information provided by available sensors are carefully considered and selected to achieve our objectives. There are three hypothesis that we can state in this work:

- GPS is the most widely used system as a base for vehicle positioning due to its low-cost and availability,
- digital map is rich with useful information for vehicle navigation which is expected to continuously improve, and
- the use of cameras on a car has become more common which can be utilized for visual odometry trajectory, aside from safety system and road features / symbolic / markings detection purposes.

In this study, we try to explore how the data fusion of low-cost GPS, digital map and visual odometry trajectory can contribute in vehicle localization improvement. As a complementary information to overcome GPS flaws, vehicle ego-motion from visual odometry trajectory is used for pose prediction. Then, the trajectory is also matched with road information from the OSM. Visual odometry has been widely studied by researchers that either improves its performance in monocular system (Scaramuzza et al., 2009, Forster et al., 2014, Engel et al., 2013), or stereo system (Kitt et al., 2010, Geiger et al., 2011, Wang et al., 2017).

Meanwhile, there are also works that proposed application in Simultaneous Localization and Mapping (SLAM) (Cvišić et al., 2017; Esparza-Jiménez et al., 2016; Moreno et al., 2016; Mur-Artal et al., 2015) which is particularly useful in robotic mapping of an unknown region with drift correction from loop-closure detection. However, in vehicle navigation, we can simplify the mapping problem since the map data can be obtained beforehand from the available resources. Therefore, our method only requires the visual odometry trajectory as an additional information in data fusion with GPS and map.

In the end, this thesis has contributed in:

- providing the proof of accountability in utilizing visual odometry trajectory in data fusion for positioning error optimization in both latitudinal and longitudinal,
- the probability of utilizing visual odometry trajectory curve in determining lane-changing events

- a practical low-cost solution in accurate localization with high robustness against different types of GPS error

1.3 Thesis Outline

This thesis consists of six chapters. The first chapter gives general introduction on the research background, motivation, and contribution of the thesis.

In Chapter 2, literature review of related works on visual odometry are presented, to provide deeper understanding of limitations faced and the research gap that we would like to address in our work. Here we also provide a review in multisensor data fusion, with benchmark scoreboard analysis.

Then in Chapter 3, we present the first approach in our method by further explaining about the fusion technique used on VO trajectory, noisy GPS and OSM data. We implemented the particle filter method and tested on a simple drive to study the localization performance. Then, a Road Probability Distribution Factor (RPDF) method is proposed and explained in this chapter to further improve the localization accuracy.

Meanwhile in the following Chapter 4, we continue to develop our fusion technique by adding another fusion level for trajectory curve comparison in order to optimize positioning accuracy. This is called the Segmented Curve Matching (SCM) since the visual odometry trajectory trail is compared with segmented roads from OSM. System robustness against severe GPS noise is also evaluated in this chapter. For performance comparison, we tested different VO approaches as the trajectory input to study the degree of improvement in accuracy.

Next, in Chapter 5, a study on lane-changing/lane-keeping detection by VO trajectory curve analysis is presented with two different approaches. We would like to show that it is possible to utilize VO trajectory in lane-changing detection, provided scale ambiguity and heading error are not too severe.

Finally, Chapter 6 will conclude this thesis and provides future works in this research. As an academician, this research does not stop upon submission of this thesis, but I will seek further knowledge and looking forward to collaborate with other researchers for future research development.

Chapter 2

Related Works on Localization with Visual Odometry

Accurate vehicle localization is a fundamental challenge in intelligent vehicle applications. It is highly desirable to have the information of one's position accurately to achieve autonomous navigation and integration with location-based services. Thus, various sensors are equipped to vehicles as mentioned in previous chapter and researchers have developed approaches for vehicle positioning from information obtained from wheel odometry, GPS, INS, radar, laser and visual odometry (VO). Each technique has its own advantages and weaknesses in terms of reliability, precision, cost, and feasibility.

In our work, we propose to focus on the employability of VO trajectory curve in our low-cost vehicle localization system. This chapter will present the related works conducted in recent years to assess merits of each technique.

2.1 Visual Odometry

The term "odometry" actually originated from the combination of two Greek words *hodos* that means "journey" and *metron* which means "measure" (Fernandez et al., 2004). Thus, measuring one's journey by estimating the change in an object's pose over time is defined as odometry. Visual odometry (VO) is a positioning technique by using a stream of images acquired from visual sensor (camera). These images contain environment information presented in colour, texture and shape that are useful for feature tracking to estimate camera movement.

The idea of vehicle positioning estimation based on visual input was firstly introduced and suggested in the early 1980s by (Moravec, 1980). Since then until the new millennia, NASA dominated VO research in preparation for Mars Mission in 2004. The term 'visual odometry' coined by (Nistér et al., 2004) was chosen due to its similarity with wheel odometry. However, wheel odometry incrementally estimates one's position or motion by integrating the number of turns of wheels over time, while VO integrates pixel displacements between image frames over time for state estimation.

VO is clearly an inexpensive option in odometry technique that yields higher accuracy than conventional techniques such as INS and wheel odometry. VO has a relative position error ranging from 0.1% to 2% (Scaramuzza et al., 2011). It can be characterized with balanced trade-off among cost, reliability and complexity of its implementation (Nistér et al., 2004). The utilization of consumer-level camera is a straightforward and low-cost solution compared with the expensive sensors or systems in laser-based localization (González et al., 2012; Nourani-Vatani et al., 2009). Although GPS are widely used and available for outdoor localization, from time to time, it still suffers from lost GPS information or signal noise that cause positioning errors. Hence, with camera as the sensor, VO can work effectively in the GPS-denied environment (Scaramuzza et al., 2011). Besides, the local drift rate under VO is much smaller that drift of wheel encoders and low precision INS (Howard, 2008). To achieve maximum accuracy, VO can be integrated with other existing system such as GPS or INS as an additional information input.

2.1.1 VO Approaches

Position estimation by VO can be mainly categorized in three approaches:

- feature-based,
- appearance-based, and
- hybrid of feature/appearance-based

A. Feature-based approach

Self-explained by its name, this method involves extracting image features such as lines, corners and curves in each image frame, that are then matched and tracked in between sequential images. This approach was previously adapted by (Benseddik et al., 2014; Cumani, 2011; Howard, 2008; Jiang et al., 2014; Naroditsky et al., 2012; Nistér et al., 2006; Parra et al., 2010). Feature matching and tracking are performed only on the distinctive features to estimate the vehicle motion. In this approach, image matching with previous frame is conducted by feature points comparison in both image frames and Euclidean distance of feature vectors are calculated in order to find candidate matching features. Then, camera (vehicle)

displacement is estimated from the velocity vector calculation between the pairs of matched feature points (Lowe, 2004; Nistér, 2003; Nistér et al., 2006).

For stereo vision system, the extracted features from the first image are matched with the corresponding points in the next frame, that will provide the 3D position of the points in space. With this, camera motion can be estimated from the feature points displacement where the camera relative pose is calculated by finding the geometric transformation between two images using a set of corresponding feature points. Nearest neighbour pairs among feature descriptors are determined to compute the matching between the feature points of two images. To achieve this, Longuet-Higgins in 1987 proposed an 8-point algorithm to compute camera pose via essential matrix (Longuet-Higgins, 1987). Other researchers had also done studies to improve the robustness of the 8-point algorithm method (Hartley, 1997; Wu et al., 2005) and efficiently solve it in a closed-form algorithm with minimal set of 5-points as proposed by (Nistér, 2003). Nister suggested that the relative camera pose can be obtained from five matching feature points while (Stewénius et al., 2006) used 6,7 and 8 feature pairs for motion estimation. This feature-based VO approach has been succesfully implemented in the navigation system of Mars exploration rovers as reported in (Maimone et al., 2007).

There are several feature detection methods usually used in this approach such as FAST, SIFT and SURF. FAST is an acronym for 'Feature from Accelerated Segment Test', which is a quick corner detection (Rosten et al., 2006). It uses a circle constructed from 16 pixels (with 3 pixels radius) around a centre pixel p to determine if p is a corner. The detection is performed by comparing pixel intensity of adjacent pixels in the ring with intensity of p as shown in Figure 2.1 below.

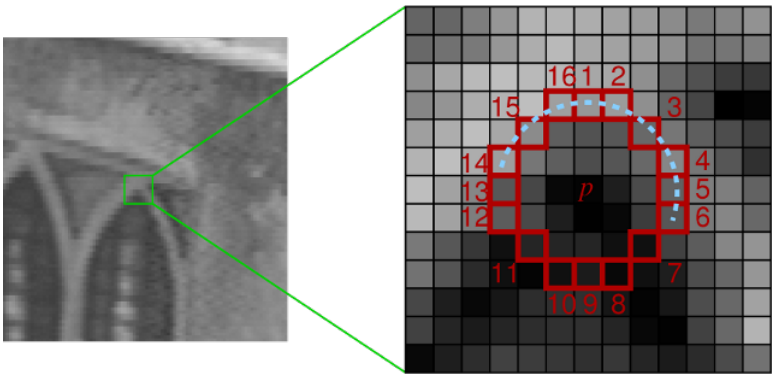


FIGURE 2.1: FAST corner detector: Pixels 11-16 and 1-6 are brighter than p, hence p is detected as a feature point.

On the other hand, SIFT (stands for Scale Invariant Feature Transform) detector is based on Difference-of-Gaussians (DoG) where it detects the centres of blob-like structures (Lowe, 1999). Even though SIFT is very efficient in object recognition applications, it requires a high computational cost which is a disadvantage for real-time applications (Lowe, 2004; Miksik et

al., 2012). Then, SURF (Speeded Up Robust Features) was proposed to improve computation efficiency of SIFT (Bay et al., 2008). SURF detect features from a blob detector based on the Hessian matrix approximation to find points of interest. However, studies conducted on feature detectors comparison show that FAST has the best overall performance above SIFT and SURF in terms of accuracy, robustness and speed (Ali et al., 2016; Guerrero, 2011; Işık, 2014).

B. Appearance-based approach

The appearance-based approach is a more recent approach, also known as direct-based approach, proposed in several VO techniques (González et al., 2012; Lovegrove et al., 2011; McManus et al., 2013; Nourani-Vatani et al., 2009; Yu et al., 2011). Appearance-based approach generally monitors the changes in image appearance and its intensity of pixel information instead of extracting and tracking image features. While focusing on the information extracted from pixel intensity, the camera motion is estimated by using optical flow. Optical flow algorithm applies the intensity values of neighbouring pixels to find the displacement of brightness patterns from an image frame to another (Barron et al., 1992; Campbell et al., 2004).

There are two types of optical flow algorithms. The first one is the dense optical flow (DOF) algorithm, introduced by (Horn et al., 1981) that calculates all image pixels' displacement by using global constraints. Another algorithm is called sparse optical flow (SOF) that only calculate the displacement of a selected number of pixels in the image, such as the Lucas-Kanade method (Lucas et al., 1981). It was found that SOF is more desirable over DOF for many VO applications due to the fact that dense algorithms are less robust to noise compared to SOF (Campbell et al., 2004; Corke et al., 2004; Nourani-Vatani et al., 2009). In SOF, the features are selected carefully based on higher variance between neighbour pixels, hence resulting in more reliable motion estimation.

One of the common method used in this approach is the template matching method. The template matching method attempts to match a selected patch or template of the current image frame with the next frame. From the template matching, vehicle displacement and rotation angle can be retrieved. Thus, template matching is an important task in this approach. It is also useful in other computer vision applications such as object detection, video compression and automatic inspection. It is the process of determining the existence and position of an object (template) inside a large scene image (search area) (Choi et al., 2002; Goshtasby et al., 1984). Template matching computes the similarity between the template and search area by moving the template over the search area and calculate the similarity degree in each location based on various measures. The shift position with highest degree of similarity is the most likely position of the template found in the search area. The template matching can be obtained through the normalized cross correlation (Aqel et al., 2016).

From the identified template location, pixel displacement between the template and the maximum correlation point is calculated to obtain the horizontal and vertical pixel displacements (Δu and Δv). In order to achieve measurements in metres, the pixel displacements are converted from the intrinsic and extrinsic camera calibration parameters through the following equations:

$$\Delta X_c = -\Delta u \left(\frac{Z_c}{f_x} \right), \tag{2.1}$$

$$\Delta Y_c = -\Delta v \left(\frac{Z_c}{f_y} \right), \tag{2.2}$$

where f_x and f_y both represent the camera focal length which typically have the same value. As for the camera coordinate plane (X_c, Y_c, Z_c), it will be converted to the vehicle coordinate plane (X_v, Y_v, Z_v) based on rotation matrix R_c obtained from when the camera rotation is 180° around Z-axis and then 180° around Y-axis. This computation is depicted in equation below:

$$R_c = R_z \times R_y = \begin{bmatrix} \cos(\theta_z) & -\sin(\theta_z) & 0 \\ \sin(\theta_z) & \cos(\theta_z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos(\theta_y) & 0 & \sin(\theta_y) \\ 0 & 1 & 0 \\ -\sin(\theta_y) & 0 & \cos(\theta_y) \end{bmatrix}. \tag{2.3}$$

Therefore, the vehicle coordinate plane conversion at frame i can be calculated as:

$$\begin{bmatrix} \Delta X_{v_i} \\ \Delta Y_{v_i} \\ \Delta Z_{v_i} \end{bmatrix} = R_c \times \begin{bmatrix} \Delta X_{c_i} \\ \Delta Y_{c_i} \\ \Delta Z_{c_i} \end{bmatrix}. \tag{2.4}$$

Assuming the motion model is based on Ackermaan-steered model, translation ΔX_i is considered from ΔX_{v_i} and rotation $\Delta \theta_i$ is obtained from:

$$\Delta \theta_i = \tan^{-1} \left(\frac{\Delta Y_{v_i}}{L_{cam}} \right), \tag{2.5}$$

where L_{cam} is a parameter defined by the distance between camera center and the vehicle's center of rotation. Thus, the new position of the vehicle, P_{new} is calculated from the previous position, P_{prev} and incremental translation in the X-axis, T and rotation matrix R rotated around the Z-axis by a heading angle when θ_{i+1}. This final output is derived from the following equation:

$$P_{new} = P_{prev} + R \times T. \tag{2.6}$$

C. Hybrid approach

This approach is a hybrid of feature-based and appearance based. The feature-based approach gives better results with textured images, such as urban environments with rough textures than in environments of a single pattern that have low textures (i.e: concrete, sandy soil). The lack of salient features to be detected in the images resulted in inefficiency of feature-based approach in such environments (González et al., 2012; Johnson et al., 2008; Nourani-Vatani et al., 2011).

On the other hand, appearance-based approach seems to be more robust and superior in low-textured images (Kicman et al., 2013). Therefore, in certain scenarios, hybrid approach appears to be the best solution by combining both techniques. This approach integrates tracking features between frames and at the same time use pixel intensity information of the image to obtain the optical flow. For instance, (Scaramuzza et al., 2008) implemented a hybrid-based approach where they utilized image features from the ground plane in estimating vehicle translation while vehicle rotation was estimated from the image appearance.

2.1.2 Prior VO Works

Majority of VO methods proposed in previous literature use either stereo or monocular camera in their systems. Monocular camera means that the system only utilizes a single camera for capturing surrounding images. Meanwhile, stereo systems use two cameras that for image capture that resembles human eyes in order to obtain object information in 3D. Both methods have advantages and disadvantages that will be explained in this subchapter.

A. Stereo system

During the early days of vehicle ego-motion system by visual inputs in the 1980s introduced by Moravec, a planetary rover with a single camera sliding on a rail was used (Moravec, 1980). The slider is called a slider stereo and the rover moved and stopped periodically. In each stop, the camera slid and captured nine images at an equal intervals. Motion was estimated by triangulation of 3D points seen at two consecutive positions after corners are detected and matched through normalized cross correlation (NCC). Although only one camera was used in this approach, it is classified as stereo VO algorithm due to the triangulation step. Basically, stereo cameras have been widely used in VO systems to capture a pair of RGB images at the same instant, thus the image scale can be retrieved by a given baseline.

Nister et al., who named the term 'visual odometry' was the first to demonstrate a real-time long-run implementation with a robust outlier rejection scheme (Nistér, 2003). They detected features independently in all frames and only allowed matches between features to avoid feature drift during cross correlation-based tracking. Since then, VO has received an increasing interest among researchers that proposed various stereo VO implementations

as presented by (Azartash et al., 2014; Geiger et al., 2011; Gomez-Ojeda et al., 2016; Howard, 2008; Kitt et al., 2010; Wang et al., 2017).

Algorithm proposed by Howard is generally consisted of feature detection in each frame pair, followed by feature matching. Then, the largest set of self-consistent matches (inliers) are identified. Finally, it calculates the frame-to-frame motion that minimizes re-projection error for features in the inlier set (Howard, 2008). In this method, the inlier detection step is the most important to distinguish feature, hence it is best described as 'inlier detection' method rather than 'outlier rejection'. On the other hand, (Kitt et al., 2010) suggested VO method with outlier rejection scheme by RANSAC. It is directly based on the trifocal geometry between image triples, resulting in less time consumption required for 3D scene structure recovery. Camera geometry is assumed to be known in this approach, and the authors employed an Iterated Sigma Point Kalman Filter (ISPKF) combined with RANSAC-based outlier rejection scheme to obtain robust motion estimation even in dynamic environments. This VO algorithm (known as Libviso) is made open-source which then allows further technique improvement as proposed by (Geiger et al., 2011).

Geiger et al. improved Libviso's performance by implementing 'stereoscan' where they proposed a sparse feature matcher for robust VO algorithm by dense 3D reconstruction from stereo sequences. The presented reconstruction pipeline combines four stages: sparse feature matching, ego-motion estimation, dense stereo matching and 3D reconstruction. This method managed to reduce running times of feature matching while at the same time allows real-time 3D reconstructions from large-scale imagery. It is then published publicly as Libviso2 that is still widely used for benchmarking.

Later in 2013, VO with Iterative Closest Multiple Lines (ICML) algorithm was proposed for an efficient line matching (Witt et al., 2013). This method contributed in fast and robust hypothesize-and-test algorithm which can act as a fallback for challenging frame pairs with textureless scenarios where pure gradient-based optimization fails. Since lines have proven to be an interesting alternative to points in man-made environments, Azartash then explored the combination of point and line features to robustly compute the six degree of freedom motion transformation between consecutive image pairs. This method, named MEVO (stands for 'multi-environment visual odometry'), achieved real-time speed of over 10 frames per second with good localization performance compared with Libviso2.

While MEVO proposed a direct combination of point and line features, (Gomez-Ojeda et al., 2016) proposed a probabilistic approach on the combination of both point and line segment for stereo VO. In order to effectively combine both types of features, projection errors of both point and line segment features were weighted according to their covariance matrices, computed from the propagation of Gaussian distribution errors in the sensor measurements. This resulted in more expensive computational cost, but it could still run in real-time and it provided an advantage of being able to be integrated straightforward into any probabilistic framework.

Recently, (Wang et al., 2017) had proposed a Stereo Direct Sparse Odometry (Stereo DSO) that aimed to integrate constraints from static stereo into the bundle adjustment pipeline of temporal multi-view stereo. With a system that combined static stereo with multi-view stereo, the absolute scale could be directly calculated, and initial depth could be estimated from the static stereo. Due to the fixed baseline, static stereo could only accurately triangulate 3D points within a limited depth range and this limit was resolved by temporal multi-view stereo.

Among the proposed stereo VO, some of the works are made available to public such as Libviso (Kitt et al., 2010), Libviso2 (Geiger et al., 2011) and stereo-DSO (Wang et al., 2017). This allows researchers to do VO performance comparison to study the strength and weakness in presented works. Besides, other well-known Simultaneous Localization and Mapping (SLAM) that utilized stereo VO in their algorithms are ORB-SLAM2 as proposed by (Mur-Artal et al., 2017) and RSO by (Moreno et al., 2016). In addition to localization capability, SLAM also performs mapping and loop closure for more accurate results although this adds computation complexity in order to achieve accurate performance.

B. Monocular system

If the distance to the scene from the stereo camera is much larger than the stereo baseline, stereo vision becomes ineffectual and it can be degraded to monocular system (Scaramuzza et al., 2011; Sünderhauf et al., 2007). Both of the relative motion and 3D structure are computed from 2D bearing data in monocular VO. Examples of recent VO works that implemented monocular system have been conducted by (Engel et al., 2013, 2015; Forster et al., 2014, 2017; Graeter et al., 2015; Lee et al., 2015a; Van Hamme et al., 2015).

(Van Hamme et al., 2015) had proposed a monocular VO algorithm that uses planar tracking of feature points on the ground plane surrounding the vehicle rather than conventional 3D pose estimation. To ease consistency of motion among features, tracking was applied in the ground plane coordinates instead of in the image coordinates of the perspective camera. Meanwhile, an online self-learning approach of monocular VO and ground classification were presented by (Lee et al., 2015a). They used a constrained kinematic model to solve the motion and structure problem besides to estimate the ground surface. The online self-learning scheme from monocular vision was achieved by a combination of probabilistic appearance-based ground classifier with geometric estimates.

(Graeter et al., 2015) on the other hand aimed to tackle issue on scale drift in monocular VO. To do so, the scene inherent information about the ground plane to estimate the scale was leveraged for usage on Advanced Driver Assistance Systems. A ground plane estimation using Structure From Motion (SFM) techniques was complemented by a vanishing point estimation to render the proposed algorithm robust in urban scenarios. A novel pitch-and-roll-tolerant scaling algorithm for monocular visual odometry, tailored for urban

environments was also proposed in their work. This was achieved by two complementary methods for ground plane normal estimation: a fit to reconstructed points and vanishing points. The scale was finally obtained from a trimmed Least Squares optimization that refined the plane.

Engel's and Forster's works are well-cited for their open-source codes on monocular VO widely known as semi-dense VO and semi-direct VO. Semi-dense VO was introduced by (Engel et al., 2013), where a semi-dense depth filtering formulation was presented to significantly reduce computational complexity. Previously, it was found that only pixels with an intensity gradient provide information for motion estimation (Dellaert et al., 1999). This semi-dense method was proposed by estimating the depth of all pixels with non-negligible image gradient and each estimate is represented as a Gaussian probability distribution over the inverse depth map for the current frame. Then, (Forster et al., 2014) proposed Semi-Direct VO combined direct tracking with keypoints, which achieved high frame-rates even on embedded platforms. This owed to their algorithm that operated directly on pixel intensities. Besides, more reliable points were obtained from a probabilistic mapping method that models outlier measurements to estimate 3D points.

Engel then developed his work on Large-Scale Direct Monocular SLAM (LSD-SLAM) which allowed to build large-scale, consistent maps of the environment. Along with highly accurate pose estimation based on direct image alignment, the 3D environment was reconstructed in real-time as pose-graph of keyframes with associated semi-dense depth maps. Slightly different from LSD-SLAM that employed a photometric error as well as a geometric prior to estimate dense or semi-dense geometry, (Wang et al., 2017) also proposed a novel method namely direct sparse odometry (DSO) that took full advantage of photometric camera calibration for further increase in accuracy and robustness.

2.1.3 Challenges and Limitations

While monocular system is admittedly the most efficient in terms of cost, calibration errors and scale uncertainty are the main challenges to obtain accurate localization with monocular camera. Furthermore, on uneven road surface, image scale has the tendency to fluctuate and the image scaling factor is also difficult to be estimated when there is a large change in the road slope or vehicle speed. The scale factor plays an important role in localization in order to compute the translation length in real world from the trajectory generated from image sequences.

Due to this, (Kitt et al., 2011) claimed that monocular vision systems are negatively affected by scale uncertainty that would fail the localization system without accurate scale factor. However, they also proposed a solution by using the Ackermann steering model and assumed that if the vehicle moves on a planar road surface, image scale could be recovered. (Guo et al., 2012) on the other hand proposed a monocular VO system with KLT feature

tracker for feature extraction and RANSAC was used for outlier rejection, while solving scale ambiguity problem by using the constraints of camera mounting and ground planar assumption. Unfortunately, this technique does not fully solve the scale problem in monocular systems but it limits the ambiguity to a certain degree.

Stereo camera, on the other hand, has the capability to obtain scale information from the image depth calculated by triangulation process with fixed stereo baseline size. However, stereo system is more expensive than monocular system and it requires more computation in calibration to obtain camera parameters. Besides, it is crucial for the pair of stereo camera to acquire images at an exact time interval which can be achieved by shutter speed synchronization or from external trigger signal. Still, many researchers prefer stereo systems due to its accuracy in localization and ability to obtain map depth information that allows map building in SLAM development.

2.2 Multisensor Localization

Accurate vehicle localization has been immensely researched within the last decades, in the development towards autonomous vehicle. While the technology has started to develop and been in the market in recent years, it does not stop the motivation for further research. Besides its application for autonomous vehicle, the current positioning and routing technology in vehicles should also be improved. There are still many issues need to be addressed as road network structure is becoming more complex with the urban development and this frequently cause interruptions in localization and path planning.

Although low-cost GPS are widely used for localization, it suffers from several conditions such as the multipath and Non-Line-Of-Sight (NLOS) effect especially in urban areas due to the dense buildings or other constructions like tunnels and bridges (Kos et al., 2010). (Zair et al., 2016) proposed to overcome GPS signal problem to improve its accuracy by detecting and removing the outliers. This resulted in reliable GPS data but their method consumed complex computation and the results is inconsistent particularly in biased GPS noise.

Therefore, data fusion with other sensors is desirable to overcome this problem. Multisensor vehicle localization can be from several sensors as presented previously and among those are lidar, GNSS receiver, camera sensor, Inertial Measuring Unit (IMU), and radar sensor. In addition, digital map can also be used as an input for the data fusion. These data and information can be used together, without requiring prior computation or data compensation, to provide a new information of an estimated state.

2.2.1 Fusion of GPS, Camera, INS and lidar

Among the common solution approach for the GPS signal degradation problem is to fuse GPS with other sensors, such as odometer, Inertial Measurement Unit (IMU), laser scanner and mono/stereo camera. One of the earliest fusion approach was by fusing data from vehicle GPS system and odometer data of the vehicle (Abuhadrous et al., 2003; Caron et al., 2006; Sasiadek et al., 2001; Sukkarieh et al., 1999). While odometer might have advantage such as minimal interference with other sensor signals, it suffers from limitations including position divergence over time, miscalculations caused by slippage or uneven roads. Besides, the accuracy is also affected by environment temperature and air pressure in tyres and the implementation requires intervention with the existing wheel sensor for data collection making it difficult to be built in standalone system.

Meanwhile, Inertial Navigation System (INS) which utilizes IMU sensor can also be used to collect the position data during GPS signal outage (Bencheikh et al., 2015). Unfortunately, its accuracy can only remain for short periods due to the accelerometer biases and gyroscope drifts (Caron et al., 2006). There are also studies on using laser scanner or lidar to obtain high resolution information of vehicle surroundings with GPS data fusion (Chong et al., 2013; Gao et al., 2015; Wolcott et al., 2014; Yoneda et al., 2014). The use of such dimensional sensors has been studied to find vehicle pose and brings to the implementation of SLAM (Simultaneous Localization And Mapping) (Durrant-Whyte et al., 2006). As mentioned previously, lidar is an active sensor that measures range distances by estimating the time-of-flight of laser beams to its surrounding objects. While a 2D lidar scans horizontal planes and is usually used to localize a robot on a flat area (Zhang et al., 2000), 3D lidar has also been studied for 6DOF localization (Moosmann et al., 2011).

In recent years, many works proposed VO fusion with GPS, odometer and lidar (Graeter et al., 2018; Pandey et al., 2011; Wolcott et al., 2014; Zhang et al., 2015). The proposed approaches showed performance accuracy of up to merely 0.60% in translation error as recorded from V-LOAM method by (Zhang et al., 2015). These studies exhibited interesting outcomes, but the high cost and long scanning period making it impractical for consumer level application.

On the other hand, (Aynaud et al., 2017) suggested a top-down approach in multisensor localization with triplets detection by lidar and choose the most efficient set of sensor, landmark, and detector. The results demonstrated that the top-down approach is applicable in vehicle localization. However, even with lidar, there were localization errors due to lack of landmarks detection. Another method was recently proposed by (Nguyen et al., 2017) to overcome GPS signal problem by WiFi finger printing, fused with lidar-based SLAM. However, despite the use of high performance sensor like lidar, the fusion output reached up to 8m of localization error during weak GPS signal. Therefore, thiese methods might require additional sensors to improve localization accuracy.

2.2.2 Low-Cost Data Fusion with Camera

Since cameras have relatively lower-cost and can also be integrated in the localization system, VO has become one of the popular approaches for localization. VO can be classified into monocular or stereo vision system in which stereo system usually performs better since it can avoid scale ambiguity inherent in monocular VO. Presented works for both monocular and stereo VO systems have been explained in the beginning of this chapter with their challenges and limitations. Therefore, to overcome this problem and achieve better localization, several approaches by implementing GPS fusion with VO and other sensors were proposed. For the sake of practicality and affordable implementation, fusion with lidar sensor is not discussed in this section.

Although VO trajectory itself is prone to accumulated drift and can be affected by scale ambiguity, by integrating it with other sensors would possibly solve this problem. For instance, in (Peker et al., 2014), monochrome camera and Kinect sensor were used to detect traffic sign and match with map information to improve vehicle localization. This highly relies on the availability of traffic sign recorded in map information hence is not optimized for global localization. Meanwhile, (Suhr et al., 2017) proposed sensor fusion-based low-cost vehicle localization system that fuses data from GPS, an IMU, a wheel speed sensor, a single front camera, and a digital map via the particle filter. This requires a low-volume digital map where road markings are expressed by a minimum number of points. However, aside from VO purposes, we found that most of the works in vehicle localization utilized cameras for road markings detection to improving positioning accuracy (Lee et al., 2015b; Schreiber et al., 2015; Suhr et al., 2017; Tao et al., 2013). Stop lane marking or intersection detections are used to correct the longitudinal position but there is a possibility of occluded lane markings or roads without lane markings that can degrade the localization performance. Besides, we also need to consider environmental influence such as weather and construction effects on the visibility of the markings. Hence, relying on the symbolic road or lane markings has some disadvantages for a robust localization.

On the other hand, other researchers such as (Floros et al., 2013) used Chamfer matching on the street graph with Monte Carlo localization to match the vehicle trajectory with the road network and (Brubaker et al., 2016) utilized road network information from OpenStreetMap by identifying the actual link between vehicle and road map through stereo vision system during GPS signal outage. Both of these methods have shown improvised localization trajectory of the vehicle on the correct road, but lane-level and longitudinal positioning accuracy has yet to be addressed. In (Floros et al., 2013), the method performed an exhaustive matching over all possible positions and orientations against the template edge map. Despite this, its localization error of has yet to achieve below 10m of average on two datasets tested as presented in the paper. As for (Brubaker et al., 2016), the proposed approach managed to achieve an average of 3.7m positioning error but there were cases of

ambiguous road networks that resulted in failed localization.

2.3 Datasets for VO Benchmarking

For VO localization performance benchmarking, several datasets have been published to the public that mostly contain data collected using IMU, GPS, cameras, and lidars. The datasets are KITTI dataset (Geiger et al., 2012), CMU Visual Localization dataset (Badino et al., 2011b), and Malaga dataset (Blanco et al., 2014). Among these, KITTI dataset has the widest coverage, consisting sequences in city, residential, highway road and in campus environment.

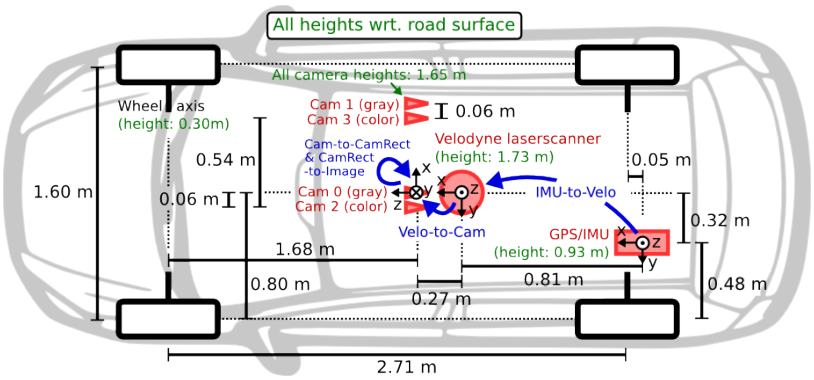


FIGURE 2.2: Top view of sensors setup

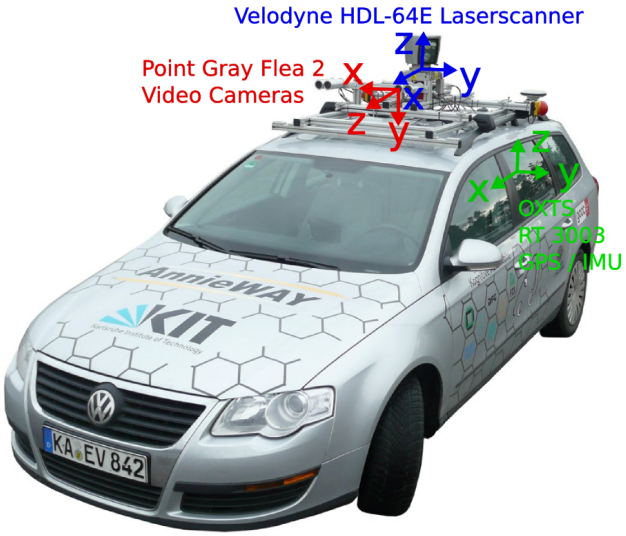


FIGURE 2.3: Front view of moving platform used to collect data

The KITTI dataset has been recorded from a vehicle driven in and around Karlsruhe, Germany. It includes stereo camera images, laser scans, high-precision GPS measurements

and IMU accelerations from a combined GPS/IMU system and the setup is depicted in Figure 2.2 while its real implementation can be viewed in Figure 2.3. The main purpose of this dataset is to encourage the development of computer vision and robotic algorithms targeted to autonomous driving. The development kit and calibration file is also made available online for test validation. This makes it mostly used by researchers to evaluate and compare their state-of-the-art computer vision methods.

VO methods that has been published are compiled for performance comparison as shown in Table 2.1. Some of the methods also included their codes online for public reference. From all test sequences provided by KITTI, the methods evaluation computes translational and rotational errors for all subsequences of length (every 100 metres). The evaluation table shows the ranking of methods according to the average of both errors. Methods marked as (*) denote the availability of its source code online. Meanwhile, (**) indicates that the open source methods did not use lidar in their systems. The KITTI dataset can be obtained from http://www.cvlibs.net/datasets/kitti.

TABLE 2.1: Performance comparison on VO / SLAM methods conducted on KITTI dataset

Method	**Translation**	**Rotation**
V-LOAM (Zhang et al., 2015)	0.60 %	0.0014 [deg/m]
LOAM (Zhang et al., 2014a)	0.61 %	0.0014 [deg/m]
SOFT2 (Cvišić et al., 2017)	0.65 %	0.0014 [deg/m]
IMLS-SLAM (Deschaud, 2018)	0.69 %	0.0018 [deg/m]
RotRocc+ (Buczko et al., 2016a)	0.83 %	0.0026 [deg/m]
* LIMO2_GP (Graeter et al., 2018)	0 .84 %	0.0022 [deg/m]
GDVO (Zhu, 2017)	0.86 %	0.0031 [deg/m]
* LIMO2 (Graeter et al., 2018)	0.86 %	0.0022 [deg/m]
SOFT (Cvišić et al., 2015)	0.88 %	0.0022 [deg/m]
RotRocc (Buczko et al., 2016a)	0.88 %	0.0025 [deg/m]
DVSO (Yang et al., 2018)	0.90 %	0.0021 [deg/m]
* LIMO (Graeter et al., 2018)	0.93 %	0.0026 [deg/m]
Stereo DSO (Wang et al., 2017)	0.93 %	0.0020 [deg/m]
ROCC (Buczko et al., 2016b)	0.98 %	0.0028 [deg/m]
cv4xv1-sc (Persson et al., 2015)	1.09 %	0.0029 [deg/m]
MonoROCC (Buczko et al., 2017)	1.11 %	0.0028 [deg/m]
DEMO (Zhang et al., 2014b)	1.14 %	0.0049 [deg/m]
** ORB-SLAM2 (Mur-Artal et al., 2017)	1.15 %	0.0027 [deg/m]
NOTF (Deigmoeller et al., 2016)	1.17 %	0.0035 [deg/m]
** S-PTAM (Pire et al., 2017)	1.19 %	0.0025 [deg/m]

**S-LSD-SLAM (Engel et al., 2015)	1.20 %	0.0033 [deg/m]
VoBa (Tardif et al., 2010)	1.22 %	0.0029 [deg/m]
LiViOdo (Graeter et al., 2018)	1.22 %	0.0042 [deg/m]
SLUP (Qu et al., 2018)	1.25 %	0.0041 [deg/m]
FRVO (Wu et al., 2017)	1.26 %	0.0038 [deg/m]
MFI (Badino et al., 2013)	1.30 %	0.0030 [deg/m]
TLBBA (Lu et al., 2013)	1.36 %	0.0038 [deg/m]
2FO-CC (Krešo et al., 2015)	1.37 %	0.0035 [deg/m]
SuMa (Behley et al., 2018)	1.39 %	0.0034 [deg/m]
** ProSLAM (Schlegel et al., 2017)	1.43 %	0.0040 [deg/m]
StereoSFM (Badino et al., 2011a)	1.51 %	0.0042 [deg/m]
** SSLAM (Fanfani et al., 2016)	1.57 %	0.0044 [deg/m]
eVO (Sanfourche et al., 2013)	1.76 %	0.0036 [deg/m]
Stereo DWO (Huai et al., 2015)	1.76 %	0.0026 [deg/m]
BVO (Pereira et al., 2017)	1.76 %	0.0036 [deg/m]
D6DVO (Comport et al., 2007)	2.04 %	0.0051 [deg/m]
PMO / PbT-M2 (Fanani et al., 2017b)	2.05 %	0.0051 [deg/m]
** SSLAM-HR (Fanfani et al., 2016)	2.14 %	0.0059 [deg/m]
FTMVO (Mirabdollah et al., 2015)	2.24 %	0.0049 [deg/m]
PbT-M1 (Fanani et al., 2017a)	2.38 %	0.0053 [deg/m]
**VISO2-S (Geiger et al., 2011)	2.44 %	0.0114 [deg/m]
MLM-SFM (Song et al., 2014)	2.54 %	0.0057 [deg/m]
GT_VO3pt (Beall et al., 2010)	2.54 %	0.0078 [deg/m]
RMCPE+GP (Mirabdollah et al., 2014)	2.55 %	0.0086 [deg/m]
VO3pt (Alcantarilla, 2011)	2.69 %	0.0068 [deg/m]
TGVO (Kitt et al., 2010)	2.94 %	0.0077 [deg/m]
VO3ptLBA (Alcantarilla, 2011)	3.13 %	0.0104 [deg/m]
PLSVO (Gomez-Ojeda et al., 2016)	3.26 %	0.0095 [deg/m]
BLF (Velas et al., 2018)	3.49 %	0.0128 [deg/m]
CFORB (Mankowitz et al., 2015)	3.73 %	0.0107 [deg/m]
VOFS (Kaess et al., 2009)	3.94 %	0.0099 [deg/m]
VOFSLBA (ibid.)	4.17 %	0.0112 [deg/m]
BCC (Velas et al., 2018)	4.59 %	0.0175 [deg/m]
EB3DTE+RJMCM (Boukhers et al., 2018)	5.45 %	0.0274 [deg/m]
** VISO2-M + GP (Geiger et al., 2011)	7.46 %	0.0245 [deg/m]
BLO (Velas et al., 2018)	9.21 %	0.0163 [deg/m]
** VISO2-M (Geiger et al., 2011)	11.94 %	0.0234 [deg/m]

OABA (Frost et al., 2016)	20.95 %	0.0135 [deg/m]

2.4 Problem Statement

From the performance evaluation data conducted on sequences provided by KITTI dataset, we find an interesting similarity in all techniques without lidar fusion that recorded worst performance in Sequence 01 among 11 sequences tested. In this particular sequence, the drive was conducted on a highway with long stretch of a straight road. Other sequences with many turns, curved roads, and junctions performed well in localization. This shows that without high precision information from lidar, it is challenging to obtain accurate localization on a straight road, most possibly due to drift and longitudinal error.

Besides, limitations in VO systems are identified regarding the scale uncertainty and calibration errors that affects its localization performance. As presented in the evaluation table, VO methods accuracy may vary from 20.95 % & 0.0135 [deg/m] to 0.86 % & 0.0031 [deg/m] of translational and rotation error, in accordance to system complexity. From the varying results, rotational error of VO methods is not severe as its translational error. Therefore, we can make three hypotheses that:

- the problem lies more on the scale ambiguity,
- trajectory direction maintains good accuracy due to small rotational error, and
- with the consistent direction accuracy, trajectories from VO method would mostly concede with road network shape on map.

These hypotheses are also in line with our finding regarding the higher localization error for drive sequence on a straight road – localization performs worse when there is less heading rotation during the drive.

2.5 Conclusion

Since its introduction in the early 21st century, VO has been studied vigorously for its application in mobile robotics and vehicle localization. Aside from having better performance than wheel odometer, it can even replace the use of laser scanner for map building in SLAM methods (Cvišić et al., 2017; Engel et al., 2015; Fanfani et al., 2016; Mur-Artal et al., 2017; Schlegel et al., 2017). At the same time, vehicle equipped with camera allows other intelligent systems to be deployed for advanced driver-assistance systems (ADAS) such as pedestrian detection and road signs recognition.

Although we see vast development in SLAM research with highly accurate localization results, it requires more complex computation for map building and with loop closure detection to compensate drift error that would result in delayed localization correction. This

is practical for mobile robot applications in an unknown environment and does not require high accuracy positioning in real time. However, as for vehicle localization, digital map of the world is publicly available, hence new map building is deemed unnecessary if one's position can be estimated on earth. Besides, with the abundant information provided by digital map nowadays, we believe that VO trajectory can be useful for global localization with map fusion and its performance can be on par with highly accurate GPS or lidar if optimized correctly. Furthermore, with the integration of low-cost GPS data, rough position of the vehicle can be estimated and this can reduce unnecessary computation of map area outside region of interest. The next chapter demonstrates our fusion technique that integrates the information from GPS, VO and digital map for localization accuracy improvement.

Chapter 3

Data Fusion with Road Probability Distribution Factor for Localization Improvement

As one of the approaches to non-stationary inverse problems, state estimation has been of a great interest in many practical applications (Kaipio et al., 2006). In such cases, the available data obtained from measurement of environment information are fused together with the prior knowledge of state condition to continuously produce estimates of the desired dynamic outputs. This can be accomplished in such a manner that the error is minimized statistically after the predicted state is corrected with additional information (Maybeck, 1982).

For instance, the position of a moving object can be estimated through the time integration of its velocity components since departure. In our case, the prediction of the vehicle position is based on visual odometry displacement. Then, this prediction will go through a correction phase, where the assumptions are measured according to the likelihood. GPS data and information from OSM are used in this phase to narrow down the possibility of obtaining the best position estimation. In short, the vehicle state estimation problem deals with the combination of the movement prediction from the displacement measurements (the integration of the velocity components may contain measurement errors), then corrected by the GPS data that also has no guarantee to be error-free, together with OSM data. This fusion is conducted in order to obtain more accurate estimations of the system variables, which is the vehicle position. In this chapter, the fusion technique is presented with our proposed Road Probability Distribution Factor (RPDF) method based on OSM data to observe

the improvement in localization performance.

3.1 Filters for Data Fusion

State estimation problems are commonly solved with the Bayesian filters (Gordon et al., 1995; Kalman, 1960; Maybeck, 1982) as a method in data fusion. In the Bayesian approach, an attempt is made to utilize all available information in order to reduce the amount of uncertainty present in an inferential or decision-making problem and as new information is obtained, it is combined with previous information to form the basis for statistical procedures (Orlande et al., 2008). The formal mechanism used in the information fusion with the previously available knowledge of the system state is known as Bayes' theorem (Bayes et al., 1763; Winkler, 1972).

Kalman filter is the most widely known Bayesian filter method to date which has undergone many improvements by researchers (Kalman, 1960; Maybeck, 1982; Sorenson, 1970; Weiss et al., 1980; Winkler, 1972). However, the application of the Kalman filter is restricted to linear models with additive Gaussian noises. Some extensions of the Kalman filter were developed in the past years to partly overcome the restriction by using linearization techniques (Arulampalam et al., 2002; Kaipio et al., 2006; Ristic et al., 2004). The extended Kalman filter performs linearization around the estimation state and force the filter to use the linearized version as a model.

At the same time, Monte Carlo methods have also been developed in order to represent the posterior density in terms of random samples with associated weights (Doucet et al., 2001). The Monte Carlo methods, also known as particle filter among other designations found in the literature, has the advantage of not requiring the restrictive hypotheses of the Kalman filter. Therefore, the particle filter can be applied to non-linear models with non-Gaussian errors and is more suitable for the real-world applications.

3.1.1 Kalman Filter

The Kalman filter (Kalman, 1960), is one of the most popular data fusion algorithms in the field of information processing. It is typically derived by using vector algebra as a minimum mean squared estimator from first principles considering a simple physical example exploiting a key property of the Gaussian distribution (Bandemer, 1978). The approach is mean to translate some original observables y_k into Wold decomposition (Wold, 1938) of innovations and estimates of the state x_k. From the innovations, the likelihood function of the dynamic model can be built and with the states estimates, it allows us to forecast and smooth out the stochastic process.

Behaving as a recursive estimator, only the estimated state from the previous time step and the current measurement are required for the computation of the current state estimation in Kalman filter. It is different with the batch estimation techniques where it requires history of observations and its estimates. The state in this method is represented by two variables; x_k, as the posteriori state estimate at time k, provided observations up to time k, and P_k which denotes the posteriori error covariance matrix to measure the accuracy of the state estimation.

The Kalman filter is most often conceptualized into two distinct phases: "Prediction" and "correction". The *prediction* phase uses the state estimation from the previous time step to calculate a new estimate of the state at the current time step. This predicted state estimation is also regarded as the a priori state estimate because although it is an estimate of the state at the current timestep, it has yet to include observation information from the measurement at the current timestep. Meanwhile in the *correction* phase, the priori predicted state is combined with the current observed information to refine the state estimate. This improved updated estimate is termed as the 'posteriori state estimate'.

With its fusion technique development, some variants such as the Extended Kalman Filter (EKF) (Groves, 2013; Hoshiya et al., 1984; Weiss et al., 1980) and unscented Kalman filter (UKF) (Julier et al., 2004; Wan et al., 2000) that involves linearization of the problem were developed. In systems with nonlinear dynamics, the observation equations are linearized by Taylor series expansion or partial derivatives of nonlinear state functions. However, the EKF is deemed to be lack of stability and converges very slowly to the correct solution. Most of the times, it only performs well with small nonlinearities, making it impractical for highly nonlinear models.

To solve the limitations in EKF, UKF was introduced where it is based on Unscented Transformation (UT). It uses a deterministic sampling approach to obtain the mean and covariance estimates from a finite set of sample points. These points are chosen deterministically and transformed to a new space via nonlinear process to describe the distribution statistics. The UKF is an expansion of UT which consists of initialization, prediction and actualization steps. Compared with EKF, the UKF has an advantage of being independent on the local approximation, allowing it to perform in higher order nonlinear models. Unfortunately, similar with other Kalman filter, the UKF can only be used for models driven by Gaussian noises and it is not a global approximation, due to the small set of sample points.

In other cases where the hypotheses of linear Gaussian evolution-observation models are invalid, the use of the Kalman filter does not result in optimal solutions since the posterior density is not analytic - even with the use of EKF and UKF filter because they still assume Gaussian distribution in the derivation. Thus, the application of Sequential Monte Carlo technique appears as the most general and robust approach to non-linear and/or non-Gaussian distributions (Arulampalam et al., 2002; Johansen et al., 2008; Kaipio et al., 2006; Liu et al., 1998). More details on Sequential Monte Carlo are explained in the following

section.

3.1.2 Particle Filter

The Sequential Monte Carlo technique, also known as the particle filter method, also offers a solution in the state estimation problem. Other names for the particle filter include bootstrap filter, condensation algorithm, interacting particle approximations and survival of the fittest (Arulampalam et al., 2002). The main idea is to represent the required posterior density function by a set of random samples (particles) with associated weights, and to compute the estimates based on these samples and weights. When the number of samples are very large, this Monte Carlo characterization behaves as an equivalent representation of the posterior probability function, and the solution approaches the optimal Bayesian estimate.

One of the particle filtering method is the Sequential Importance Sampling (SIS) algorithm, which includes a sampling step at each instant (Arulampalam et al., 2002; Ristic et al., 2004). This algorithm involves using importance density, which is a density proposed to represent the sought posterior density in the present case to solve the recursion equation. Then, samples are drawn from the importance density. In certain importance functions, the variance of the importance weights can increase over time (Kong et al., 1994). This leads to the degeneracy phenomenon and it has a harmful effect on the estimation accuracy because after a few steps, it is possible that all but one particle will have negligible weight. This results in a wasteful amount of computational effort that will be devoted to updating a contribution that has almost zero weight. One way to solve this degeneracy problem is by increasing the number of samples, N, which is impractical and costly. Another way is by minimizing degeneracy through a good choice of importance function and by including a resampling step in the SIS algorithm which brings the evolution to the Sequential Importance Resampling (SIR) algorithm as illustrated in Figure 3.1.

Resampling is conducted by mapping of the random measure $\{x_k^i, w_k^i\}$ into a new random measure $\{x_k^j, N^{-1}\}$ with uniformed importance weights. It is executed when the number of effective particles (or effective sampling size) with large weights falls below a certain threshold number. Alternatively, resampling can also be applied indistinctively at certain time t_k, as in the Sampling Importance Resampling (SIR) algorithm described in (Arulampalam et al., 2002; Ristic et al., 2004). SIR algorithm is summarized in the following steps, as applied in our data fusion in section 3.4.2:

i In the prediction stage, generate particles based on the predictive density of respective old particles x_{k-1}^i:

$$x_k^i \sim p(x_k \mid x_{k-1}^i). \tag{3.1}$$

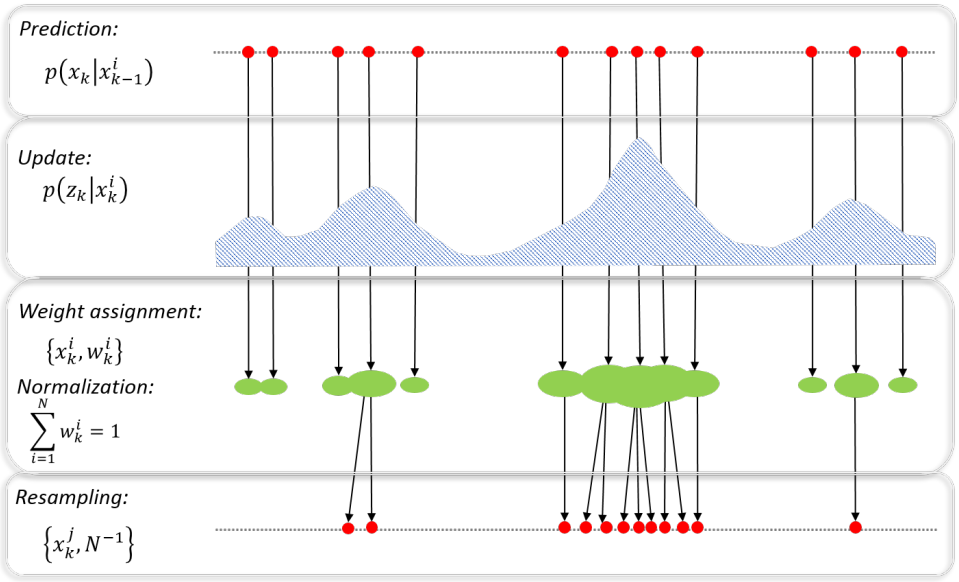


FIGURE 3.1: Representation of the Sampling Importance Resampling (SIR) algorithm of the Particle filter

ii The correspondent importance weights are updated based on the likelihood density up to a normalizing constant:

$$w_k^i = w_{k-1}^i \frac{p(z_k \mid x_k^i) p(x_k^i \mid x_{k-1}^i)}{\pi(x_k^i \mid x_{0:k-1}^i, z_{1:k})}. \tag{3.2}$$

iii Total weight of all particles, T_w is calculated

$$T_w = \sum_{i=1}^{N} w_k^i, \tag{3.3}$$

and the particles weights are normalized

$$\tilde{w}_k^i = \frac{w_k^i}{T_w}. \tag{3.4}$$

iv Effective number of particles is computed as

$$N_{eff} = \frac{1}{\sum_{i=0}^{N} (\tilde{w}_k^i)^2}. \tag{3.5}$$

v When the effective number of particles is smaller than given threshold, $N_{ess} < N_{thres}$, resampling occurs where samples with lower weights are removed and replaced by duplicating those with higher weights. This is done by constructing the cumulative sum of

weights with $c_0 = 0$, for $i = 1, ..., N$;

$$c_i = c_{i-1} + w_k^i. \tag{3.6}$$

Then, new particles are assigned with probabilities proportional to their importance weights while particles with lower weight are dismissed. Then, after resampling, all particles has uniformed weight of N^{-1}.

3.1.3 Summary

There has been extensive research on the usage of filter techniques for position estimation. State estimation by means of Kalman Filter (KF), as well as Extended Kalman Filter (EKF) and Particle Filter (PF) have been reported in studies for several applications. Brief comparison between filter techniques for position estimation was presented in (Toro et al., 2015). It shows that PF does not require the restrictive hypotheses of the KF approaches, which makes PF a better solution approach for non-linear models for multi-sensor data fusion (Cappello et al., 2015). Besides, PF provides a natural way to incorporate road map information into vehicle position estimation and it is capable of handling multi-modal distributions.

Thus, in this research, PF is chosen over other fusion technique for its ability to perform multimodal and non-linear estimation. Besides, PF is proved to be more robust and superior than the EKF in terms of the accuracy of state vector estimation (Lin et al., 2002; Rigatos, 2010). PF uses random samples (particles) to represent the posterior density of vehicle position in a dynamic state estimation framework such as a road network. With PF, we estimate the instate-space models inference by expressing the posterior probability density in terms of randomized particles associated with importance weights. However, due to the randomized nature of PF, it has a drawback where the exact result generation is impossible. Every new run of the program will produce different set of results for evaluation.

3.2 Conventional Visual Odometry

The initial approach of our research adopted a conventional visual odometry technique to fuse with GPS data for vehicle localization. Visual odometry, as illustrated in Figure 3.2, is the estimation of position and orientation of a moving object by analysing sequence of images from visual sensor attached to the object. The input would be a stream of images, captured at time k and $k + 1$, referred as I_k and I_{k+1}. These image pairs are compared to obtain translation vector t and rotation matrix R to describe object movement whereby with

heading rotation θ, rotation matrix is represented as

$$R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.7}$$

and vehicle pose can be defined as $X_k^\top = [x_k, y_k, z_k]^\top$.

For image feature detection, FAST algorithm (Rosten et al., 2006) is applied in our approach due to its computational efficiency compared with other feature detection methods such as SIFT (Lowe, 1999) and SURF (Bay et al., 2008) as described in Chapter 2. The features detected by FAST are then tracked on the preceding image by using Kanade-Lucas-Tomasi (KLT) feature tracker algorithm (Tomasi et al., 1991). The tracker utilizes spatial intensity to approximate the search for the position that yields the best match by finding sparse pixel wise correspondences. It is expected to work well with corner-like features for matching and tracking. New feature detection in images is only repeated after number of tracked features falls below a certain value.

From the tracked features, the Essential Matrix E is computed by random sample consensus (RANSAC) method which has an iterative algorithm (Nistér, 2003). For every iteration, five points are sampled from the set of correspondences, and the essential matrix is estimated. Then, total number of other points which are inliers when using the essential matrix is accumulated. After a fixed number of iteration, the essential matrix with maximum number of inlier points is selected. The essential matrix can be decomposed into rotation and translation as described in (Hartley, 1995)

$$E = [t]_\times R, \tag{3.8}$$

where $[t]_\times$ denotes the cross product matrix with translation vector $t = [t_x, t_y, t_z]^\top$ and can also be written as

$$[t]_\times = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}. \tag{3.9}$$

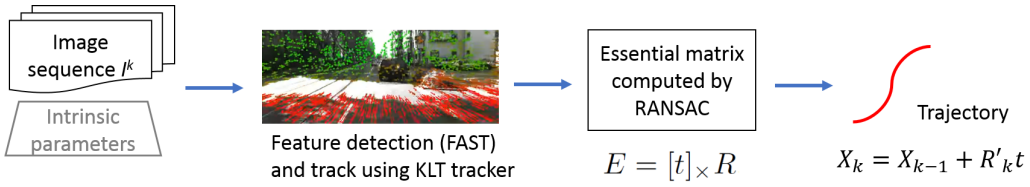


FIGURE 3.2: Conventional VO algorithm applied in our approach

Taking singular value decomposition of essential matrix (Longuet-Higgins, 1987) with two equal singular values $(D : diag(1, 1, 0))$ this can be solved by

$$E = UDV^{\top}, \tag{3.10}$$

where the columns of U and V are 3×3 orthogonal matrices which consist of the left and right singular vectors. Here, W is defined as

$$W = \begin{bmatrix} 0 & \pm 1 & 0 \\ \mp 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.11}$$

There are two possibilities of rotation matrix, R value as $R_1 = UWV^{\top}$ and $R_2 = UW^{\top}V^{\top}$. However, via triangulation, only one solution is feasible from the condition called cheirality constraint and an efficient decomposition of E into R and t is further described in (Nistér, 2003).

Translation matrix can then be computed from the obtained R as shown in 3.12.

$$\begin{bmatrix} x_k \\ y_k \\ z_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ z_{k-1} \end{bmatrix} + \begin{bmatrix} \cos\theta_k & -\sin\theta_k & 0 \\ \sin\theta_k & \cos\theta_k & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}. \tag{3.12}$$

Then, output trajectory of the visual odometry for the current image sequence I^k is presented as

$$X_k = X_{k-1} + R'_k t, \tag{3.13}$$

where the rotation matrix R_k is updated from the relative rotation matrix of each image pair as

$$R'_k = R_k \times R_{k-1}. \tag{3.14}$$

Here, we implemented a conventional visual odometry approach without accuracy optimization with the objective of obtaining the trajectory. We assumed that even without proper optimization, the trajectory curve has enough information to describe the position estimate of the vehicle with data fusion.

3.3 OSM Data Structure

Map-matching is a process to match a sequence of real-world coordinates into a digital map. Its main purpose is to identify the correct road link on which the vehicle is moving and to determine the vehicle location on that link (Quddus et al., 2006). Map-matching is not only

able to localize the vehicle position, but also improves the positioning accuracy as long as good spatial road network data are available.

The OSM uses a topological data structure with four core elements:

- *Nodes*: points with a geographic position coordinate stored in latitude and longitude according to WSG84.
- *Ways*: ordered lists of nodes representing a polyline, or a polygon if they form a closed loop. They are used to represent linear features such as streets and rivers, and areas, like forests and lakes.
- *Relations*: ordered lists of nodes, ways and relations. They are used for representing the relationship of existing nodes and ways such as turn restrictions on roads, routes that span several existing ways, and areas with holes.
- *Tags*: key-value pairs (both arbitrary strings). They are used to store metadata about the map object properties. Tags are always attached to a node, a way or a relation.

These *ways* contain *node* reference to locate the coordinates of road connection and tags that has the road specification. In our proposed method, other map attributes are neglected to focus on road network to be rendered by using OSM Mapnik rendering rules so only *ways* of roads labelled as *tags* with *key=highway* are extracted. These *highway* in map data is specified with information that may include road name, type, number of lanes, allowed directions, maximum speed, and even the road surface material.

An example of ways formed by connected nodes is shown in Figure 3.3 with its data structure recorded in an .osm file. Nodes are assigned unique *ref* each as reference number and the ways can be specified by their respective *id*s. Within the way description, metadata is stored in *tags* that contain abundant information of the way.

The first step in our approach for fusion with OSM data is by filtering only the road network and use Mapnik rendering style to estimate the road width on the map. Then, mask image is created from the rendered map to determine road and non-road area on map. With this, the probability factor when the particle falls within the road area, is defined as $\alpha^i = 1$. Otherwise, the value falls exponentially with distance from road boundary, d_r and calculated as $\alpha^i = e^{-d_r}$.

3.4 Fusion Validation of VO with GPS and OSM

The conventional VO approach is our main input for state estimation during prediction for our fusion technique. The trajectory obtained from VO is fused with noisy GPS data to obtain first estimation of vehicle pose on map. Then, further correction on the state estimation is conducted based on the road width information rendered from the OSM. The fusion is expected to be able to reduce VO drift and increase localization accuracy at the same time.

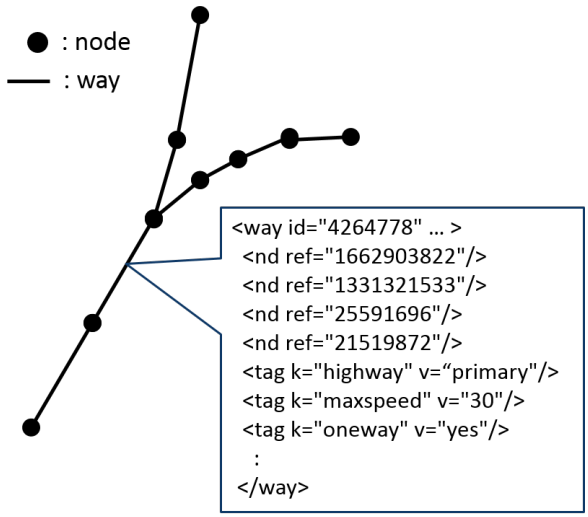


FIGURE 3.3: Example of road representation in OSM map and the data structure in .osm file

3.4.1 KITTI Dataset

In this work, KITTI dataset (Geiger et al., 2012) was utilized to validate the efficiency of data fusion between VO, GPS and OSM. The database contains sequence of images along with data from velodyne sensor and GPS. Since the GPS data is obtained from a high precision device that has up to 0.01m accuracy, it is treated as the ground truth in method validation. To test the ability of our method to deal with noisy low-cost GPS positions, we demonstrated the random noise of about 10m, added to the ground truth data with 1Hz of frequency. Short sequences were used to observe the accuracy and precision level of our proposed approach in lane-level localization.

Our proposed approach was performed on Dataset 1 on 400 images to verify the improved accuracy of visual odometry and noisy GPS fusion result on a straight road with double lane and two-way characteristics in Dataset 1 and a multilane road that diverged to a curved road in Dataset 2. Trajectory output was compared with ground truth data to acquire the absolute distance error. In the results section, localization accuracy and precision is defined by distance error mean and standard deviation, respectively.

3.4.2 Data Fusion by Particle Filter

Particle filtering method serves the purpose for performing inference interstate-space models where it represents the posterior density in terms of random samples and associated weights. The first step in this research after obtaining visual odometry trajectory is to fuse output from visual odometry with noisy GPS to determine the functionality of the filter and compare the accuracy with visual odometry localization. Particle filter consists of several steps as illustrated in Figure 3.4 and further described below:

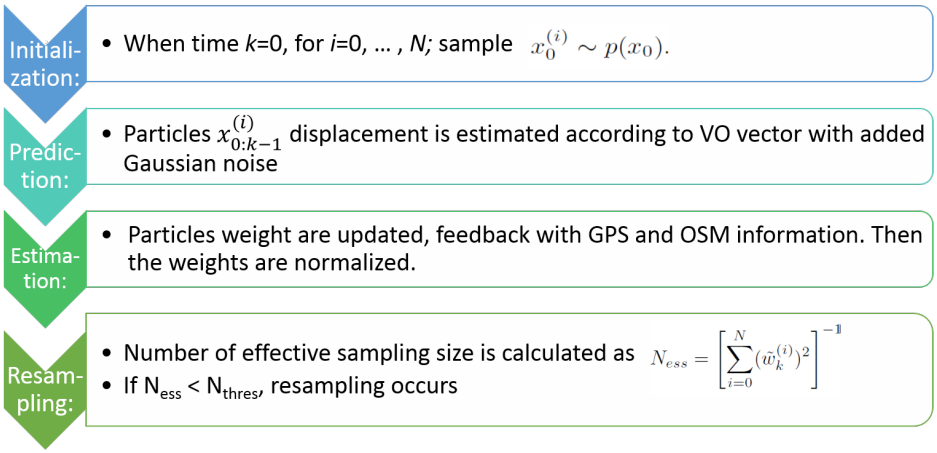


FIGURE 3.4: Data fusion steps in particle filter

i *Initialization*: In the beginning when time $k = 0$, N number of particles are generated around $X_0$with a non-linear model with equal weight $1/N$. Therefore, for $i = 1, ..., N$, sample x_0^i

$$x_0^i \sim p(X_0). \tag{3.15}$$

ii *Prediction*: For $i = 1, ..., N$, particles are displaced according to the VO trajectory vector (Equation 3.12) transition equation, f_k with dynamic noise, v which is random and independently calculated for each particle i. This will disperse the sample movements to find the probability in a wider area.

$$x_k^i = f_k(x_{k-1}^i, v_k^i) \tag{3.16}$$

iii *Weight update*: Using GPS data and particle position on the road as the new measurement vector, weight of each particle is updated according to probability density function as

$$w_k^i = w_{k-1}^i \alpha^i \frac{e^{\frac{-d^2}{2\sigma^2}}}{\sigma\sqrt{2\pi}}, \tag{3.17}$$

where

- d : relative distance between particle x_k^i and GPS coordinate,
- σ : coefficient value of the probability distribution determined from GPS error variance in the system,
- α^i : particle position whether inside/outside the road area.

Before resampling, the probability distribution function particle weight is normalized to $\tilde{w}_k^i$ so that $T_w = 1$.

iv *Resampling*: Number of effective sampling size is calculated and if $N_{eff} > N_{thres}$, resampling occurs. During the resampling step, we utilize the most common resampling method called 'select with replacement' by duplicating particles with higher weight and removing particles with negligible weights (Hoover, 2016). Firstly, the particles are sorted according to its importance weight. Then, the resampling algorithm is shown as follows:

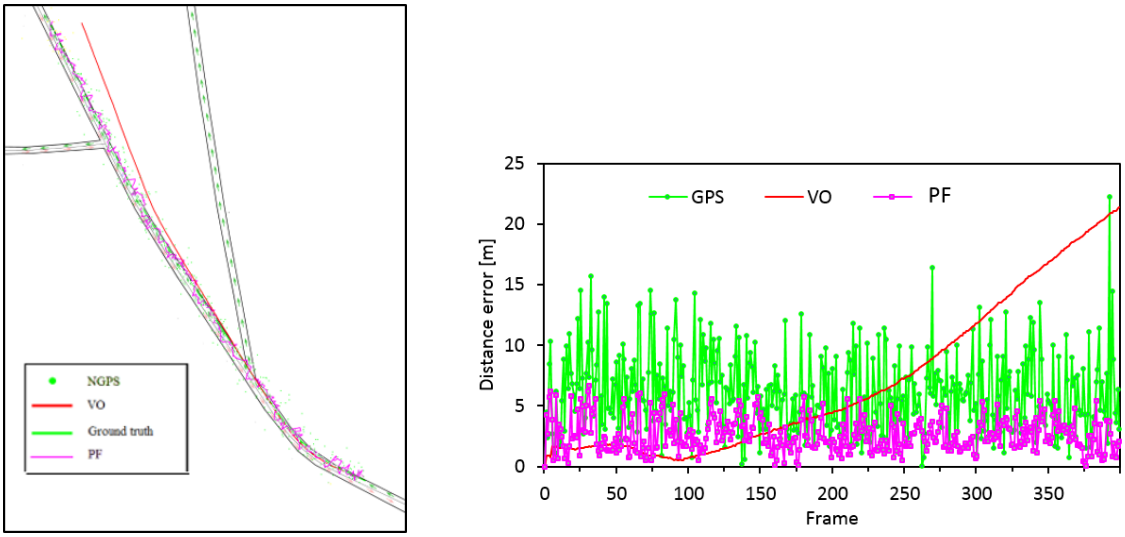
Algorithm 1 Resampling steps

```
Q = cumsum(W);          # cumulative sum of particle weights
X = sort(x, w);         # particles are sorted as an array, based on weights
i = j = 1;              # array starts
while (i <= N) do
    if (X[i] < Q[j]) then
        index[i] = j;
        i = i + 1;
    else
        j = j + 1;
    end if
end while
for (i = 1; i <= N; i = i + 1) do
    NewP[i] = P[index[i]];   # new particles are duplicated
    NewW[i] = 1/N;           # weights are distributed equally to 1/N
end for
```

3.4.3 Results and Discussion

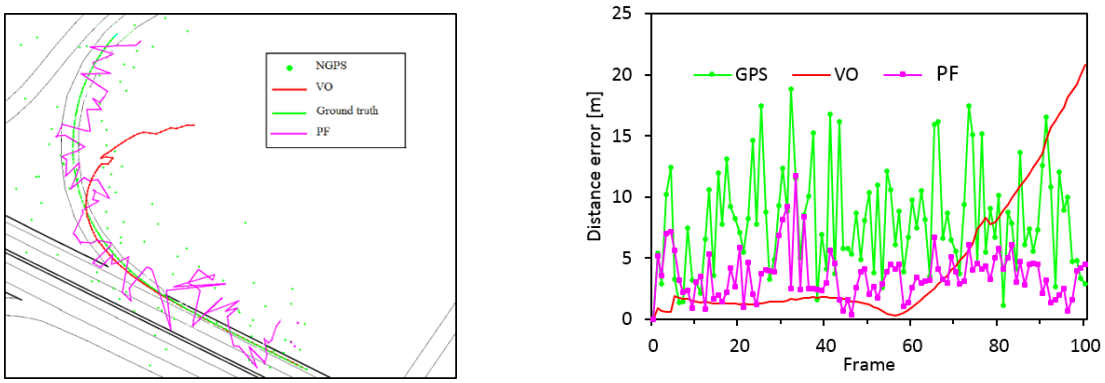
Firstly, we observe that the visual odometry trajectory as shown in Figure 3.5a suffered from drift error, due to the inaccurate scale estimation. This was expected since no optimization was done on our visual odometry technique. However, we can see that the drift was greatly reduced with the fusion of GPS and OSM as illustrated in Figure 3.5b, despite the trajectory drift after frame 200. The distance error achieved for this dataset was (2.619±1.397)m.

The same test was conducted to a different dataset (Dataset 2) that has highway road with multiple lane and vehicle movement behaviour of turning into a diverged exit as shown in Figure 3.6a. Here, our visual odometry contained severe error that mainly affected the trajectory scale on the curved road. Besides, there was also an instant of noisy visual odometry trajectory due to the shadow and lighting condition around frame 75 to 79 but since it only occurred within a short period, this did not affect the fusion performance.



(A) Improved vehicle trajectory (B) Distance error

FIGURE 3.5: Localization results for Dataset 1



(A) Improved vehicle trajectory (B) Distance error

FIGURE 3.6: Localization results for Dataset 2

The distance error comparing visual odometry and fusion output of GPS and visual odometry is displayed in Figure 3.6b, whereas the error mean and standard deviation obtained for our data fusion was (3.608 $\pm$ 1.968)m.

3.5 Road Distribution Probability Factor

3.5.1 Fusion with OSM Data

As mentioned in Section 3.3, OSM provides map data including road information such as road type, number of lanes, direction and maximum speed limit. Meanwhile road width

is either specified in the .osm file itself or predefined by the rendering rule. This rule is constantly updated and the road width is defined according to the specified road type. Previously, the validation results were generated to exhibit the localization improvement of fusion between noisy GPS, conventional VO, and road boundary from OSM. Next, we extract number of lanes and road direction information from OSM for Road Probability Distribution Factor (RPDF) to improve positioning estimation. Generally, most of the roads are categorized into several types:

i. One-lane road that always has one way direction.

ii. Two-lane road: can be one way direction or not (bi-direction).

iii. Three-lane road or more: can be one way direction or not. (Lane direction distribution is provided by OSM)

Firstly, particles are defined to be on the road if they fall within the boundary line of roadside on the rendered map, and then confirmed true if its accumulated lateral distance between left and right boundary line is less than predefined road width. The road details will be the main contribution in determining the RPDF along with the vehicle position initialization. Then, the RPDF, represented as α, for each event is illustrated in Figure 3.7 and defined as:

$$\alpha_A = P(R|A) = m, \tag{3.18}$$

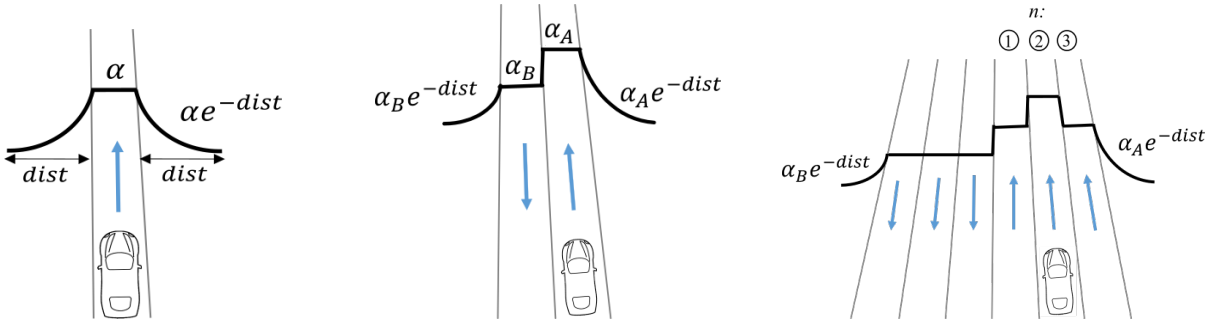
$$\alpha_B = P(R|B) = 1 - m, \tag{3.19}$$

where m is the factor for the particles on the lane with rthe ight direction.The classification of each event is described as:

- event R: particles are on the road,
- event A: particles are on the road with the right direction, and
- event B: particles are on the road but in opposite direction.

If the road has two lanes that are not 'one-way', m value is set to be 0.8 as the probability on the road with correct direction. This means that we do not ignore the manuevering probability during the drive. For multiple lane roads (ie. highway), the RPDF is estimated further to improve the accuracy of being in the correct lane. Suppose L is the total number of lanes with the same direction on the road, for $n = 1, ..., L$, the probability factor for each lane in similar direction is

$$P(R|A) = \sum_{n=1}^{L} P(R|A_n). \tag{3.20}$$



(A) single lane, one-way road (B) double lane, two-way road (C) multiple lane, two-way road

FIGURE 3.7: Road Probability Distribution Factor (RPDF) for different road types

When the particle position x_k^i is on lane n, assuming true lane l_t is known, the lane difference from the true lane l_t is defined as l;

$$l = |l_t - n| + 1, \tag{3.21}$$

and finally the RPDF for each lane with similar direction can be computed as

$$\alpha_{An} = P(R|A_n) = \frac{m^{l+1}}{\sum_{l=0}^{L-1} m^l}. \tag{3.22}$$

While $P(R|B)$ is as defined in Equation 3.19, for multiple lane cases the probability factor $P(R|B_n)$ for each lane n on opposite way will be defined as

$$\alpha_{Bn} = P(R|B_n) = \frac{1-m}{L}. \tag{3.23}$$

During the estimation step, RPDF factor α is multiplied to w_k^i to increase weight of particle that has more likelihood to be on the correct road lane.

$$w_k^i = \alpha w_k^i. \tag{3.24}$$

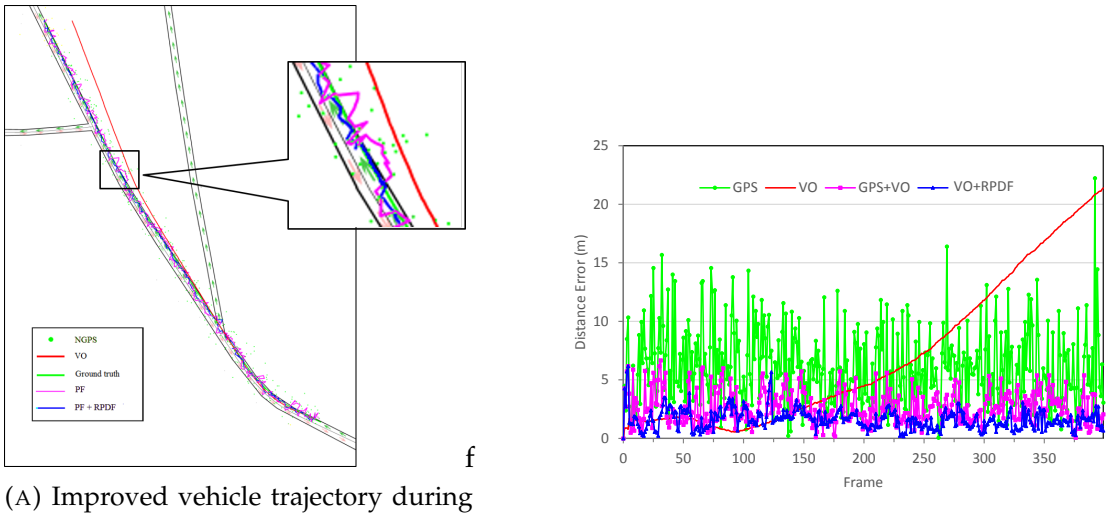
Meanwhile, for particles identified to be outside the road, weights are decreased exponentially as in Equation 3.17 according to its distance from the road as illustrated in Figure 3.7.

3.5.2 Method Validation

Localization performance evaluation was conducted on the same datasets as the previous validation to observe the positioning improvement. For fusion with OSM, the data was extracted beforehand and road networks were filtered where other information other that roads were abandoned. Then, specific useful details of the road were stored such as number

of lanes, road type, allowed directions and speed limit. In the Dataset 2 where it contains multiple lane road, an assumption of initial was made to observe the localization accuracy after the RPDF implementation which includes probability of neighbouring lanes.

3.5.3 Results and Discussion



f

(A) Improved vehicle trajectory during road diversion

(B) Distance error

FIGURE 3.8: Localization results with RPDF for Dataset 1

From the results of Dataset 1 (Figure 3.8b), in overall, VO fusion with RPDF has relatively lower error than the others. However, between frame 123to 126, the result shows error spike for the proposed method. This is due to the existence of road divergence which affected particles weight tendency to be on the other road although the vehicle did not intend to turn. This instant is illustrated in Figure 3.9. As for the distance error, mean and standard deviation obtained for the fusion trajectory result was (2.619 $\pm$ 1.397)m. When RPDF approach was applied to the same sequence, we managed to improve its position error to (1.636 $\pm$ 0.851)m. This is about 38% improvement for the error mean and 39% improvement for the standard deviation. These results are summarized in Table 4.4 together with results achieved in Dataset 2.

TABLE 3.1: Results summary of localization distance error

Dataset	Distance Error ($\mu \pm \sigma$) [m]	
	VO+GPS	VO+RPDF
Dataset 1	2.619 $\pm$ 1.397	1.636 $\pm$ 0.851
Dataset 2	3.608 $\pm$ 1.968	2.522 $\pm$ 1.566

Meanwhile, in Dataset 2, RPDF approach achieved improved localization as shown in the trajectory output in Figure 3.10a and the distance error is improved to (2.522 $\pm$ 1.566)m. Despite the fact that the improvement percentage is lower than sequence with simpler road in Dataset 1 due to RPDF influence from other lanes in the same direction, the outcome is still better than localization results with other methods. Besides, our RPDF effects on multilane road almost achieved lane-level accuracy where it managed to remain on the correct lane except for two occurrences around frame 20 and 40. During these instances, the noisy GPS data were found too far from the true position that made particles on the neighbouring lanes had higher importance weights.

3.6 Synthesis

Our research had proposed visual odometry fusion of GPS, aided with map information and RPDF to improve vehicle global localization. The approach combined a conventional visual odometry system fused with GPS and road information provided by OSM to determine the

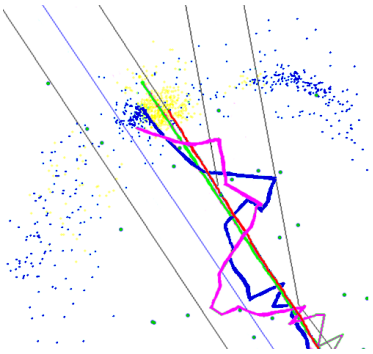
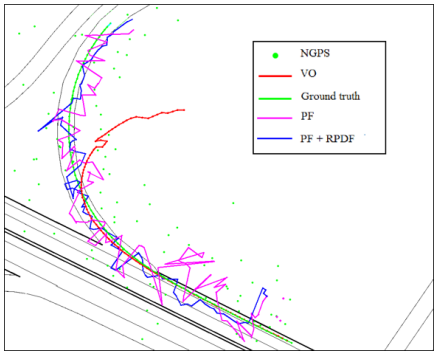
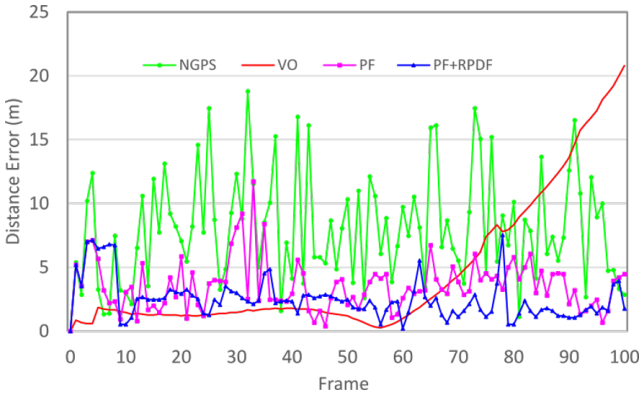


FIGURE 3.9: Error spike during wrong road choice due to GPS influence



(A) Improved vehicle trajectory during road diversion



(B) Distance error

FIGURE 3.10: Localization results with RPDF for Dataset 2

probability factor for each lane and manipulate particles weight to get more accurate localization. This method had been evaluated and compared with the particle filter which showed localization improvement in terms of accuracy and precision from the error analysis. Although road width was predefined by rendering rule which is not guaranteed to be error-free, this can further be improved with the crowdsourcing OSM that constantly provides up-to-date data. Besides, a consistent lane-level accuracy has yet to be achieved due to the influence of severe GPS noise in some instances. Thus, a balance of GPS effect and particle effective sampling size need to be considered to achieve consistency in lane-level localization. Lane-keeping and lane-changing behaviour detection may also be useful in this case to avoid positioning error on the incorrect lane.

The outcome of this fusion technique shows that visual odometry trajectory can be an interesting key input of a data fusion for vehicle localization. This is because despite the drift and scale error of our conventional method, the localization accuracy and precision can be increased with data fusion. With the fusion from OSM map data, the localization is constantly on the road, without requiring map building or road lane detection by image processing. From the error that occurred at road divergence, it might be interesting to utilize the characteristic of visual odometry trajectory to compare with road shape on the digital map. This would avoid discrepancy in vehicle route planning that affect the localization accuracy itself. Besides, the localization accuracy is expected to be further improved if tested in more mature visual odometry methods. In the next chapter, we will explore more on this possibility with longer and different datasets.

Chapter 4

Robust Localization Using VO-OSM Segmented Curve Matching

4.1 Overview

In the previous chapter, the information from OpenStreetMap (OSM) database has been used for lateral error correction by utilizing the predetermined road width to alter particle's weight distribution factor namely RPDF. This chapter proposes an improvement strategy for longitudinal error in vehicle localization. Hence, with the map data provided by OSM, our method exploits this information fused with VO trajectory and low-cost GPS without utilizing additional sensor.

In localization, position difference in lateral axis (left / right of the road) is described as lateral error, ϵ_{lat} and longitudinal error, ϵ_{lon} is the distance difference in longitudinal axis which makes the localization to appear ahead or behind the ground truth data (Figure 4.1). While lateral error can easily be minimized by considering the road width factor, longitudinal position error has been a challenging task in localization especially because most of the time vehicle moves in a forward heading (longitudinal) direction, unless it turns into a junction or curve. It is not less crucial than lateral error, while the latter might affect in precise lane localization, longitudinal error also plays an important factor to determine its accurate position and whether or not the vehicle has passed certain features while moving forward.

Generally, there are two strategies in order to minimize positioning error. Firstly, by drawing a certain limit or boundary as a cutoff area for the most unlikely placement of the state - just like lateral error correction that utilizes road width boundary (Hara et al., 2015;

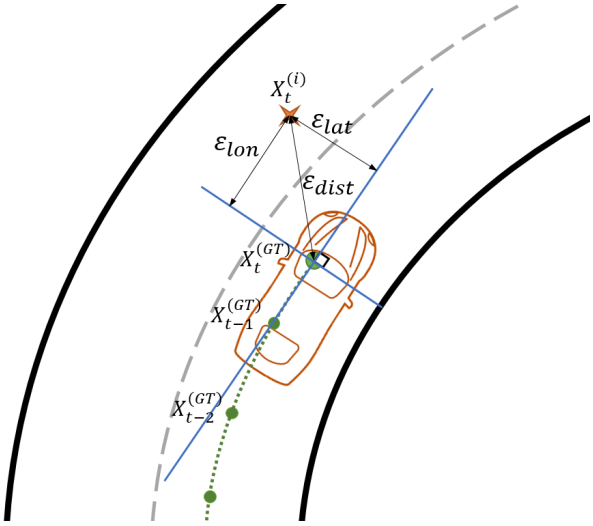


FIGURE 4.1: Localization error

Lu et al., 2014; Tao et al., 2013; Vivacqua et al., 2017); or secondly, by recognizing landmarks and match them into the map to obtain the true current position (Hata et al., 2014; Mattern et al., 2010; Suhr et al., 2017). Approaches have been proposed that suggested using road lane markings, stop markings, curb curve matching and so forth, but there are pros and cons of these methods which we will present in the next section.

In this chapter, a segmented curve matching (SCM) method is proposed that compares VO trajectory trail with road curve segments from the digital map to reduce the longitudinal error and simultaneously increase localization accuracy and precision. The ability of improving localization is studied with common GPS noises. It is a novel approach for curve matching as it does not require a graph-based representation of the map and the curve comparison is simplified into *'node-trios'* (sets of three nodes) instead of integral calculation of the whole curve. Besides, it also does not rely on advance image processing technique to recognize any landmarks or road lanes which makes the computation even easier.

This chapter will explain how we came up with this strategy and its validation, which is organized as follows: Section 4.2 introduces previous approaches in multisensor data fusion that addressed longitudinal error evaluation, followed by Section 4.3 that presents our method in VO-OSM curve similarity matching in localization. Then the method validation will be explained in detail in Section 4.4.1 with the results evaluation and analysis. Finally, Section 4.6 concludes this chapter.

4.2 Related Works

Researchers have conducted studies on localization strategies for longitudinal error compensation. Among the most common approach is by using visual sensors to detect road

markings such as lane marks, stop lines, and symbolic marks. Localization optimization using lane marks detection methods propose detecting and tracking white lines on the road to estimate lateral position of the vehicle and for global localization (Hara et al., 2015; Lu et al., 2014; Tao et al., 2013; Vivacqua et al., 2017), some works performed matching with satellite imagery of digital map (Mattern et al., 2010). Although this method is most widely used for vehicle localization and while it works well for lateral position correction, the information is insufficient for longitudinal localization. This occurs especially when the specific features are not available on digital maps used for matching and obstacles during detection for accurate line matching.

Then, another method is by using stop lines and crosswalks for longitudinal error correction by restraining trajectory drift of inertial sensor (Hata et al., 2014). The researchers only performed image processing during curve lane and stop line detection, thus it can save the computational power. Experimental results showed satisfactory localization performance of submeter accuracy but this method required in-depth road line information from pre-generated map produced by 3D lidar for curve and stop line matching. Hence, this is not feasible for global localization and consumes high cost sensor. Besides, stop line only exists at intersections, thus it only corrects the longitudinal position after some period of time.

For a more cost-effective approach, (Suhr et al., 2017) proposed fusion of low-cost sensors - GPS, IMU, speed sensor, camera - with digital map for vehicle localization by applying symbolic road markings detection and matching with map imagery. It is an interesting approach since although the detection was less frequent than lane line detection, it was enough to generate accurate position without requiring the vehicle to be at intersections for stop line detection. However, this approach is restricted to the availability and clarity of the symbolic road markings (ie. highway roads have less symbolic marking) to be detected, and also satellite imagery could not provide information for matching in cases of the road is out of view (ie. tunnels or multilayer roads). Therefore, although it seems to be the most practical solution for longitudinal error correction, road markings detection methods prove to have a lot of limitations to achieve robust localization in various circumstances.

Multiples researchers have also proposed data fusion techniques for accurate localization without using road markings detection. For instance, (Brubaker et al., 2016) had demonstrated an outstanding performance of map localization in 'kidnapped scenario' (unknown initial position) by only matching visual odometry curve with road network from digital map. Without any GPS data, they generated a probabilisitic map based on the travelled visual odometry trajectory and while the vehicle turns into intersections, the probability of the matched roads increases. Most of the experiments evaluated on KITTI dataset managed to localize the vehicle after a few seconds with average accuracy of 4m. Unfortunately, the system was unable to localize itself in some cases where the sequences were fundamentally ambiguous. Besides, since the information source was solely on visual odometry and digital map, failure in any of these inputs will severely affect localization performance. Hence, GPS

data - albeit noisy - is required especially to provide global position of the vehicle.

Zeng et al., 2016 on the other hand implemented trajectory curve matching between GPS and road curve in map where they introduced curvedness feature constrained map matching method to retain the curveness feature in GPS tracks. This method however mainly focused on curve matching technique in localization and does not present its contribution in reducing positioning error. In addition, GPS data typically contains unpredictable noise depending on the receiver type which makes it impractical for coupling GPS tracks with road map directly for accurate localization.

Therefore, we came up with the idea of assisting GPS, and digital map data fusion with curve matching from VO trajectory for more accurate vehicle localization with the hypothesis that VO curve matching will be able to reduce longitudinal positioning error. This method does not require image processing on road lines, thus is not restricted to out-of-view of vehicle speed limitations. GPS data behaves as a global position reference and limit the curve matching searching boundary. Further details of our proposed approach will be described in the next section.

4.3 System Outline

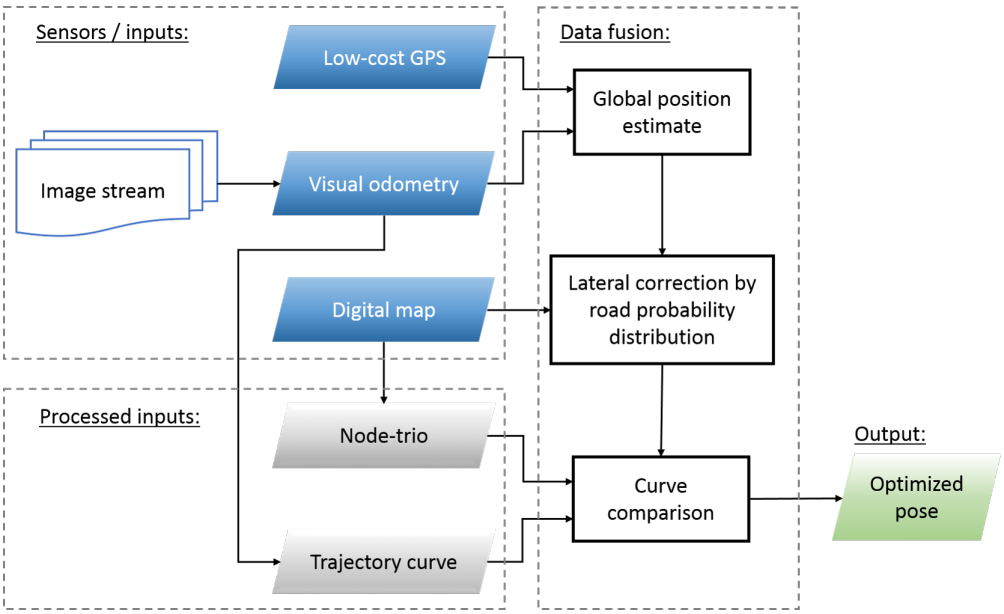


FIGURE 4.2: System overview of proposed approach

The algorithm of our proposed system is illustrated in Figure 4.2. This method mainly consists of three types of input for data fusion; stream of images from camera, noisy GPS and a digital map. Firstly, the images are processed to generate trajectory based on VO that detects and tracks moving features. It is fused with the low-cost GPS data to estimate the

global position. Then, applying our previous approach, road information is extracted from the map mainly for lateral error correction by road probability distribution factor (RPDF). Then, our method is enhanced by utilizing the result of VO and GPS fusion to act as a reference point for road segments curve estimation in the map with weight assigned based on its distance from the centre and similarity with VO curve. From VO trajectory output, a specified length of latest trajectory curve is compared with road segments from the map as a weighing factor for filtering stage during the next step of data fusion. To observe the localization robustness of our proposed method, different types of noise such as random noise, biased noise and an instant where GPS signal is lost are simulated.

4.3.1 VO and OSM Segmentation

In our approach, we utilize the last recent segment of visual odometry to match with road curve in OSM to find the most probable road candidates within search area. Instead of using the road curve line from the map for matching purpose, we simplify the curve matching by using node-trios provided by OSM. As explained in previous chapter, road networks in OSM consist of sets of *nodes* that are connected as *ways*. The distances in between the nodes are not fixed in specified length; the nodes are distanced further in straight roads and closer in curved roads as shown in Figure 4.3 (nodes are represented by yellow dots). In short, the nodes are spaced according to the road's shape linearity. Although at first the distance inconsistency seems to be a problem in point-to-point curve matching with visual odometry trajectory, it is in fact a merit point. This is because our visual odometry trajectory curve snippet is time-based, hence the distance travelled is proportional to the vehicle speed. Since vehicle speed is decreased in curved roads, this will result in shorter trajectory and the shorter node-trios in curve area agrees to this situation and vice versa.

In order to match trajectory curve with map, the knowledge on curve length, orientation, and heading variation is required to optimize longitudinal positioning accuracy. Since OSM nodes are spaced unequally, we need to define the distance threshold between nodes to obtain node-trios with similar length with VO curve for the last m poses, especially in the events where the nodes are too distant on a straight road. This algorithm is illustrated in Figure 4.4 and more details are as follow:

(a) Firstly, the value of m is determined to specify the required VO length. Further details on the determination of m value is described in Section 4.4.1 for validation part. Then, the curve matching is performed on the last m poses, $X_{k-m} \sim X_k$ where the duration of the curve fragment, t relies on VO's frame rate, f.

$$t = \frac{m}{f}. \tag{4.1}$$

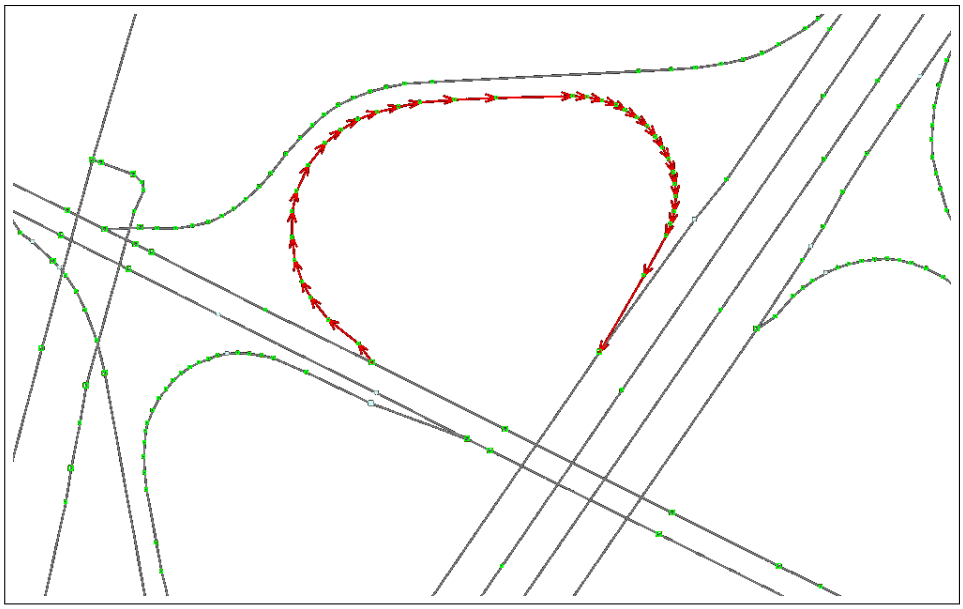


FIGURE 4.3: Node placements in OSM file

(b) Knowing the vehicle average speed $\bar{v}$, the first assumption of distance threshold between nodes d^n_{th} can be calculated as in Equation 4.2. In order to obtain the distance in metre, vehicle speed (km/h) is converted by multiplying $\frac{10^3}{3600}$. Since we utilize sets of three nodes to define OSM curve, the distance threshold between two nodes is divided by 2.

$$d^n_{th} = \frac{1}{2}\bar{v}t \times \frac{10^3}{3600}. \tag{4.2}$$

(c) At the same time, the travelled distance of VO curve, d^{vo} within the last m poses is computed to find the difference with d^n_{th}.

$$d^{vo} = \frac{1}{2}\sum_{i=k-m}^{k} \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}. \tag{4.3}$$

(d) If the difference between d_{th} and d^{vo} is negligible, the larger distance threshold is used to minimize new middle nodes generation. Hence, d_{th} =max(d^n_{th}, d^{vo}). Else, VO trail length is chosen $d_{th} = d^{vo}$ to improve accuracy.

(e) With the first level fusion between low-cost GPS and VO, the vehicle position can be estimated on the map and the nodes around the location are found. Distances between neighbouring two nodes n from the OSM data are recorded as d_n.

(f) Finally, in order to distribute road map probability with longitudinal accuracy optimization, new nodes are replicated in between of nodes that are too distant with each other. The number of new middle nodes, n_m is obtained by rounding results of nodes

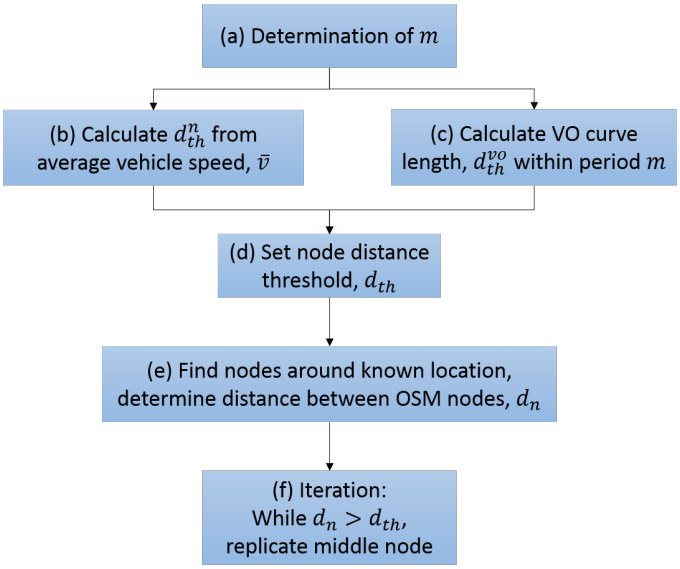


FIGURE 4.4: Algorithm for nodes generation in OSM data

distance division with threshold to the nearest integer:

$$n_m = \left\lfloor \frac{d_n}{d_{th}} + 0.5 \right\rfloor \times d_{th}.$$

4.3.2 Curve Matching

The length of node-trio curve and VO travelled curve distance is not exactly equal all the time, but the difference is small (less than 0.05%) and negligible. To compare the curve similarity between fragment of VO trajectory curve and nodes of roads on OSM, the computation of curve similarity score, $S_{cs}^{(n)}$ relies on two key parameters: initial orientation θ_k^{in} and curve heading variation ϑ_k. Parameters for VO trajectory trail are defined as

$$\theta_k^{in_{VO}} = \theta_{k-m}, \tag{4.4}$$

$$\vartheta_k^{VO} = \sum_{i=k-\frac{m}{2}}^{k} (\theta_{i+1} - \theta_i). \tag{4.5}$$

Three *nodes* curve from OSM data is illustrated in Figure 4.5 and for the n'th node found within searching radius, it forms a node-trio with the neighbouring nodes referred as sets of $[P_n, P_{n+1}, P_{n+2}]$. The initial orientation of the first pair nodes θ_n of the node-trio and curve change ϑ_n are calculated as

$$\theta^{(n)} = \arctan\left(\frac{\Delta y_{n,n+1}}{\Delta x_{n,n+1}}\right), \tag{4.6}$$

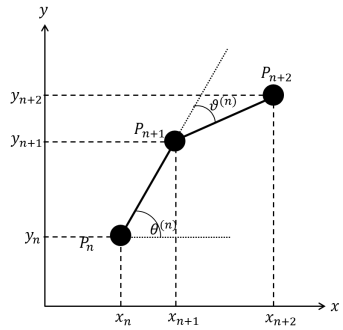


FIGURE 4.5: Angle parameters from three nodes used for curve matching

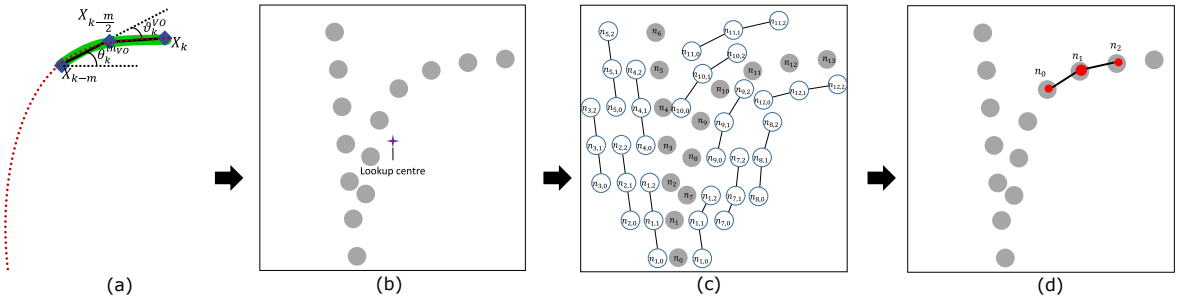


FIGURE 4.6: VO and OSM node-trio curve matching steps

$$\vartheta^{(n)} = \arctan\left(\frac{\Delta y_{n+1,n+2}}{\Delta x_{n+1,n+2}}\right) - \theta^{(n)}. \tag{4.7}$$

With these parameters, curve similarity score $S_{cs}^{(n)}$ of road segment for node n is obtained from the difference of initial orientation θ^{in} between nodes and VO curve and its heading variation ϑ which can be written as

$$S_{cs}^{(n)} = |\theta^{(n)} - \theta_k^{in_{VO}}| \times |\vartheta^{(n)} - \vartheta_k^{VO}|. \tag{4.8}$$

The curve matching steps from VO trajectory curve fragment to surrounding nodes are illustrated in Figure 4.6 where it starts from:

(a) extraction of VO trajectory trail since the last m poses, followed by

(b) connected nodes in ways are detected within searching area from lookup centre as predetermined and explained in method validation (Section 4.4.1);

(c) then, the nodes are paired with neighbouring nodes to form 'node-trios'. All sets found within the area are compared with VO trajectory curve to obtain the most similar road segment to be matched.

(d) Finally, node-trio with highest similarity score is identified and the road segment contains the highest probability that would affect particles' importance weight during data fusion.

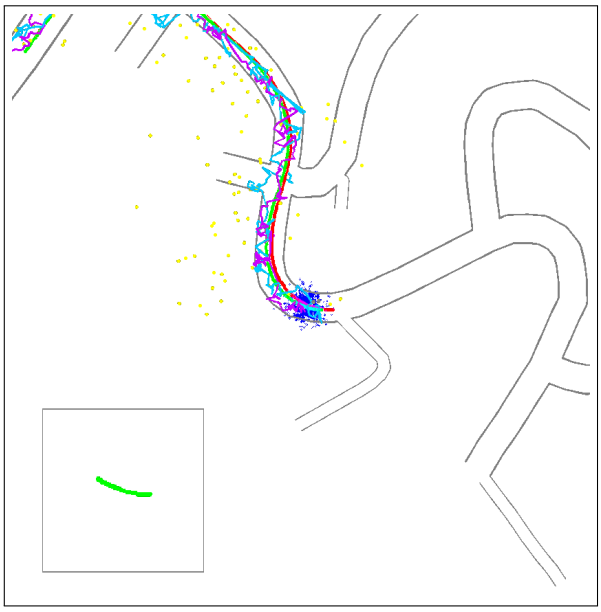


FIGURE 4.7: Double bend road in drive 34 with 2s of VO trajectory in quadratic curve appended in the small box

This approach allows the curve matching to be conducted in small fragments to obtain the most likely candidate of road segment and also reduce computational cost. This is because by assessing only small fragments of trajectory within short travel periods, the resulted curve is limited to 2nd order polynomial regardless of the overall road curve complexity. Generally, only double bend roads (as in Figure 4.7) will result in 3rd order polynomial curve or above. Still, since there is a minimum radius limit for each road curve (Kilinc et al., 2012) and we are only assessing the last small fragment of pose sequence, the trajectory curve will not be long enough to exhibit 3rd order polynomial curve.

4.3.3 Multilevel Data Fusion

Here, we used a cascaded scheme for probabilistic vehicle pose estimation by implementing particle filter (PF) technique to integrate information inputs for fusion. PF uses random samples (particles) to represent the posterior density of vehicle position in a dynamic state estimation framework such as a road network. With PF, we estimate the instate-space models inference by expressing the posterior probability density in terms of randomized particles associated with importance weights.

Our data fusion consists of three levels which are described as illustrated in Figure 4.8 with respective associated data and information used in each level. With the increasing

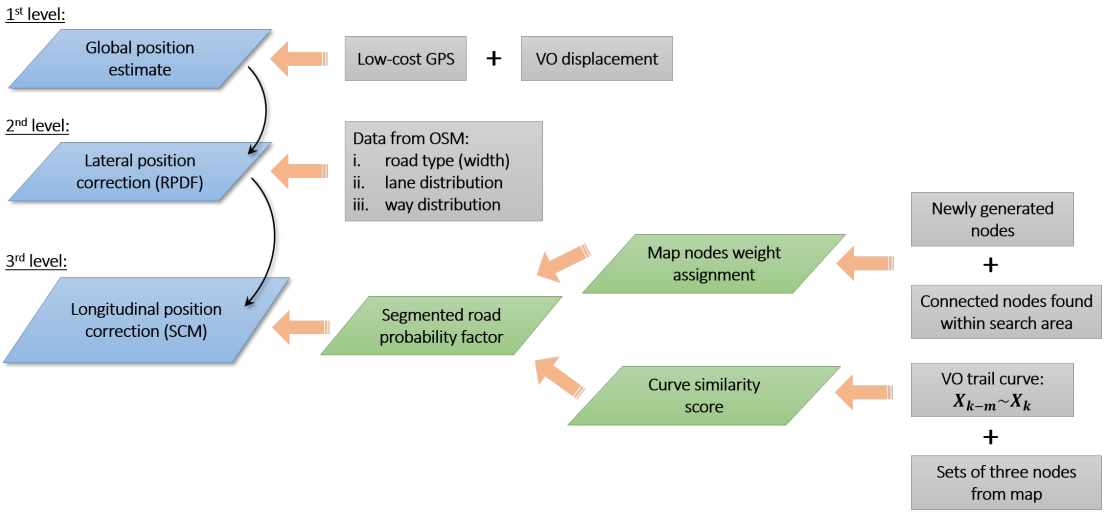


FIGURE 4.8: Multilevel data fusion and associated inputs for each level

fusion level, particles importance weight distribution will be narrowed down according to the road segment that has the most likelihood based on its distance from the first level output and curve similarity. Further details for each fusion level are explained as follows:

1st Level: Global Position Estimate

The first level of fusion between GPS and VO is similar to the method presented in Chapter 3. Through this level, the rough estimation of vehicle initial position on the map is obtained. This time, the study on system robustness is conducted by implementing different types of typical GPS noise that is explained in Section 4.4.1. We also utilized the conventional VO approach as described previously.

2nd Level: Lateral Position Correction by Road Probability Distribution Factor (RPDF)

Similar to our previous approach, we still use the road information obtained from OSM and RPDF to assign particle weights accordingly. The output results of the fusion, $X_k^{(C)}$ is then used as a lookup centre for road segments comparison with VO trajectory in the next level.

3rd Level: Longitudinal Position Correction with Segmented Curve Matching (SCM)

A. *Step 1: Map Nodes Probability Factor Assignment*

From the lookup centre $X_k^{(C)}$, nodes that form candidate ways are searched within specified radius that contains all particles i (explained in validation part of Section 4.4.1). In order to distribute the segmented roads probability factor with longitudinal accuracy optimization, new nodes are generated in between of nodes that are too distant with

each other as explained in 4.3.1 with distance threshold value d_{th} and the number of new middle nodes to be generated n_m.

B. *Step 2: Curve Similarity Score*

To simplify calculation and improve longitudinal accuracy, the trajectory length of VO for the last m poses is set to be short and can only form a second order polynomial curve. We compare the curve segment similarity between VO trajectory curve and node-trios with the computation of curve similarity score, $S_{cs}^{(n)}$ as shown in Equation 4.8.

C. *Step 3: Fusion*

In the last fusion level, probability factor is assigned to each road segments constructed by node-trios for all n'th nodes. This probability factor, denoted as $\gamma_k^{(n)}$, is estimated based on the distance of node n from the lookup centre $X_k^{(C)}$. Then, the curve similarity score $S_{cs}^{(n)}$ of the node-trio is also considered. Higher $S_{cs}^{(n)}$ indicates higher similarity between both curves and this will yield an increased probability factor of the road segment, $\gamma_k^{(n)}$. Assuming d_c is the distance between $X_k^{(C)}$ and node n, $\gamma_k^{(n)}$ is calculated from Gaussian distribution as

$$\gamma_k^{(n)} = \frac{e^{-\frac{1}{2}\left(\frac{d_c}{\sigma S_{cs}}\right)^2}}{\sigma\sqrt{2\pi}}, \tag{4.9}$$

and the importance weight for each particle i is updated as

$$w_k^{(i)''} = w_{k-1}^{(i)} \gamma_k^{(n)}, \tag{4.10}$$

where n is the nearest node found from particle i.

With this approach, we will be able to obtain the best road segments candidates from the node-trio probability especially on curved roads. Although it is difficult to optimize longitudinal error on a straight road without heading variation, our fusion technique will at least limit the longitudinal error from the probability distribution of candidate ways within the lookup range. This is illustrated in Figure 4.9.

D. *Step 4: Resampling*

In order to eliminate particles with low importance weight, resampling is required as the final step in particle filter before repeating the pose estimation steps as above again. The resampling steps are conducted similar to the algorithm as presented in the previous chapter.

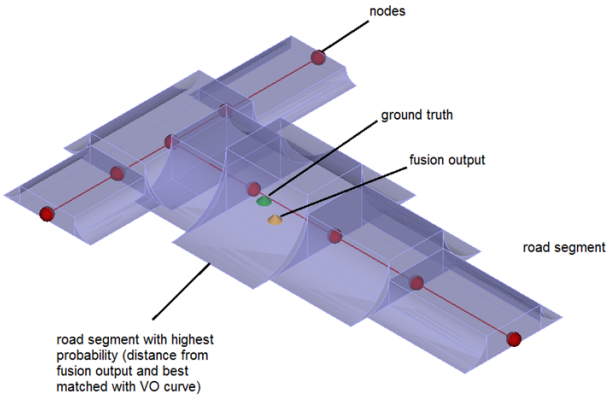


FIGURE 4.9: Probability distribution of candidate ways from fusion output as lookup centre

4.4 Validation and Results Discussion

4.4.1 Method Validation

Our approach with SCM was validated by using KITTI dataset as the benchmark for localization by VO. We tested our method to a highway drive found in sequence 01 and residential drive (sequence 02) for longitudinal error correction while at the same time observe the localization performance in path selection during road divergence. This is especially critical in highway drive where the vehicle speed is faster than those in residential or city area that makes it more challenging to detect accurate path selection simultaneously due to small heading variation. Then, to test the localization robustness, different noise models were applied in both sequences such as an extreme random noise of up to 10m, biased GPS noise and also an event of GPS signal loss.

In this work, we implemented $m = 20$, which means only the last 20 poses (2s of VO trajectory with frame rate 10Hz) was considered as the curve fragment. The curve length will differ according to vehicle travel speed and this corresponds to the node-trios that are distanced based on threshold calculation from vehicle average speed or VO curve. In the case of VO trajectory with severe scale ambiguity, speed can be estimated from the road speed limit provided by OSM data. The pre-set m value was determined based on trade-off consideration between accuracy and computational cost. In order to obtain accurate longitudinal position, the road segments should be as short as possible.

However, if we set $m = 10$ (last 1s of VO trajectory), most of the resulting segments would be mostly found as straight trajectory trail. For instance, assuming the vehicle moves at 30km/h, 1s of displacement is only about 8.3m in length. If it is a curve road, considering the least degree of a significant curve is $10°$, this would result in a minimum radius of 55.6m

for the road. However, according to (Kilinc et al., 2012), the minimum road radius with the speed limit of 30km/h is 60m. Therefore, this is sufficient for the curve analysis where the minimum curve length would be 10.5m. Besides, this would degrade the efficiency of finding the best road segment candidate with the lack of information in vehicle heading change. Hence, we concluded that 2s of travel distance is ample to obtain accurate curve segment for analysis. On the other hand, if the m value is increased to 40, the trajectory curve length is longer and the distance between nodes will increase. As a result, longitudinal positioning would be less accurate with longer road segment containing similar probability distribution.

As for the search area size, candidate ways are searched within a specified radius of about 30m from the lookup centre $X_k^{(C)}$ since we have an approximate location from the noisy GPS of up to 10m accuracy. At the same time, we need to consider the possible bias or blunders, therefore additional 20m as a buffer zone was also considered in this study. Further distance was not considered since it might result in unnecessary road segment probability appearance in ambiguous road network. Besides, expanding the search area to a larger size would only increase computational cost with a greater number of nodes found in the area for probability calculation. Therefore, we fixed it to 30m while observing the effects of lost GPS signal on localization performance.

A. KITTI Dataset

Even though KITTI dataset contains many sequences in different types of environments, only sequence 01 exhibits an event where the vehicle entered road divergence with high speed at a highway exit. Other sequences have other scenarios such as environment with lots of shadows, traffic jam, city roads with lots of other moving vehicles, and roads with pedestrians and cyclists. In our study, we are particularly interested in this sequence. The vehicle was driven on a highway which then exited with the speed of around 60km/h. Besides, it is reported that this sequence had always been the one with highest translation error from the works presented in (Cvišić et al., 2015; Engel et al., 2015; Mur-Artal et al., 2017; Qu et al., 2018; Wang et al., 2017; Zhu, 2017) compared with other sequences. In fact, some of the methods were not able to perform localization on this sequence due to the typical argument of being a highway dataset and vehicle moving at higher speed resulted in higher translation error. Therefore, the purpose of our study is to focus on optimizing localization on this specific sequence. At the same time, dataset 02 with a drive recorded in residential area with many intersections was also chosen to observe general improvement of localization accuracy.

B. GPS Noise Modeling

To test the robustness of the proposed system, we modelled three types of noisy GPS as one of the fusion inputs during method validation. To achieve this, random noise was added to the RTK-GPS data which is downsampled to 1~2 Hz of frequency. Albeit GPS signal noise varies based on the receiver device specification and surrounding environment, it generally can be characterized as a combination of flicker noise and white noise (Mao et al., 1999). Thus, we initiated the noise with specific bias and variance with 2D Gaussian distribution that represented the white noise and a coherent Perlin noise was added to simulate the flickering noise behaviour. The noise is generated from equation (4.11) below, assuming that $[x_k^{RTK}, y_k^{RTK}, z_k^{RTK}]^\top$ is the ground truth position, noisy GPS $[x_k^N, y_k^N, z_k^N]^\top$ is defined as

$$\begin{bmatrix} x_k^N \\ y_k^N \\ z_k^N \end{bmatrix} = \begin{bmatrix} x_k^{RTK} \\ y_k^{RTK} \\ z_k^{RTK} \end{bmatrix} + \varepsilon_G \begin{bmatrix} \cos\theta_p \\ \sin\theta_p \\ 0 \end{bmatrix}, \tag{4.11}$$

where $\varepsilon_G \sim N(\mu, \sigma^2)$ is a random Gaussian noise with mean μ and standard deviation σ. This Gaussian noise ε_G acts as the white noise amplitude and θ_p is generated from Perlin noise.

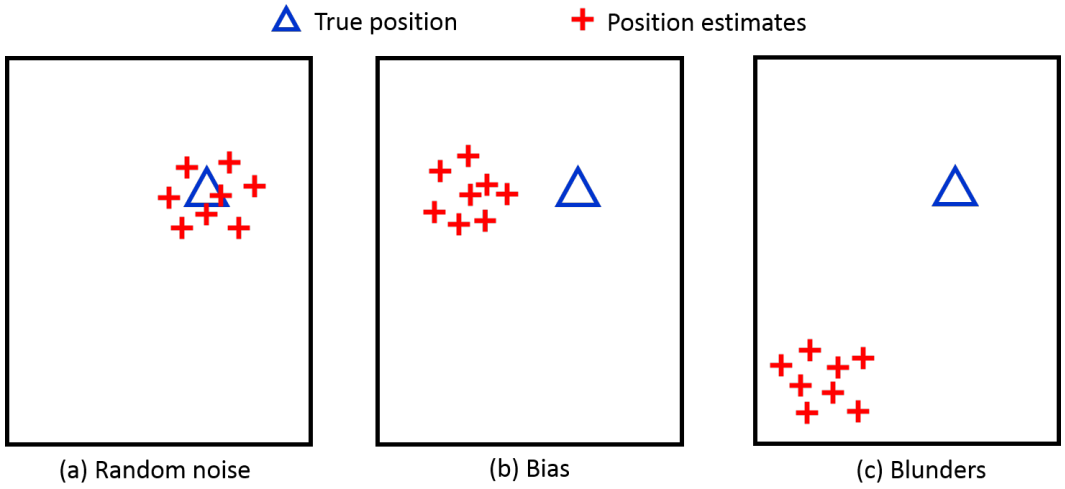


FIGURE 4.10: GPS error sources

GPS noise can be categorized in three main error sources which are: a) noise, b) bias and c) blunders as depicted in Figure 4.10. Noise error is the most common source that is the combined effect of pseudo random noise (PRN) with the noise within the receiver itself. In our validation experiment, this noise error was applied from random noise with zero bias and extreme variance of up to 10m, similar to a low-cost GPS receiver accuracy.

Meanwhile, bias error in GPS can be a result from Selective Availability (SA) or some other factors. SA is the intentional degradation of the standard positioning service (SPS) signals by a time varying bias that introduce delta error resulted from dithering the satellite clock which is a common error to all users and epsilon error which is a varying orbital error. Since SA is a changing bias with low frequency (excess of a few hours), position solutions or individual satellite pseudo-ranges cannot be effectively averaged over periods shorter than a few hours. Other factors that results in bias error include:

(a) ephemeris data errors that contains imprecise orbital data,

(b) tropospheric delays which are affected by the atmospheric pressure, temperature and humidity experienced in the troposphere layer of the atmosphere,

(c) unmodeled ionosphere delays where the ionosphere layer bends the GPS radio signal and changes its speed as it passes through the layer, and

(d) multipath effect caused by reflected signals from surfaces near the receiver that cause interference or be mistaken for the signal that follows the straight line path from the satellite. This source is difficult to detect and avoid.

We simulated the bias noise by shifting the GPS position in global lateral position from the ground truth and add Gaussian noise with the same 10 metres variance. This would result in higher tendency of events where the GPS data falls into the road with wrong direction or outside the road boundary.

Lastly, blunders is also one of GPS error sources where it can result in hundreds of meters of error due to user mistakes, including incorrect geodetic datum selection. This kind of large error can also be resulted from out of date GPS data due to signal unavailability resulted from blockades. To model this noise, the GPS data was turned off for a specific time range which made the localization can only rely on information from VO and map.

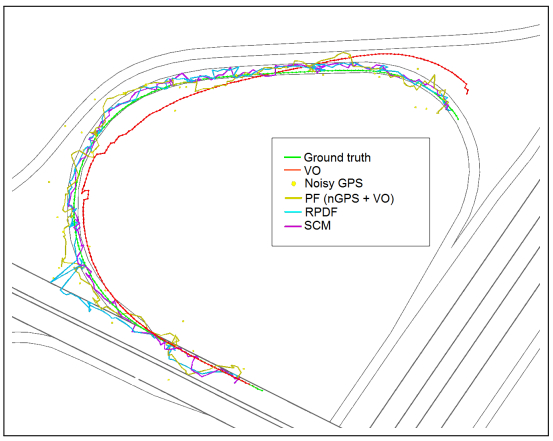
4.4.2 Results and Discussion

Sequence 01: Highway Drive

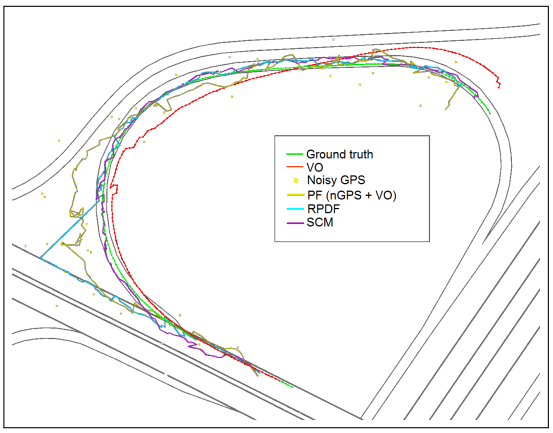
Three different noise models were applied on GPS data to imitate low-cost GPS and tested on a subset of sequence 01 to emphasize on path selection event. The localization compares the performance of GPS and VO fusion, then with RPDF, and finally with the SCM. The trajectory results for each GPS noise are shown in Figure 4.11.

A. *Positioning error results*

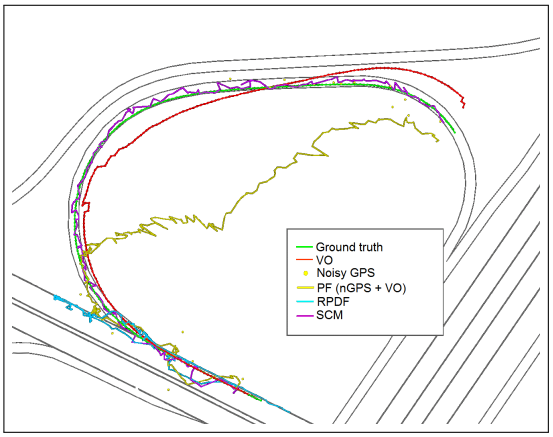
We specifically chose a subset of drive in sequence 01 to observe lane changing behaviour and localization performance from curve matching method. The test was run on 200



(A) GPS noise

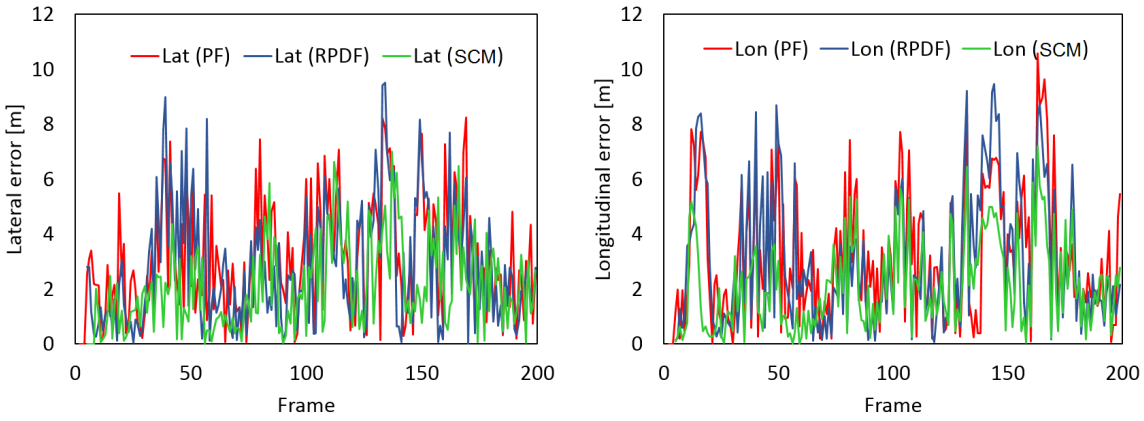


(B) Biased GPS noise

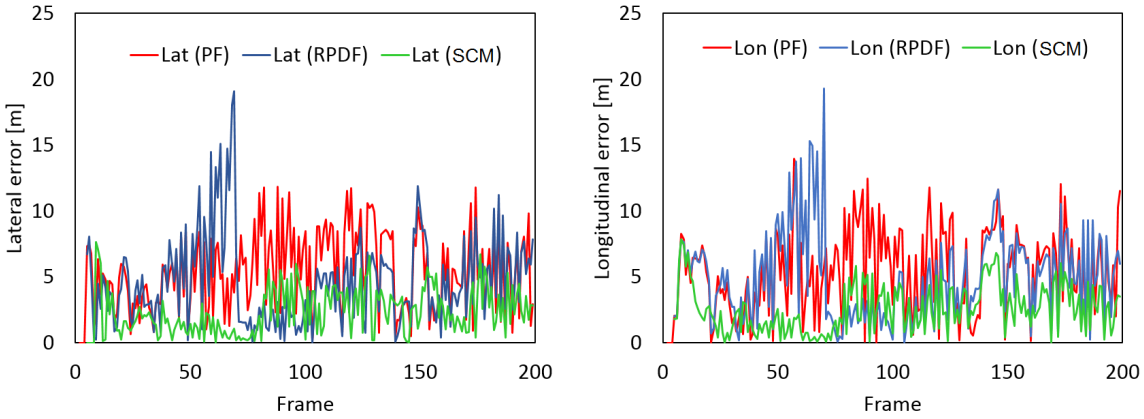


(C) GPS signal loss

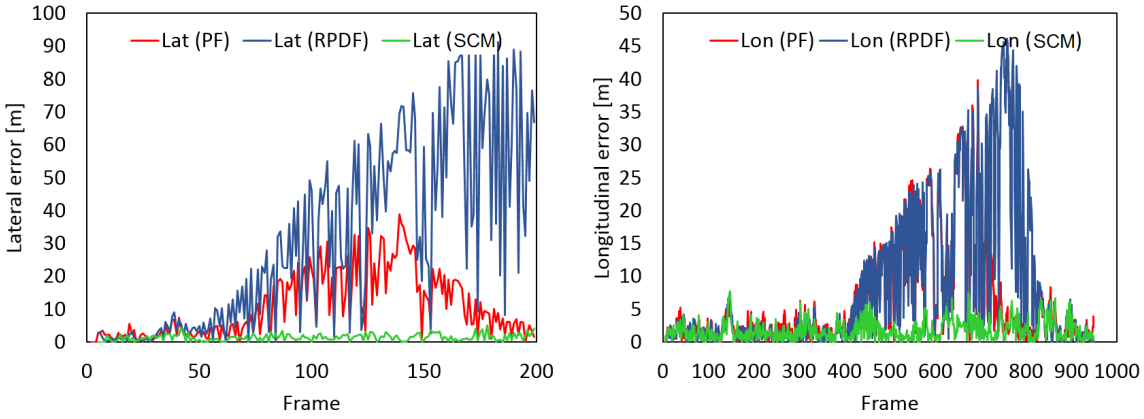
FIGURE 4.11: Trajectory results for highway drive (subset of sequence 01)



(A) GPS with random noise



(B) GPS with biased noise



(C) GPS signal loss

FIGURE 4.12: Lateral and longitudinal localization positioning error in sequence 01

frames with the total distance of 340m and tested with three types of GPS noise. For the highway drive during GPS with random noise (both unbiased and biased cases), it can be seen from Figure 4.12a and Figure 4.12b that the longitudinal error was improved significantly. This took place in the first 70 frames when the vehicle was driven on a straight road and facing road divergence. Regardless of the biased applied to the GPS noise, our method with curve matching proved to be unaffected by the noise. We are also able to select the best candidate road segment that matches with the true trajectory. As for the lateral positioning error, there was a brief improvement in random noise case especially during the road divergence in between frame 45 to 55. During this time, it showed that the lateral error tested in biased noise was significantly improved, as well as its longitudinal error.

Meanwhile, in the case of lost GPS signal, as depicted in Figure 4.12c, our previous approach without road segment curve matching suffered horrendous positioning error. This is as a result of the fact that the data fusion lost one of its important factors and it solely relied on the last GPS position and VO displacement. In fact, the localization failed to relocate its position even after the GPS signal was recovered as can be seen from the results due to the travelled distance was too far from the 'lost' particle cloud. However, with the curve matching approach, it was hardly affected by this problem owing to the curve matching of VO trajectory and road segments that enabled the localization to detect the road segment with the most probability. The longitudinal error result shows that our method managed to maintain the error below 5m for the whole drive in opposed to the error of the previous method that reached up to 20m for both lateral and longitudinal error.

The results of distance and positioning error for the compared approaches is summarized in Figure 4.13 for highway drive sequence in all noise types. In the random noise test, our proposed approach for sequence 01 showed more significant improvement in error reduction of (34.0$\pm$21.2)% in lateral error and (33.7$\pm$ 30.5)% in longitudinal error. This improvement is owing to successful path selection in diverging road. Meanwhile, we managed to achieve further improvement in biased noise where both datasets were able to reduce lateral error of (59.1$\pm$41.9)%, while the longitudinal error is reduced to (54.6$\pm$43.9)%. Finally, for the lost GPS test, the RPDF method suffered worse error than the conventional GPS and VO fusion. However, with road curve matching, we successfully maintained the localization performance with average positioning error that was constantly below 3m.

B. *Path selection response*

As for the path selection accuracy, we evaluated the performance of our SCM based approach according on the moment localization is corrected to the diverging road which is the true trajectory in this dataset compared with RPDF method (Awang Salleh et al.,

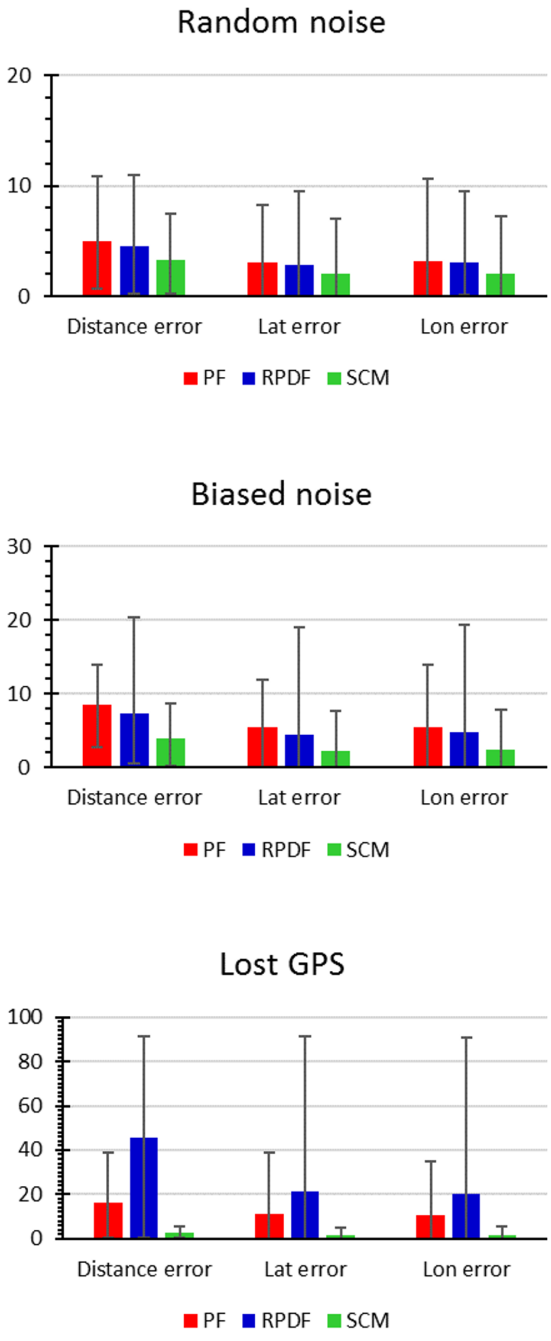


FIGURE 4.13: Error summary for sequence 01

2018b). Since GPS and VO fusion based localization did not include road probability factor for path selection, it is not valid for comparison. In the random GPS test, our curve matching technique manages to localize itself simultaneously (it entered the diverged road in frame 32) with the ground truth while the RPDF method only corrected itself at frame 58. This was even worse when biased noise was applied to the GPS position where it simulated vehicle position to be on the straight road instead. Thus, it affected path selection performance in RPDF approach by only correcting the position at frame 71. On the other hand, the SCM showed an improved performance where it was unaffected by this as can be seen in Figure 4.11b.

Lastly, during lost GPS signal, the proposed method with curve matching outperformed the RPDF approach where the latter approach was unable to update localization without GPS data as reference while being restricted to localize on road area only. Even once the GPS signal was recovered, the particle cloud was too far from the true position due to the high speed. Interestingly, for the GPS and visual odometry fusion, the localization was less severe due to the non-existence of RPDF limitation localization on the road area.

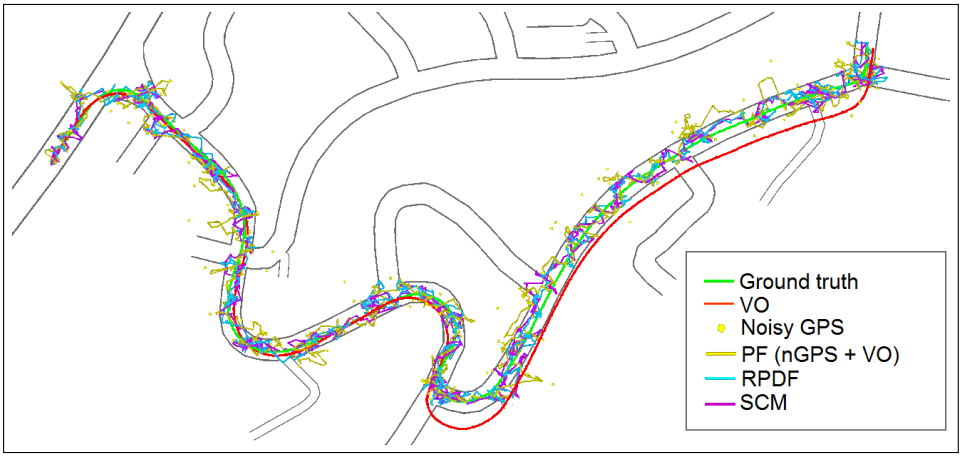
Sequence 02: Residential Drive

Similarly, three different noise models of a low-cost GPS were tested on a subset of sequence 02 which is a residential drive sequence. The localization performance of GPS and VO fusion, then with RPDF, and finally with curve matching method were compared to study the improvement of each methods. The trajectory results for each GPS noise are shown in Figure 4.14.

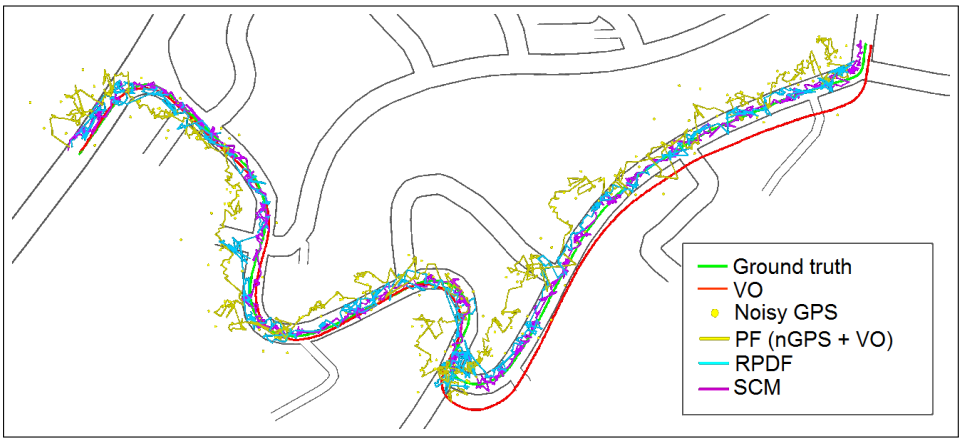
A *Positioning error results*

Our second test was conducted on a residential area for 950 frames with the total travelled distance of 924m and error results are shown in Figure 4.15a, 4.15b, and 4.15c. During the random noise test, it did not show remarkable improvement in positioning error because both RPDF and SCM approaches have the similar basic approach in localization; unless it involves severe GPS error that affects path selection. Since the vehicle speed in residential drive is lower and the intersections are easily distinguished, the localization showed a consistent performance even without curve matching method. However, in overall, our proposed approach still managed to reduce positioning error compared with other methods in some instances where the RPDF longitudinal error peaked up to more than 5m.

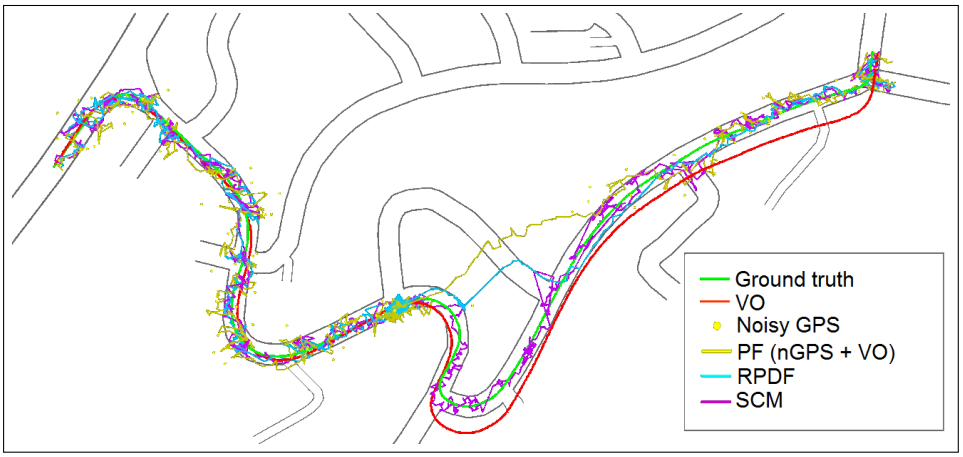
In the biased error test for residential drive, the results as displayed in Figure 4.15b is particularly interesting for the amount of error degradation in the result of method without curve matching in three events. The first event was during frame 70 to 120 when the vehicle moved along a straight road and there was a junction on the right where most GPS



(A) Trajectory during random GPS noise



(B) Trajectory during biased GPS noise



(C) Trajectory during GPS signal loss

FIGURE 4.14: Trajectory results for residential drive

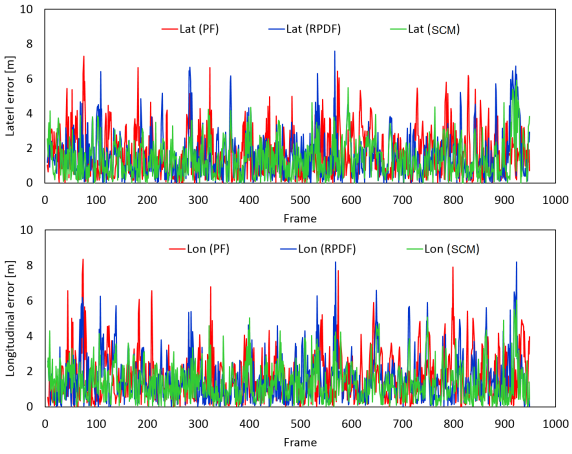
points were positioned due to the bias applied; the RPDF method accidentally assumed that the vehicle turned into the junction. While the second event happened around frame 520 to 590, where the vehicle was moving on a sharp curve that resembled making a U-turn twice, thus resulting the biased GPS fall into the previous lane. As a result, the localization mistakenly concluded that the vehicle had made a U-turn. On the other hand, our proposed approach with curve matching managed to cope with these situations by integrating the VO curve with road curve fragments for all occurrences.

Lastly, we simulated an instance of GPS signal loss to the residential drive and the results obtained are shown in Figure 4.15c. The GPS data was turned off in between frame 400 to 700 when the vehicle was moving along a curved road with multiple intersections on its sides, turning into an intersection and it travelled on a sharp curve resembling a U-turn. As shown in the error graph, it is clear that the recent SCM based approach managed to obtain a consistent low positioning error and unaffected by this lack of data compared with the results of previous fusion method. The RPDF method suffered severe error of over 40m during the signal loss, but it was able to recover the localization after two attempts. This is due to the slower vehicle speed and road curves, thus the travelled distance during GPS signal loss is still reachable by the particle cloud, unlike sequence 01.

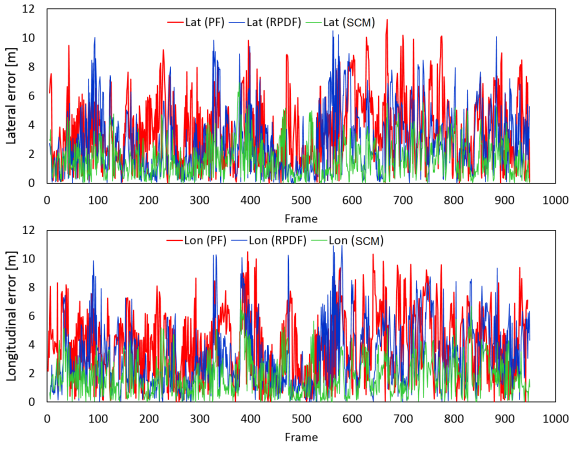
Localization error summary for the residential drive is illustrated in Figure 4.16. The distance, lateral and longitudinal errors show declining trend except when biased noise and signal loss were applied to the GPS. This is where the RPDF approach performance was deteriorated in comparison with the conventional PF (visual odometry and GPS) outcome. However, with curve matching, we were able to reduce the error further as low as below 2m for lateral and longitudinal error in all cases. This lower mean was mainly contributed by the road network itself and vehicle speed that was slower than in highway drive (Awang Salleh et al., 2018a).

B *Path selection response*

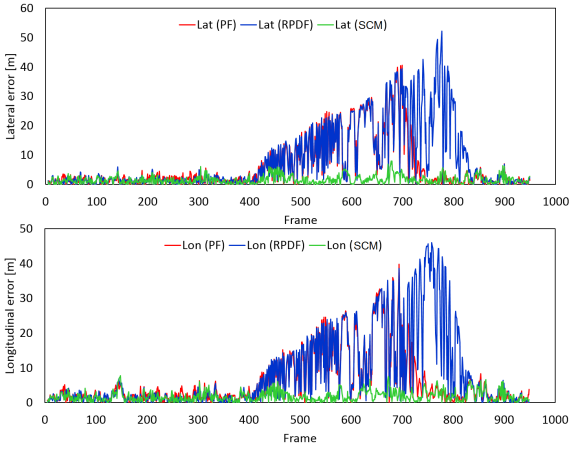
Since the road network for the residential dataset had multiple intersections or curves, and vehicle speed was slower, path selection performance for localization during random GPS noise had less significant difference. However, the positioning error accuracy and precision were slightly improved from the summarized bar graph in Figure 4.16. Nonetheless, our proposed approach showed localization improvement in both biased and lost GPS events. During biased GPS noise, the GPS+VO approach was affected by the bias which showed a shifted trajectory in lateral position. Meanwhile, the RPDF approach outcome had mistakenly selected the wrong road particularly in frame 228 to 252, 520 to 581, and 646 to 666 due to the biased noisy GPS. Lastly, for the lost GPS case, both GPS+VO and RPDF localization results were 'lost' for a moment since frame 400 and finally managed to regain its position in frame 740 for GPS+VO approach and RPDF in



(A) GPS with random noise



(B) GPS with biased noise



(C) GPS signal loss

FIGURE 4.15: Lateral and longitudinal localization positioning error in sequence 02

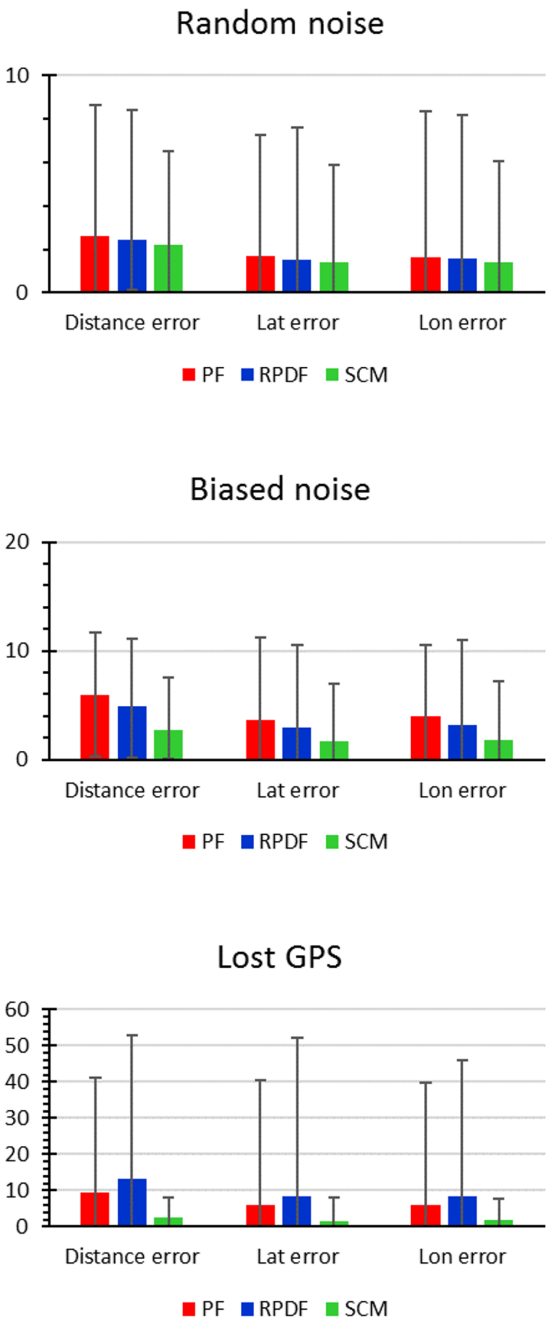


FIGURE 4.16: Error summary for sequence 02

GPS noise	Sequence 01		Sequence 02	
	ϵ_d[m]	Path correction delay (ms)	ϵ_d[m]	Path correction delay (ms)
Random	3.29±1.78	3	2.22±1.07	0
Bias	3.88±1.85	10	3.96±1.91	0
Lost	2.40±1.28	3	3.42±2.01	0

TABLE 4.1: Localization performance with SCM technique

839th frame. From the trajectory we can see that localization by RPDF snapped into nearest roads before locating the correct road. Our proposed approach with SCM however succeeded to perform localization with excellent path selection for the whole drive in all noise conditions.

Summary of the localization performance with curve matching technique for both sequences is displayed in Table 4.1. Here, the distance error and path selection for our proposed approach are presented with different GPS noise types.

Comparison with other VO methods

In order to study the positioning accuracy after data fusion with curve matching, we compared the lateral and longitudinal error of the short trajectory results generated from several other VO methods - Libviso2 (monocular and stereo) (Geiger et al., 2011), and ORB-SLAM2 (Mur-Artal et al., 2017). Although these VO trajectories are without fusion with GPS and OSM data, the idea for this comparison is to find the degree of improvement in localization performance of the published VO methods with conventional VO method with fusion.

The VO trajectories are obtained from the open source VO methods. In Chapter 2 we also mentioned that LSD-SLAM proposed by (Engel et al., 2015) and there is also SSLAM that provides code (Fanfani et al., 2016). However, the provided code is not suitable for implementation with KITTI dataset due to the higher image frame rate required (above 30 fps). Besides, we would like to focus on VO method without mapping or loop closure (SLAM). Note that Libviso2 for both monocular and stereo were purely based on VO, but ORB-SLAM2 also performed local mapping and loop closure detection for drift correction.

Although ORB-SLAM2 also provides localization mode without local mapping and loop closure, its localization for sequence 01 suffers severe rotation error. Therefore, we leave the SLAM mode on and regard this as VO with the best performance in translation error of only 1.15%. ORB-SLAM2 ranked at the 30th place in KITTI evaluation scoreboard, while Libviso2 (S) and Libviso 2 (M) were placed at rank 65 and 87 respectively. This means that we can classify ORB-SLAM2 as a high performance localization, Libviso2 (S) as the localization

TABLE 4.2: Positioning error comparison with other VO works

Method	Sequence 01 (Highway)		Sequence 02 (Residential)	
	ϵ_{lat}[m]	ϵ_{lon} [m]	ϵ_{lat} [m]	ϵ_{lon} [m]
Conventional VO	9.78±15.93	8.92±16.62	5.22±16.47	5.72±17.37
Libviso2(M)	37.24±62.62	23.91±64.44	9.24±41.48	10.34±41.19
Libviso2(S)	6.45±19.93	6.83±19.68	3.09±18.71	3.00±18.7
ORB-SLAM2	1.77±6.66	2.22±6.62	2.31±14.83	2.38±14.81
SCM (random)	1.40±1.06	1.37±1.03	2.05±1.62	2.08±1.58
SCM (bias)	1.66±1.23	1.76±1.32	2.27±1.76	2.45±1.89
SCM (lost GPS)	1.41±1.37	1.30±1.48	1.34±0.93	1.42±1.25

with average performance and Libviso(M) as a low-performance localization. Meanwhile, our conventional VO method achieves localization performance in between Libviso(M) and Libviso(S). The comparison analysis of positioning error for each method with our proposed approach is presented in Table 4.2.

From the comparison table, we can see that our low-performance conventional VO localization accuracy is increased drastically after data fusion. The positioning accuracy is slightly better than ORB-SLAM2 that performs local mapping and loop closure. The fusion results shows over 160% of improvement during sequence 01 while for sequence 2, the improvement level average is around 550%.

4.4.3 Synthesis

A method with data fusion of VO, GPS and digital map for vehicle localization has been proposed in effort to optimize localization accuracy with challenging sequence that has yet to be addressed carefully in this research area. It is a difficult task to ensure the longitudinal position is not affected by GPS noise and VO drift, thus our strategy is to match the last fragment of VO curve with the road segments curve to find the most likely candidate path, namely SCM. The SCM approach reduces the computation complexity while increasing the probability of suitable candidate ways based on its prior curve similarity score. This resulted in a localization system that could withstand various types of GPS noise and VO drift with the overall mean error less than 3m. The longitudinal accuracy and precision were increased in all cases proving that the segmentation concept was able to compensate the longitudinal error. Not only that, it could also increase road selection accuracy immediately for swift path planning that had been an issue during a high speed drive with diverging roads.

TABLE 4.3: Translation error of different VO methods

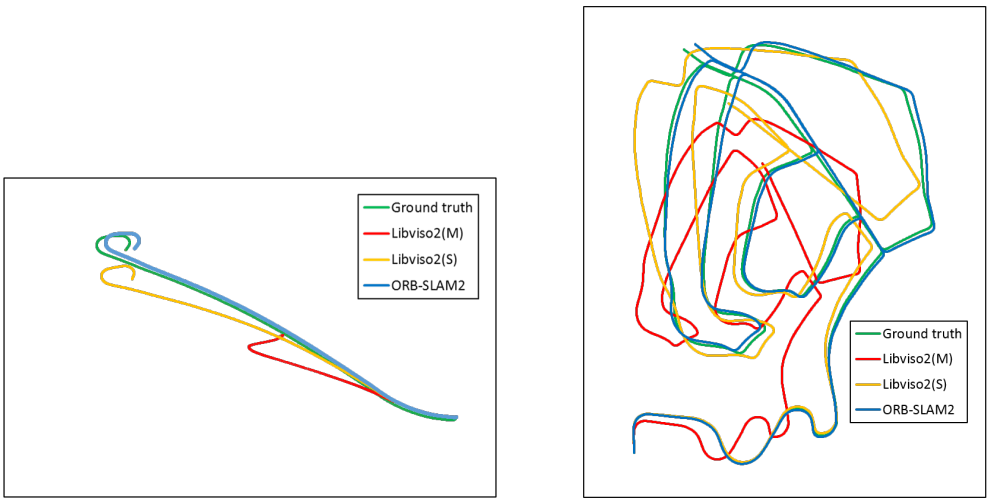
$t_{rel}(\%)$	Libviso2 (M)	Libviso2 (S)	ORB-SLAM2
Seq. 01	25.85	5.00	1.10
Seq. 02	8.13	1.13	0.74

A quantitative analysis on the positioning accuracy in comparison with other localization techniques were also presented. Our method achieved accuracy as good as ORB-SLAM2 without requiring map building and loop detection. In fact, our localization performance is proved to be better than ORB-SLAM2 except during severe GPS noise with bias, where our mean longitudinal error was 3% worse than ORB-SLAM2's, although this is considered as really small. Therefore, we expect our method can further improve the existing VO localization accuracy, despite the accumulated drift resulted in middle or low performance VO.

4.5 Implementation with Recent VO Methods

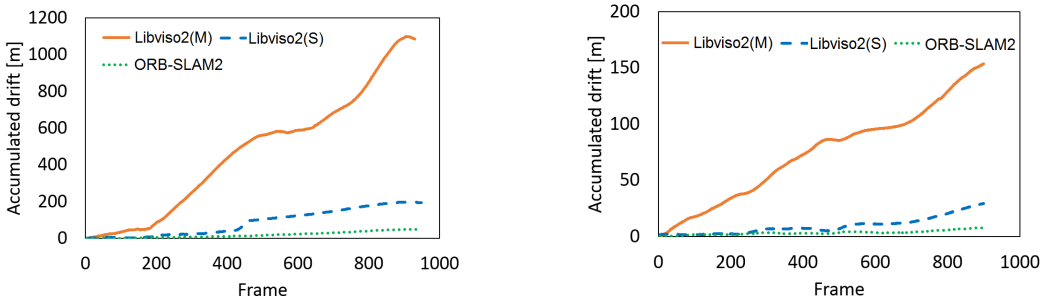
With the developing VO methods presented by other researchers, we are interested to know the effect of employing VO trajectory with different performance on our fusion technique. By doing this, we are able to tell how far our system can improve the localization error despite trajectory drift and scale ambiguation. Therefore, we utilized methods such as Libviso2 (monocular and stereo) (Geiger et al., 2011) and ORB-SLAM2 (Mur-Artal et al., 2017) as the VO trajectory input as low-performance, average performance and high performance visual-based localization. The system performance is evaluated from the lateral and longitudinal error of both datasets, for all different GPS noise types. This time, both sequences were tested in a longer period (1000 frames). This added a challenge in longitudinal error optimization in sequence 01 due to the long stretch of straight road during the drive.

As a reference, Table 4.3 shows the relative translation error t_{rel} of each VO methods for sequence 01 and 02. Obviously, monocular VO by Libviso2 had the worst translation error for both sequences while ORB-SLAM2 performed best in sequence 02. The severity of monocular VO is also displayed from overall trajectory comparison shown in Figure 4.17. Localization drift by time illustrated in Figure 4.18 shows the severe error in monocular VO mainly due to the scale inaccuracy during a high-speed drive on a highway road. Detail positioning results are summarized in Table 4.4.



(A) (B)

FIGURE 4.17: Trajectory of different localization methods for (a) sequence 01 and (b) sequence 02



(A) (B)

FIGURE 4.18: Trajectory accumulated drift for (a) sequence 01 and (b) sequence 02

4.5.1 Validation Results

From the results, we found that the positioning error for Libviso2(S) and ORB-SLAM2 achieved almost similar performance despite the differences in relative translation error of each VO trajectory without fusion with GPS and map. Even for Libviso2(M), it performed well in sequence 02 where the average positioning error could achieve as low as 1.7m during random GPS noise test. However, in sequence 01, our fusion method could not improve the localization in Libviso2(M) due to the severe drift and scale error where it resulted in earlier highway exit as shown in Figure 4.19.

TABLE 4.4: Comparison of positioning accuracy

			Libviso2 (M)		**Libviso2 (S)**		**ORB-SLAM2**	
			ϵ_{lat}	ϵ_{lon}	ϵ_{lat}	ϵ_{lon}	ϵ_{lat}	ϵ_{lon}
Random GPS	Seq 01	Mean, μ	6.78	23.30	1.54	2.29	**0.78**	**1.87**
		Stddev, σ	20.43	21.18	2.77	5.38	0.59	1.36
	Seq 02	Mean, μ	**0.95**	2.16	1.06	1.79	1.02	**1.47**
		Stddev, σ	0.72	1.24	0.74	1.06	0.71	0.98
Biased GPS	Seq 01	Mean, μ	28.31	143.59	1.81	6.83	**0.79**	**4.80**
		Stddev, σ	75.42	94.79	3.07	4.83	0.60	1.94
	Seq 02	Mean, μ	1.52	4.01	**1.35**	3.81	1.43	**2.83**
		Stddev, σ	1.63	2.94	1.00	2.07	1.03	1.52
Lost GPS	Seq 01	Mean, μ	24.17	96.40	1.17	2.63	**0.78**	**1.75**
		Stddev, σ	64.54	74.73	1.30	2.47	0.61	1.18
	Seq 02	Mean, μ	1.30	3.22	1.14	2.05	**1.07**	**1.50**
		Stddev, σ	1.24	2.31	0.68	1.45	0.77	0.91

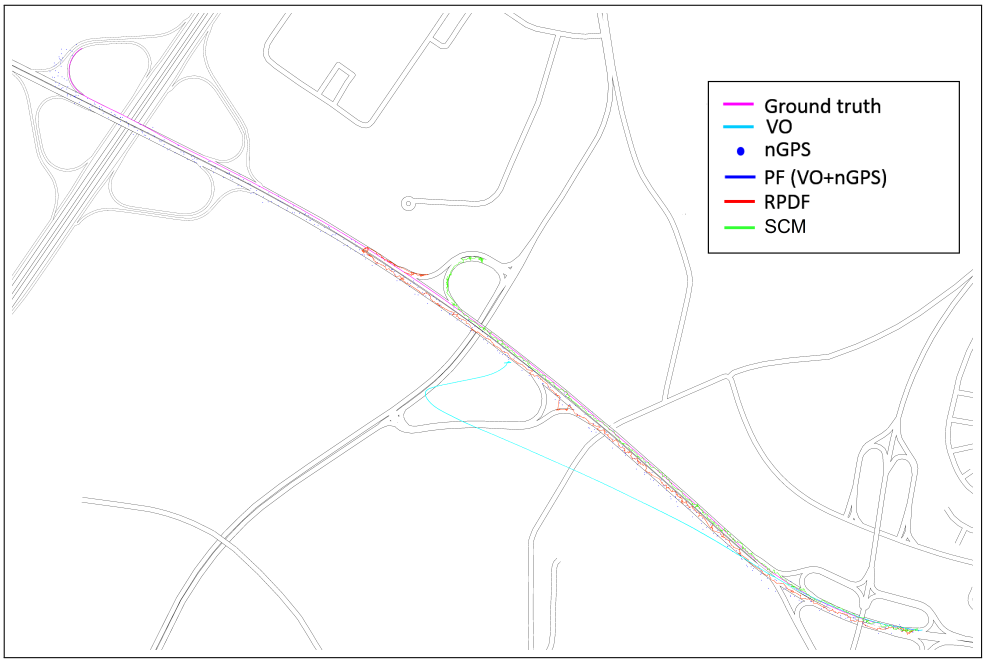


FIGURE 4.19: Severe translation error in Libviso2(M) trajectory resulted in wrong path chosen during fusion

The localization performance for ORB-SLAM2 when applied to our fusion technique with SCM undoubtedly showed the best accuracy especially with lateral error less than 1m in sequence 01 and longitudinal error achieved as low as 1.47m in sequence 02. Meanwhile,

Libviso2(S) showed impressive performance where its lateral error was constantly less than 2m and its minimum longitudinal error was 1.79m in sequence 02 during random GPS error in sequence 02. On the other hand, Libviso2(M)'s best performance was on its lateral error of 0.95m and 2.16m of longitudinal error for sequence 02, also during random GPS error.

As for the comparison in between localization results of different GPS noise models, random GPS noise and lost GPS noise showed little difference in the best positioning error recorded. This means that our fusion technique is able to compensate the absence of lost GPS data by estimating the road segments probability and limits the positioning error with particle filter. However, this is not the case for biased noise where the lowest longitudinal error was 4.80m for sequence 01 and 2.83m for sequence 02 which was achieved by ORB-SLAM2 method as trajectory input. This is due to the effect of fusion with extreme GPS bias error that resulted in unsuitable probability distribution on the road segments. In sequence 01, since the drive is mostly on a straight road with high speed, bias could easily affect the longitudinal error. However, this can be compensated when the heading rotation changed, which will correct the trajectory curve with the right node-trio match. This is the reason sequence 02 achieved better longitudinal error, because the drive itself contained multiple turning points for curve matching.

4.5.2 Synthesis

A method with data fusion of VO, GPS and digital map for vehicle localization has been proposed to reduce vehicle positioning error for both lateral and longitudinal. We tested our approach by comparing several SLAM/VO methods from the publicly available codes with difference performance level. The initial accuracy of each method varies greatly from the monocular Libviso2 that ranked #84 (to this date) on KITTI leaderboard score, and ORB-SLAM2 that has the highest ranking in localization without lidar usage. The robustness of our system was tested with extreme GPS noise for random, biased, and lost signal occurrences.

From our positioning error results, it supported the hypothesis where accurate VO input will contribute in better fusion performance as proved by results achieved from ORB-SLAM2. Although the difference is comparatively small with those of Libviso2(S), ORB-SLAM2 showed 100% better accuracy in sequence 01 for random GPS noise while Libviso2(S) achieved better lateral error in sequence 02 during biased GPS. Furthermore, considering ORB-SLAM2 consumed higher computational cost, the localization performance achieved by Libviso2(S) could be regarded as exemplary where its performance improved significantly – almost on par with SLAM method.

Meanwhile, Libviso2(M) showed good localization for sequence 02 where the lateral error average was also constantly below 2m and longitudinal error average was 3.13m. However, its performance in sequence 01 was not satisfying and we think that this is due to the

scale ambiguity problem that is often faced by monocular system. In overall, it can be concluded that our fusion technique is able to improve localization accuracy despite the drift in VO trajectory but up to the limit where the initial translation error is less than 10%.

4.6 Conclusion

This chapter presented the extended approach to our previous method with RPDF fusion. Segmented curve matching (SCM) technique was introduced to compare VO trajectory curve with road shape on digital map. Starting with a conventional VO approach tested on a challenging drive on a highway road, our proposed localization performance by multilevel data fusion achieved high accuracy, on par with localization by ORB-SLAM2 which requires higher computation cost. Besides, our method proved to be robust against different types of GPS noise, where we applied severe noise variance of up to 10m.

Then we performed further investigation on the effect of having various levels of position accuracy from several VO/SLAM methods that were made available to public for evaluation. Based on the results obtained, the localization performance was greatly improved especially on residential drive (sequence 02) regardless of VO drift and scale ambiguity, thanks to the nature of trajectory shape with multiple turns matched with road shape within the search range. This however remains a problem in the case of straight road as resulted from sequence 01 especially for monocular VO. The lack of trajectory curve in addition to bad scaling factor made it impossible to reduce the positioning error. This should be solved with scaling technique designed for monocular system as proposed by other researchers such as presented in (Fanani et al., 2017a; Zhou et al., 2016).

As a conclusion, the key to the achievement of our localization results was the VO trajectory curve that complement the existing digital map road network for curve comparison. As long as the VO trajectory drift and rotation error is not too severe, the localization accuracy can be significantly improved with segmented curve matching. Therefore, it would be interesting to know how far the trajectory generated by VO can be used for accurate low-cost vehicle localization. Our next step is to determine whether the lane-changing behaviour can be observed from this generated trajectory.

Chapter 5

Lane-keeping / Lane-changing Detection for Lane-level Localization

5.1 Overview

Navigation systems nowadays provide lane-level instructions in taking a route for completing a journey. For instance, multi-lane roads may have different destinations for each lane, hence the routing service or navigation application could provide instructions to use the specific lane for the driver. However, vehicle localization itself has yet to provide lane-level position that would complement the service. Thus, lane-level positioning system would be highly desired in outdoor localization technology where it will estimate vehicle's current lane correctly.

Lane-level positioning is not only useful for navigation system, but it is also required in emerging technologies such as advanced driver assistance systems (ADAS) (Hofmann et al., 2009), road traffic estimation in lane-level detail, and electronic toll-collection system. The current positioning system technology can provide sub deci-meter accuracy and while it satisfies the need for ordinary location-based services (Aly et al., 2014; Ye et al., 2010), it lacks information on vehicle's exact lane position. This brings to an interesting field of research where many researchers had proposed different techniques to achieve lane-level accuracy in localization (Cui et al., 2014; Tao et al., 2013; Toledo-Moreo et al., 2010). Unfortunately, these systems need high specification GNSS device such as RTK-GPS or laser sensors that are expensive and not practical for ubiquitous implementation. Meanwhile, low-cost solutions as proposed in (Cui et al., 2014; Du et al., 2016; Rabe et al., 2016) used camera as visual

sensor to detect road and lane markings. However, these methods will face difficulties in case the road markings are faded, or camera's line-of-sight is obstructed.

Besides developing localization system that is able to sense the correct lane of the vehicle, lane-level accuracy can also be achieved if lane-changing or lane-keeping is detected and correctly identified. This information can then be fused with the available road information on lane diffusion from digital map like OpenStreetMap to obtain the lane-level position. In this chapter, we would like to explore the possibility of employing visual odometry trajectory in lane-keeping or lane-changing detection, without utilizing any additional sensor. This study assumes that provided that the visual odometry trajectory has acceptable drift and scale factor, the trajectory curve can provide ample information for the detection.

5.2 Related Works

Recent studies on lane-changing detection can mainly be categorized into three approaches: trajectory prediction model and matching (Houenou et al., 2013; Salvucci, 2004; Woo et al., 2017; Yao et al., 2012), road lane marking interception (Satzoda et al., 2015; Weidl et al., 2015), and vehicle kimatics based characterization (Aly et al., 2015; Chen et al., 2015).

The first approach requires trained lane-changing trajectory model and matched with the vehicle trajectory to observe driver's intention of changing lane. For instance, (Houenou et al., 2013) utilized the Constant Yaw Rate and Acceleration (CYRA) model for trajectory prediction and others trained datasets of possible trajectories from experiments with different drivers. However, the studies are more practical in the early detection of lane-changing to prevent collision with other vehicles and the confirmation on lane-changing was not observed. Besides, since predicting vehicle trajectory also relies on driver's driving habit, it is not a deterministic task to confirm any lane-change that actually took place at a specific time.

Meanwhile, the second technique detects interception or vertical movement of road lane marking to update vehicle's lane position. Again, this is prone to the visibility state of the markings and is challenging in bad weather conditions. This method was also meant for early recognition of manoeuvers and risk assessment. Lastly, in the third approach, detection by characterizing vehicle kinematics technique was proposed by observing the inertial change of the vehicle. The suggested methods used inertial sensor equipped on the vehicle's steering to observe yaw rate and lateral acceleration for lane-changing characterization. In fact, the lane detection techniques as proposed in (Satzoda et al., 2015; Weidl et al., 2015) also applied the same in-vehicle sensor to capture vehicle kinematics aside from lane-marking cross analysis for lane-change detection. The kinematics only method as presented in (Aly et al., 2015; Chen et al., 2015) is an interesting approach because it does not rely on prediction model nor affected by weather constrains.

Therefore, instead of using inertial sensor, our research will observe lane-changing characteristics and detection visually from the trajectory of the drive. Since cameras can also be used for lane-marking detection as suggested by (Satzoda et al., 2015; Weidl et al., 2015), our method with visual odometry curve exploitation could be a complimentary system for them. In this work, we analyse visual odometry trajectory curve by considering lane-changing characteristics to study the prospective of utilizing only visual sensor to determine the completed task of lane-changes.

5.3 System Overview

Our method proposes an integration of visual odometry trajectory curve analysis with road lane distribution from digital map. Previously, we have worked on fusion between GPS, visual odometry and map in which we found that abundant information in digital map should be utilized for lane-changing detection (Awang Salleh et al., 2016). Thus, without requiring trained trajectory model and additional input sensor aside from camera, we analyse the possibility of lane-changing detection by trajectory curve analysis combined with road lane information. Unlike early detection of lane-changing possibility as presented in previous works (Houenou et al., 2013; Salvucci, 2004; Weidl et al., 2015; Woo et al., 2017; Yao et al., 2012), ours is a deterministic task to confirm any lane change event for lane-level localization update that would ensure the vehicle's accurate positioning.

The curve analysis is done by CUSUM chart (Page, 1954), for small change detection in yaw rate, and curve-fitting error of heading angle extracted from visual odometry trajectory points. Lane-change detection is determined from certain conditions described in Section 5.3.1 from the result of the trajectory curve analysis and the performance of both methods are compared. Details for both curve analysis methods will be further explained in Section 5.4.

5.3.1 Lane-changing

Firstly, in order to determine an event of lane changing, it has to meet several criteria based on lane-changing behaviour in normal circumstances. Referring to the illustration in Figure 5.1 and as reported in several studies on trajectory prediction (Houenou et al., 2013; Yao et al., 2012), lane-changing can be described when:

(a) the vehicle movement has a small change in lateral direction (x-axis) within a limited distance d_l, assuming a normal lane-change should shift just one lane in a single occurence;

(b) its heading angle θ_i before and after lane-changing is almost equal (particularly on a straight road) or the yaw rate ψ_i is constantly proportional (curve road):

$$\frac{d\psi_{i_0}}{dt} \propto \frac{d\psi i_1}{dt}, \tag{5.1}$$

(c) it happens within a limited period of time, t_{lc}, and

(d) longitudinal speed is almost unaffected while lateral speed shows abrupt change.

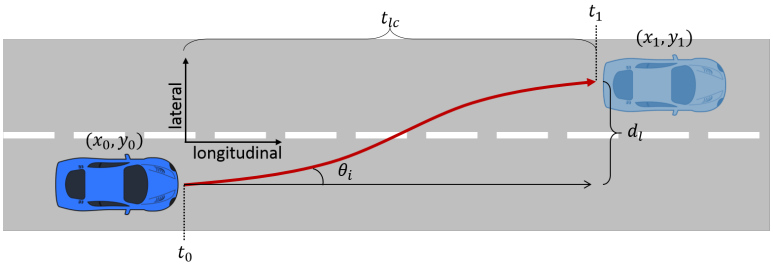


FIGURE 5.1: Lane-change trajectory

Putting these characteristics into consideration, we are able to analyse vehicle lane-changing possibility from its trajectory curve to observe its lateral speed and yaw rate. As for the lane-changing duration, t_{lc}, according to (Lee et al., 2004), the mean duration for lane-changing is around 6.28s for a single lane-change. However, this includes the cases of heavy vehicle (trucks, lorries, busses) that took longer duration for lane changing. Meanwhile, other researchers performed study on standard vehicle lane-change duration outcome that averaged at 4.6s (Toledo et al., 2007). In addition, the minimum average that can describe a lane-change is 1.25s from a study conducted by (Worrall et al., 1970).

The mean differences were as a result of the definition of lane-changing start and stop time, analysis methods, and diversity of vehicle types. In our work, we specify the lane-changing characteristics observation within the given lane-changing duration average of minimum 1.25s and maximum 6.28s. Therefore, we assume the lane-changing duration, (t_{cl}) should be in between 2s to 8s, considering some delay in detection from trajectory curve analysis while avoiding false positive detection from the trajectory noise if the duration is too short.

5.3.2 Visual Odometry

In this study, the existing visual odometry algorithm proposed by (Geiger et al., 2011) for both stereo and monocular vision system was adapted as our visual odometry trajectory input. In their proposed monocular visual odometry, the scale is estimated by assuming that the camera is fixed on the vehicle over a certain height from the ground. Meanwhile, for

stereo vision, the authors demonstrated accurate dense 3d reconstructions from the image sequences to obtain vehicle trajectory.

We also tested on localization trajectory generated from ORB-SLAM2 method as recently presented by (Mur-Artal et al., 2017) that contains three main parallel threads: 1) tracking thread for camera localization by feature matching to the local map and applies motion-only bundle-adjustment to minimize error, 2) local mapping thread, and 3) loop closing thread for drift error correction when loops are detected. This method has 'Simultaneous Localization and Mapping (SLAM)' mode and 'localization' mode. SLAM mode allows map building of an unknown environment and localization of the sensor within the map at the same time by performing loop closure that corrects accumulated error over time. Meanwhile, during localization mode, local mapping and loop closing threads are deactivated and tracked points are matched between ORB in the current frame and 3d points from the previous frame. We purposely tested in both modes to find the variation of yaw rate in different level of visual odometry performance.

In short, the trajectory input that are used in this study greatly varies in terms of positioning accuracy. Monocular Libviso2 has the worst drift error among compared with other methods due to the scale factor limitation, which is followed by Libviso2's stereo visual odometry, ORB-VO (without loop closure), and finally ORB-SLAM2 with loop closure has recorded the highest precision in localization. With this varying visual odometry performances, we would like to analyse the feasibility of using the trajectory curve in determining vehicle lane changing occurrences.

5.3.3 OpenStreetMap

From the OSM file, we extract data of *ways* of roads that provides many information as described in previous chapters, that includes number of lanes and the trafic distribution of each lane. For validation, we tested on 3 sets of drive, namely Drive13, Drive29 and Drive42 where there were several instances of lane changing events in the raw KITTI dataset. The road lane distribution and its specification found in OSM file are described in Figure 5.2.

As depicted in Figure 5.2b, the road has 3 lanes with one-way direction (3-lane highway road). The lane traffic distribution is described in the *tag k='turn:lanes'* where the value contains *'left | through | through'*. This indicates that the most left lane is specifically for turning left, while middle and right lane are for going straight at the traffic light. We can confirm this from the road symbolic marking captured by camera as shown in the subfigure. This is also the case for Drive 42 (Figure 5.2c) where the road is a two-lane one-way road with different directions for left and right lane.

Thus, from the information on number of lanes for the specific road, we are able to determine whether lane-change is possible or not. Besides, it can differentiate between possible lane-changing and maneuver scenario based on the allowable driving direction obtained

(A) Drive 13

(B) Drive 29

(C) Drive 42

FIGURE 5.2: Road segments where lane-changing occurred (highlighted in red) with respective descriptions in OSM file

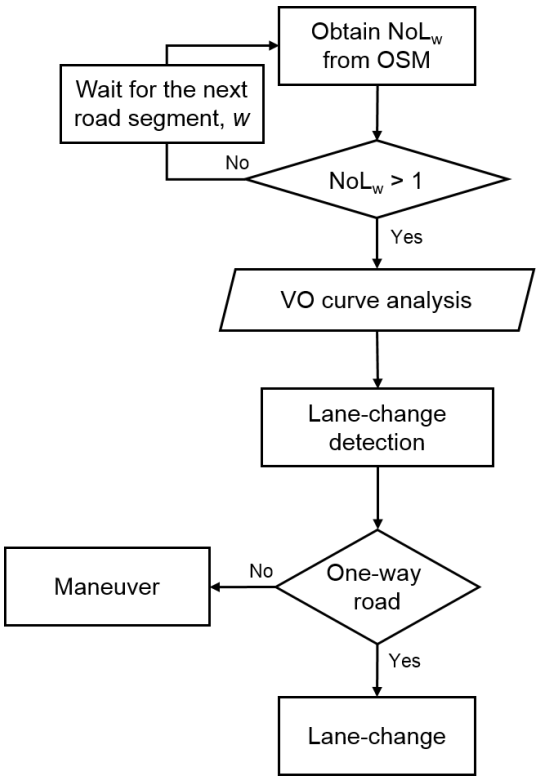


FIGURE 5.3: Lane-change decision flowchart based on lane information from OSM

from OSM data. If the road is multiple lane but with bi-directional, there is a possibility that the vehicle is just taking over the front vehicle and no lane-changing will be recorded since it will return to its initial lane after maneuvering. This estimation is conducted as shown in the flowchart (Figure 5.3) with NoL_w represents number of lane at road segment w.

5.4 Trajectory Analysis

For trajectory curve analysis, we noted that vehicle's vertical acceleration or steering angle is most often used to observe lane-changing event (Aly et al., 2015; Satzoda et al., 2015). Since inertial sensor was adapted in this work, we attempted to extract yaw rate ψ_i calculated from trajectory point at frame i as

$$\psi_i = \frac{\delta\theta_i}{\delta t}, \tag{5.2}$$

where heading angle θ_i is updated from the condition

$$\theta_i = \begin{cases} \vartheta_i + \frac{\pi}{2}, & \text{if } \vartheta_i < 0 \\ \frac{\pi}{2} - \vartheta_i, & \text{otherwise} \end{cases}, \tag{5.3}$$

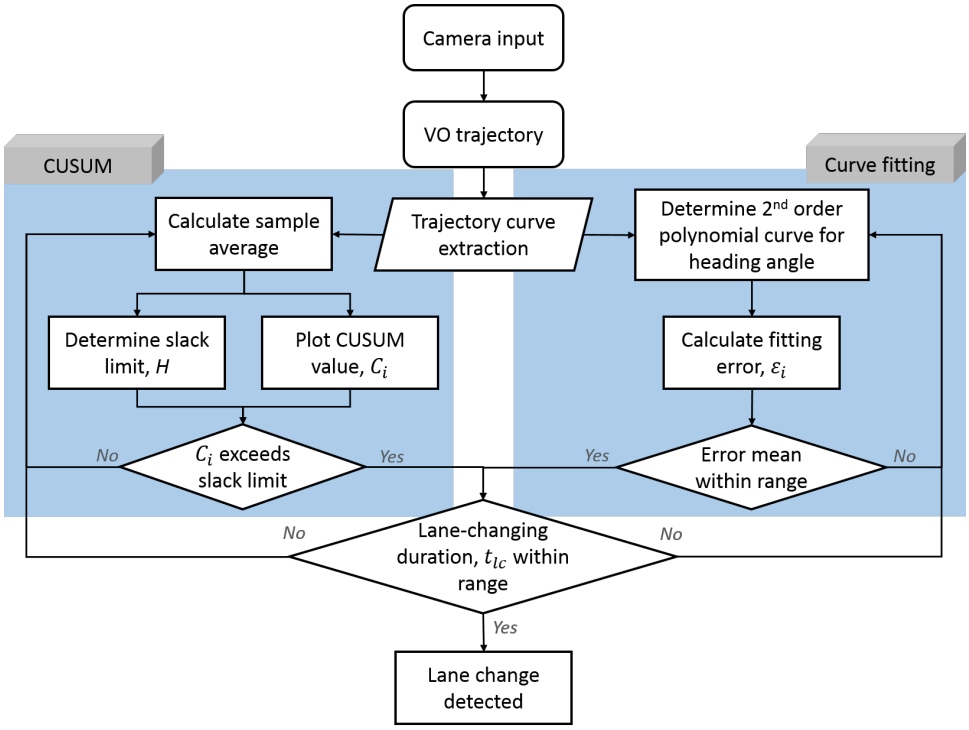


FIGURE 5.4: Framework overview of lane-changing detection

and ϑ_i is the vehicle heading variation obtained from the lateral and horizontal displacement of VO points

$$\vartheta_i = tan^{-1}\left(\frac{\Delta y_{i,i-1}}{\Delta x_{i,i-1}}\right). \tag{5.4}$$

The heading angle and yaw rate will then be utilized for detecting any abrupt change in vehicle heading variation. This change may or may not be observed visually from a raw yaw rate graph, depending on the trajectory noise. Furthermore, a classification technique is required to extract the time and magnitude of lane-changing possibility that occurred within the drive.

In order to detect temporary change in vehicle heading variation during a lane-change, we figured out two approaches that are able to classify a time-based randomized data. As depicted in the framework overview in Figure 5.4, the trajectory analysis is conducted with two approaches. The first method is by CUSUM chart analysis which is useful in detecting small shifts in the mean of a data. As long as we can specify the shift duration and slack limit, this would be a straightforward task in determining the lane-change. The second one is by observing fitting error from curve fitting analysis. Both methods will be presented in detail in the next subsections followed by validation and results.

5.4.1 CUSUM Chart Method

Based on the lane-changing characteristics, it is clear that vehicle orientation change happens in a small magnitude within a short time (Worrall et al., 1970). To detect this behaviour, firstly we proposed CUSUM (cumulative sum) analysis to observe the little change in vehicle heading direction. CUSUM is a type of control chart used to monitor the deviations of individual data or subgroup averages from a target value. These deviations are summed cumulatively over time. A raw CUSUM chart illustrates change in average value of a data differs from target by its slope.

For the decision making, we plotted two 'one-side' CUSUM chart for positive deviations and negative deviations to detect any change that happened above or below respective target. With the decision target plotted on the chart, CUSUM resembles a Shewhart control chart. However, Shewhart chart sometimes failed to detect small shift because it is more effective if the shift magnitude is larger (1.5σ to 2σ).

The CUSUM value above target, C_i^+ and below target CUSUM value, C_i^- can be plotted from Equation 5.5 below. With initial C_i values for both sides are set to 0, the succeeding values are calculated as

$$C_i^+ = max[0, C_{i-1}^+ + \psi_i - \mu_i - k_u], \tag{5.5}$$

$$C_i^- = min[0, C_{i-1}^- + \psi_i - \mu_i + k_l], \tag{5.6}$$

where

- ψ_i : observed data value (yaw rate in radian) for the i^{th} frame,
- μ : mean value for each subgroup,
- k_u : upper allowable slack,
- k_l : lower allowable slack.

The allowable slack k is determined based on sample size n and standard deviation σ as

$$k = \frac{\sigma_i}{2\sqrt{n}}. \tag{5.7}$$

Then, to detect lane-changing behaviour, the slack limit, $H = h\sigma$ is defined and if CUSUM value exceeds this limit, it is considered as lane-change detected. The h value is usually set between 4 to 5 according to (Montgomery, 2009).

5.4.2 Curve-fitting Method

Our second approach is by 2nd order curve fitting method on the vehicle heading angle θ_i, with the assumption that θ_i change is linear on a straight road and curve road can be represented in 2nd degree polynomial curve. For quadratic regression, it has 3 coefficients (a_2, a_1, a_0) that can be obtained by solving the following linear equations by inversion method or Gaussian elimination.

$$\begin{bmatrix} n & \sum x_i & \sum x_i^2 \\ \sum x_i & \sum x_i^2 & \sum x_i^3 \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^4 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum x_i y_i \\ \sum x_i^2 y_i \end{bmatrix}, \tag{5.8}$$

$$f(\theta_i) = a_0 x_i^2 + a_1 x + a_2. \tag{5.9}$$

We choose to fit heading angle curve instead of trajectory curve because it can be plotted on time-based and for its sensitivity in sensing variation. Besides, trajectory curve is susceptible to accumulated drift which may result in inaccurate coefficient calculation results if it is to be matched with the road map where scale and rotation are the main concerns. With curve fitting, the plotted polynomial curve, $f(\theta_i)$ is compared with the actual heading angle data to find any interception within the specified time range based on the error ε_i,

$$\varepsilon_i = \theta_i - f(\theta_i). \tag{5.10}$$

In order to determine lane-changing behaviour, considering there is a small amount of noise in visual odometry trajectory, we assume that:

(a) the fitting error ε_i has an ideal mean error of 0,

(b) ε_i will stay on positive/negative side within a specific time range, t_{lc} during lane-changing,

(c) average magnitude of ε_i within possible lane-changing time range is more than 1°, and

(d) for any adjacent occurence, lane-changing priority to the one with maximum $\bar{\varepsilon}_i$.

5.5 Validation and Results

5.5.1 Datasets

To compute the probability of lane-changing in a drive, we analysed visual odometry trajectory according to lane-changing characteristics as mentioned in Section 5.3.1. Three datasets (Drive 13, 29 and 42) from KITTI were identified to contain lane-changing events. Drive 42

TABLE 5.1: Lane-changing events

Drive	Start frame, i_s	End frame, i_e	Lane-changing direction
13	40	80	right
29	65	90	left
42	40	70	left
	190	235	right

had two lane changing events on a multi-lane highway road, while both of drive 13 and 29 changed lane once to the right and left respectively. The lane changing for each drive happens within the time range as shown in the Table 5.1 with respective direction as observed visually from the image sequences and from the ground truth trajectory. These are also illustrated as shadowed area later in results charts (Figure 5.6, 5.7, 5.8 and 5.9) for clearer view.

For validation, the subgroup mean value μ_i was calculated from on a 10s window. This was defined based on our assumption in Section 5.3.1 where considering detection delay and VO trajectory drift, a complete lane-change should take place within 2s to 8s. Thus, to detect a complete event of lane changing, 2s buffer time was added to the maximum lane-change duration. As for the slack limit, we fixed $h = 4$ to increase detection sensitivity.

Dataset 13 had the shortest drive with 143 frames of images. It had a simple lane-changing event where the vehicle moved from the left lane to the right lane on a two-lane straight road. Meanwhile, drive 29 contained 430 poses where the vehicle moved along a two-lane road that evolved to three-lane road before a traffic light stop. The drive initially started on the left lane (L2 on two-lane) and at frame 65 it moved to the additional left lane (L3 on three-lane road) that was specifically for turning left. The vehicle stopped at the traffic light from pose 200 to 265 before turning left into a single-lane road. The third dataset, drive 42, was taken on a highway road that started with the vehicle driven on the right lane on a two-lane road (L1). Since the right lane was assigned for turning right to a highway exit, but the driver intended to drive straight, at frame 40, the vehicle changed to the left lane (L2) and kept driving on the same lane until frame 195 where it moved back to the right lane (L1) that had the same traffic direction with other lanes. This drive consisted of 500 poses for trajectory analysis and the highway road was slightly curved along the way.

In order to test the feasibility of lane-changing detection by visual odometry curve analysis, these datasets were tested with four different visual odometry approaches - monocular Libviso2, stereo Libviso2, ORB based VO and ORB-SLAM2 - as explained in Section 5.3.2 - these are later indicated as Libviso2(M), Libviso2(S), ORB(VO) and ORB-SLAM2 in the results presentation respectively. Trajectories obtained from each dataset are shown in Figure

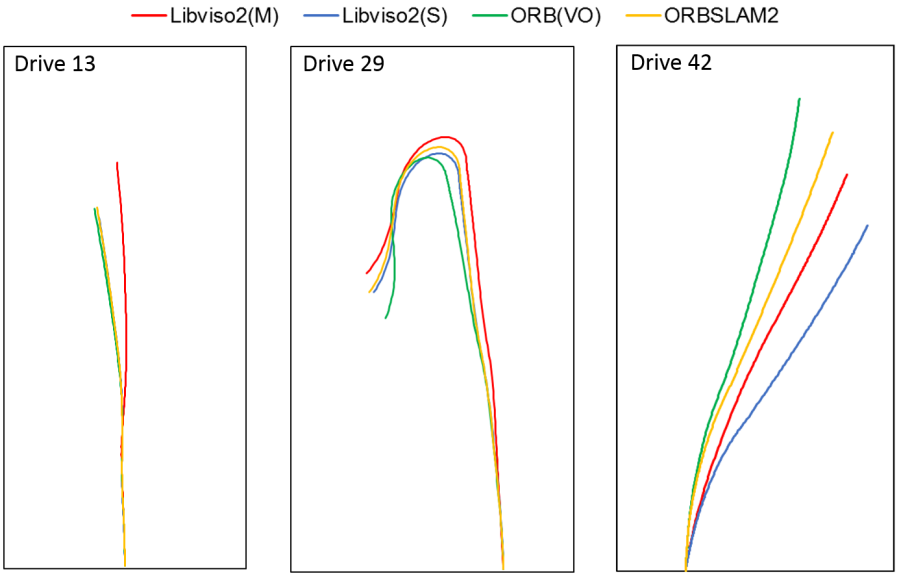


FIGURE 5.5: Vehicle trajectories generated from VO/SLAM methods

5.5 where in drive 13, Libviso2(S) trajectory was overlapped for being too similar to ORB-SLAM2.

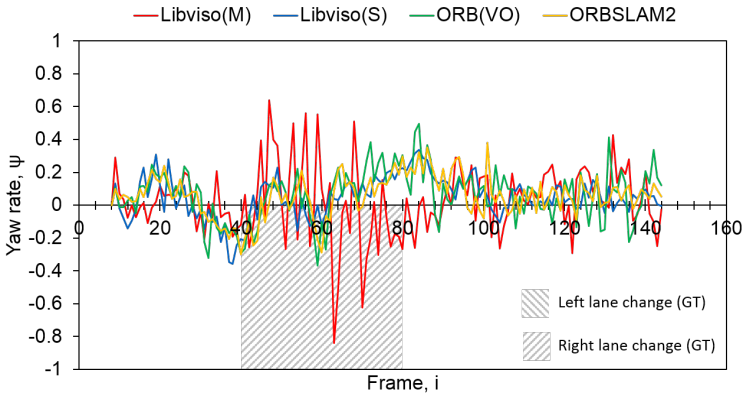
5.5.2 CUSUM Chart Results

For curve analysis by CUSUM, the trajectory was smoothened and noise in yaw rate was reduced by using moving average filter. With window size $N = 2k + 1$, the filtered data can be obtained from:

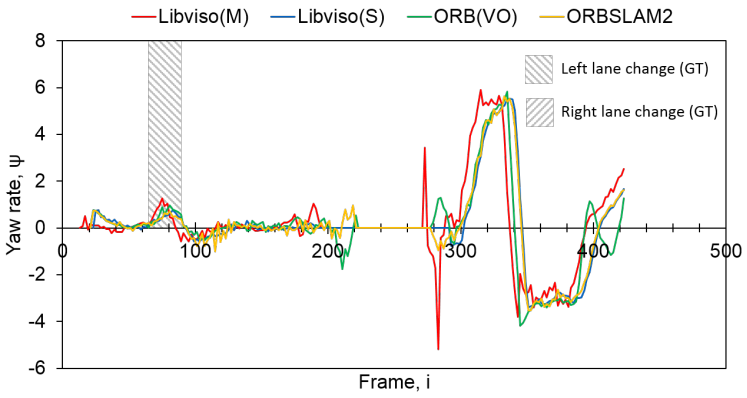
$$F_i = \frac{1}{N} \sum_{j=-k}^{+k} S_{i-j}. \tag{5.11}$$

In the following Figure 5.6, a moving average with window size of 5 points was applied to yaw rate of the raw trajectories. It can be seen that the lane-changing is not easily distinguishable from the yaw rate obtained from visual odometry trajectory curve even after filtering out noise. However, by plotting CUSUM chart, we will be able to detect any small movement change in a random data.

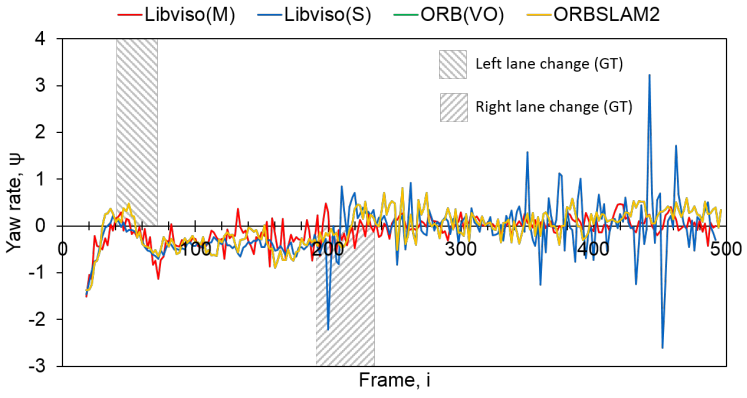
We calculated the CUSUM value C_i in 100 frames cycle and the chart is plotted for each dataset with visual odometry performance comparison as shown in Figure 5.7. Window size of 100 will cause a delay of 10s from the start time of lane-change. However as mentioned earlier, this study aimed to determine a lane-changing behaviour based on visual odometry trajectory after any occurence to confirm the current lane position, thus the delay was not as crucial as for application in risk assessment for lane-changing detection. The straight lane in Figure 5.7 shows the C_i limit determined from $\pm 4\sigma$ and any value that exceeded the limit



(A) Drive 13



(B) Drive 29



(C) Drive 42

FIGURE 5.6: Yaw rate based on VO curve after noise filtering

is considered as lane-change detected. The direction of lane-change was determined by the upper limit for changing to the left lane and lower limit for changing to the right lane.

From Figure 5.7a, Libviso2(S) managed to detect lane-changing between frame 35 to 80 while Libviso(M) detected 2 false positives on the opposite direction (frame 61 and 126). This was due to the severe residual noise in heading angle obtained from visual odometry trajectory that could be seen on the yaw rate graph. Since the scale in Libviso2(M) was estimated by assuming a fixed camera height over ground, any discrepancy would result in scaling error and drift that affected vehicle trajectory points and heading angle calculated from these points individually. Besides, the lane-changing event was in fact followed by gradual change in road direction (slightly curved), which caused the heading angle became more sensitive towards the noise. On the other hand, ORB(VO) and ORB-SLAM2 both successfully detected change in trajectory yaw rate from frame 37 to 77 and frame 39 to 81 respectively. Even though ORB(VO) showed ranging noise after frame 120, the change in its yaw rate did not exceed the slack limit, hence this was not classified as a lane-change.

In drive 29, trajectory analysis from all methods were able to detect lane-changing to the left lane correctly with C_i exceeding upper limit around frame 70 to 90. This dataset exhibited a lane changing to the left lane before traffic light stop and taking a left turn later. Peaks after frame 200 were ignored for lane-changing decision because the road's NoL obtained from OSM data confirmed that it was a single-lane road that made it impossible for a lane-change or maneuver.

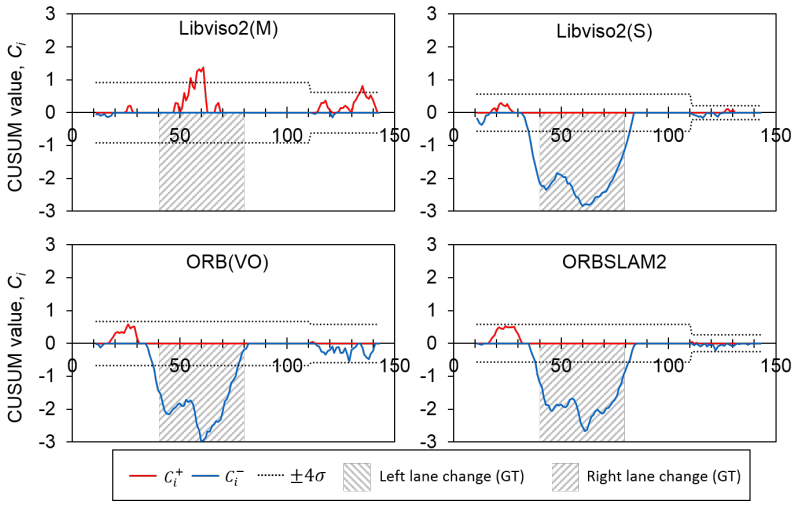
Lane-changing events for drive 42 were also detected by all visual odometry methods. For Libviso(M) result, the exceeding peak was merely detected for the first lane-changing to the left at frame 62 although was not as significant as peaks detected in other visual odometry methods. There were also noise observed as small changes after frame 250 in Libviso(M) but again, since the value did not exceed $\pm 4\sigma$ slack limit, it was not regarded as a lane-change. The second lane-change around frame 200 was detected from the peaks in all methods exceeding the lower CUSUM limit, meaning it was slipping to the right lane.

5.5.3 Curve-fitting Results

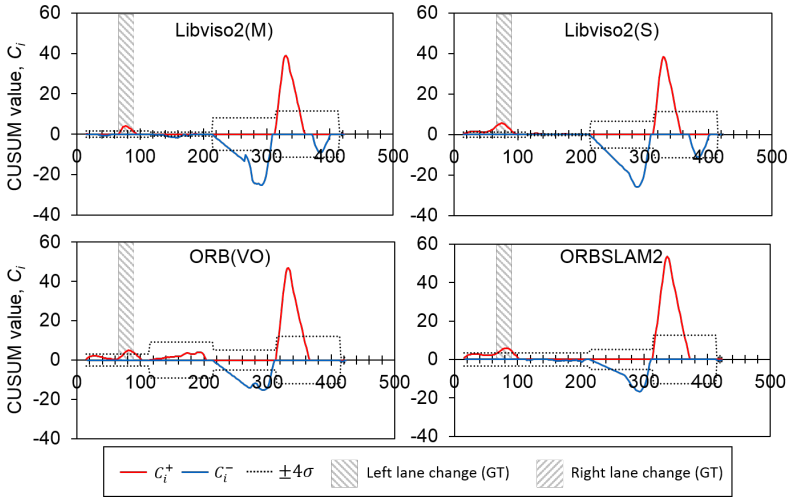
For lane-detection by curve-fitting analysis, heading angle data was utilized and compared with the 2nd order polynomial curve obtained from the quadratic coefficient that fitted the data of 100 window. The curve fitting error graph is shown in Figure 5.8.

Assuming the error should vary around 0 during lane-keeping, if error value ε that remained above/below 0 within t_{lc} of 2 to 8s range and if $1^{\circ} < \bar{\varepsilon}$, it will be regarded as lane-changing. As depicted in Figure 5.9, the lane confidence level in the left (upper side) and right (lower side) lane-change was obtained from the $\bar{\varepsilon}$ value.

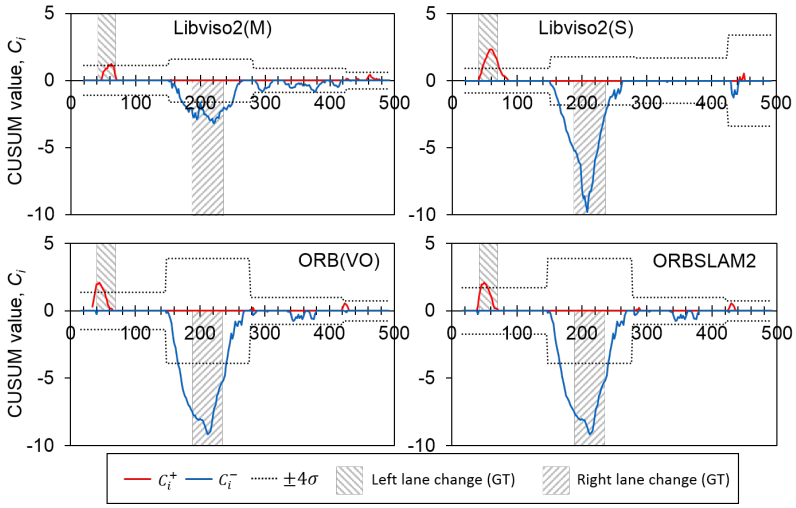
Lane-change to the right was detected in drive 13 from Libviso2(S), ORB(VO), and ORB-SLAM2 trajectory curve analysis where the most accurate lane-changing time was achieved



(A) Drive 13

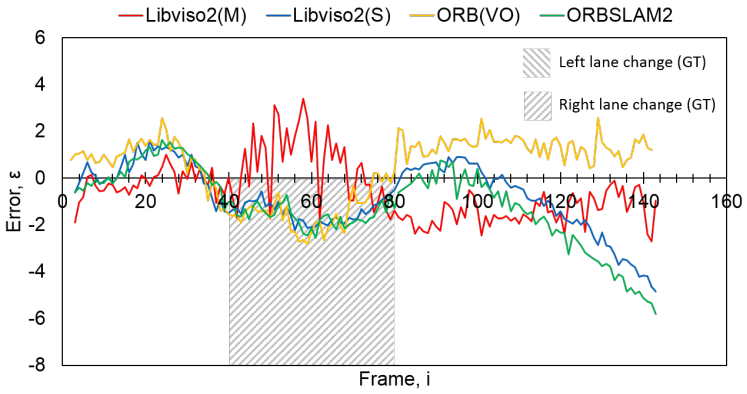


(B) Drive 29

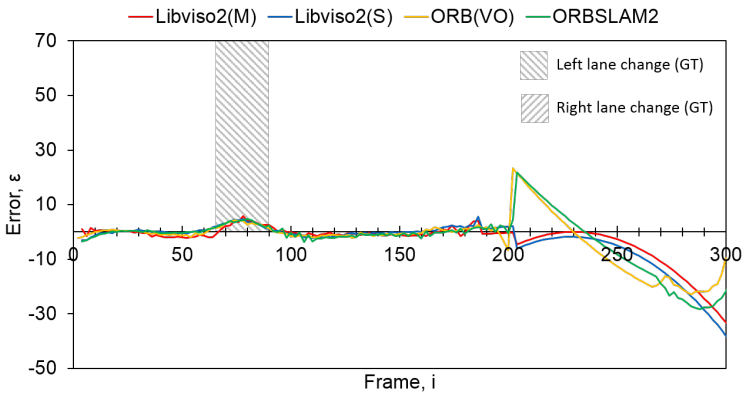


(C) Drive 42

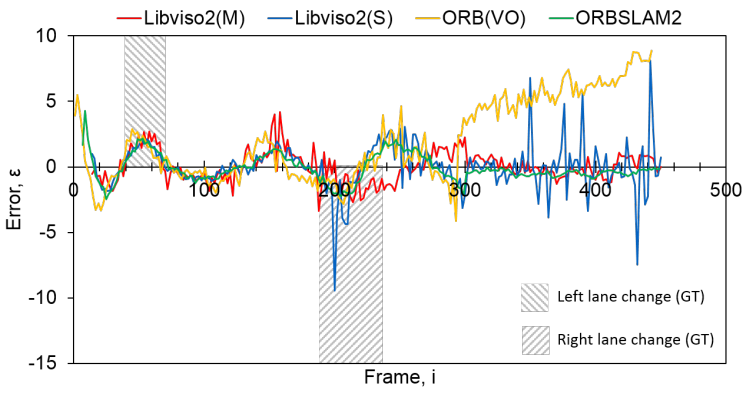
FIGURE 5.7: Lane-changing detection from CUSUM plots



(A) Drive 13

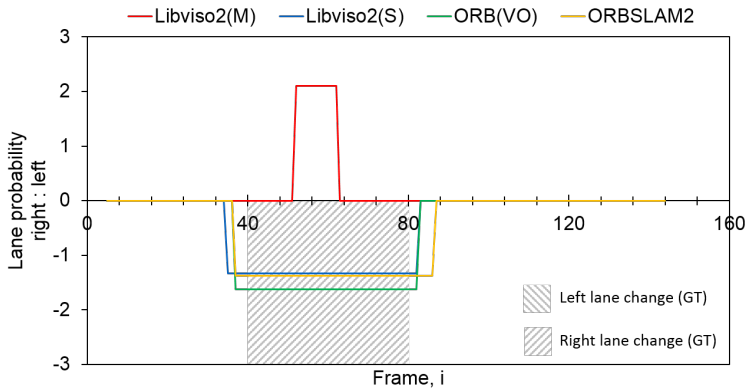


(B) Drive 29

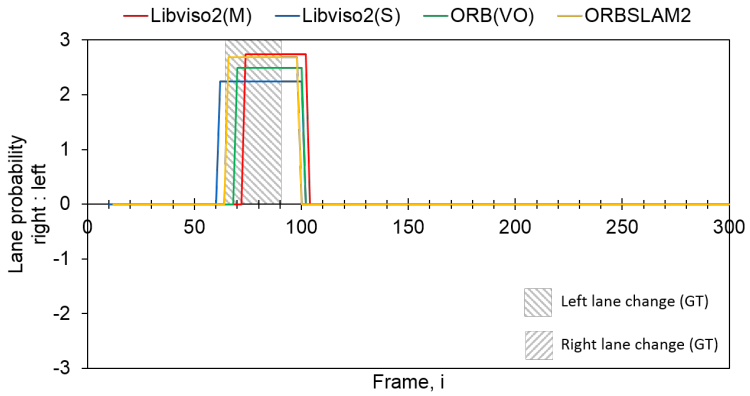


(C) Drive 42

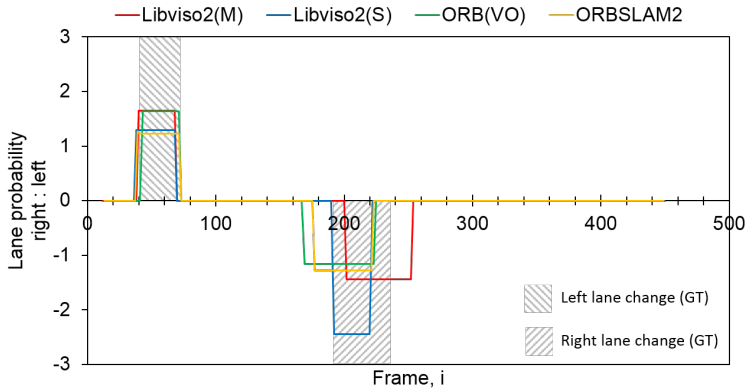
FIGURE 5.8: Heading angle curve fitting error



(A) Drive 13



(B) Drive 29



(C) Drive 42

FIGURE 5.9: Lane-changing detection results by curve-fitting

TABLE 5.2: Overall lane-changing detection results

Drive	CUSUM chart			Curve-fitting		
	TP	FP	FN	TP	FP	FN
13	3	2	1	3	1	1
29	4	0	0	4	0	0
42	8	0	0	8	0	0

by ORB(VO) followed by ORB-SLAM2 and Libviso2(S). Meanwhile, Libviso2(M) once again failed to detect the correct lane-changing due to the residual noise in its trajectory. Our results showed that it almost detected a lane-change in a false positive on the opposite direction around frame 51 to 61, but since the duration was only 1s, which is lower than the 2s minimum average of lane-changing duration, it was not considered as a lane-change.

Next, trajectory analysis of all visual odometry methods were able to detect lane-changing in drive 29 around frame 65 to 100 as seen in Figure 5.9b. Detection from Libviso2(M) was delayed about 7ms from the original lane-changing start time due to the scale drift in the trajectory. However, this was not affected during the first lane-changing event in drive 42 where all odometry methods achieved the most precise detection result at frame 40 to 70. As for the second lane-changing that took place between frame 195 to 235, Libviso2(S) had the most accurate time range followed by ORB-SLAM2, ORB(VO) and Libviso2(M). The latter method showed delay in detection by 1s which was due to the accumulated scale error in monocular system.

The overall results achieved by the different vehicle trajectories and lane-changing detection techniques are represented in Table 5.2. It can be concluded that curve-fitting method has better accuracy in lane-change detection with zero false positive and only one case of false negative in drive 13 from trajectory generated by monocular Libviso2. For this particular drive, CUSUM method unfortunately detected two false positives and one false negative resulted also from the same visual odometry approach. This is a result of the bad scale factor that contributed in severe drift even in a short drive for a monocular system. The short and straight trajectory path made the lane-changing detection by CUSUM more sensitive to any small change, causing incorrect detections. On the other hand, curve-fitting method was less sensitive due to the buffer in fitting error that should exceed 1° to be regarded as lane-change, but this also resulted in detection failure. On top of that, raw data of monocular Libviso2 itself for drive 13 (refer Figure 5.6 and 5.8) has too much noise that was difficult to be filtered due to the window size and trajectory span which made it impossible to detect lane-change event correctly.

5.6 Implementation Challenges

Although it was proved that CUSUM and curve fitting methods were able to detect lane-changing of a vehicle from visual odometry trajectory to a certain extent, real implementation for fusion with OSM data seemed to face several difficulties. Firstly, in order to employ our previously proposed RPDF approach, we need to know the exact lane of vehicle position at least during initialization. Else, with wrong lane decision, this would affect probability distribution of the neighbouring lanes and result in inaccurate lateral position estimation.

Secondly, when there is a change in number of lanes on the road, the OSM data only provides the change in number, hence missing the information of whether the new or missing lane is on the right or left side. Without this information, we are unable to fuse with RPDF correctly. For instance, in drive 42, the drive initially started with two lanes (for RPDF calculation, we have L1 and L2) that became a single lane road (reduced to only L1) after the right lane was diverged as a highway exit. Then, there was a new additional lane on the right again (we have probability distribution for L1 and L2 again), and after some time, a new lane was added also on the right (now we have L1, L2 and L3). Although the vehicle actually stayed on the same lane after it became a single lane, the RPDF estimation would be based on L1 and with a new addition of lane on the right, it would assume of being on L1 (right lane) instead of L2 without the knowledge of on which side is the new lane. The problem worsens with further additional of the third lane that will result in incorrect estimation of probability distribution for each lane.

5.7 Synthesis

Lane-changing detection by visual odometry curve trajectory analysis from CUSUM chart and curve-fitting method was presented in this chapter with comparison on different visual odometry approaches. CUSUM chart detected small change that happened within a certain time range while 2nd order polynomial was fitted on vehicle heading angle data to detect lane-change from fitting error. The analysis were conducted on three datasets with lane-changing events and visual odometry trajectories were obtained from four VO/SLAM trajectories. From the trajectories, heading angle and yaw rate were extracted for curve analysis.

Results on both curve analysis showed consistent output particularly for Libviso(S), ORB(VO) and ORB-SLAM2 approach that achieved 100% successful detection rate. Meanwhile, Libviso(M) failed to detect lane change correctly in drive 13 for both curve analysis methods, although it performed well in the other datasets with some detection delay in curve-fitting method. This is mostly due to monocular visual odometry performance that only has scale factor from assumption and this affects noise in heading angle between each image pairs.

In overall, lane-changing detection by visual odometry trajectory analysis with curve-fitting technique performed better than CUSUM chart analysis technique in terms of detection rate and precision. With this method, we were able to detect lane-changes correctly with start and end time estimation especially in stereo systems. Besides, it does not require parameters settings of k or slack limit 4σ estimation like CUSUM chart does. As for the trajectory performance effects on detection accuracy, visual odometry trajectory with minimal noise are required to acquire reliable heading angle information for more accurate results. Nonetheless, both curve analysis techniques were able to detect most lane-changes despite the overall trajectory drift.

However, in order to apply our method for lane-level localization, we need more information on the road lane distribution especially when there is a change in the number of lanes. Since OSM does not contain detailed information on this, we might have to perform image processing in road lane marking detection to determine the current lane position from time-to-time. This will only be required periodically or when the system received any change in number of lanes, hence minimizing computation effort for the line detection. Therefore, we can conclude that our lane-changing detection approach could be a complementary system with the existing road marking or line detection for an optimized and accurate lane-level vehicle localization.

Chapter 6

General Conclusions and Future Works

6.1 Conclusions

The research work in this thesis was focused on the optimization of vehicle localization accuracy with low-cost sensors. Our main interest was to study how visual odometry (VO) can be used for localization with other available information such as Global Positioning System (GPS) and digital map. While the current most commonly used navigation system is provided by GPS data due to its affordability, users still find inaccuracy in the data which makes it unreliable in certain conditions due to the urban environments or signal outages. This makes an integration with additional sensor input necessary to compensate this problem and improve vehicle localization performance.

Therefore, we proposed to utilize VO trajectory obtained from standard camera to assist localization by fusing with GPS data and publicly available digital map. Cameras can easily be found on vehicles for safety reasons (video recording) and in some high-end cars, cameras perform image processing for pedestrian or lane detection. Aside from these purposes, it can also be used to obtain the vehicle trajectory from the captured image sequences and this has been immensely developed in recent decade. VO trajectory proved to be more accurate than wheel odometry, hence making it an interesting subject to be studied as a fusion input.

Chapter 2 in this thesis provided a wide coverage on literature review of VO techniques with challenges to obtain accurate localization. Besides, we also presented some of the existing works that also utilized VO with other sensors where low-cost sensors fusion techniques have yet to achieve high localization accuracy as performed by fusion with high precision

sensors such as LIDAR. To evaluate the proposed methods of vehicle localization, KITTI dataset was presented and has been widely used as a benchmark and scoreboard. From the published VO/SLAM performance, we can conclude that localization problem mostly lies in the translation error due to scale ambiguity or drift, rather than rotational error. This resulted in better localization on roads with multiple junctions or curved roads while straight roads still face higher localization error.

Our proposed technique aims to solve positioning accuracy in localization including its longitudinal error that remains difficult to overcome particularly on straight roads. This study began with the data fusion of GPS with VO and OpenStreetMap (OSM) with particle filter to verify the effectiveness of the fusion technique in improving localization accuracy as presented in Chapter 3. A conventional VO method was applied and from the information extracted from OSM, a Road Probability Distribution Approach (RPDF) was proposed to enhance lateral position of the vehicle. With the assumption that initial lane is known, and the vehicle does not change lane on a multi-lane road, our technique managed to increase the localization accuracy as tested in two datasets despite the drift and scale error of the conventional VO method. From the error spike occurred at a road divergence, we were interested to utilize the characteristic of VO trajectory curve to compare with road shape on the digital map.

In Chapter 4, we developed our multi-level fusion technique by matching VO trajectory curve segments with road segments on OSM map to avoid discrepancy in road selection and increase longitudinal positioning error. The proposed method called Segmented Curve Matching (SCM), was also tested with low-cost GPS with random error, biased error and an event of GPS signal loss to observe the robustness. Then, advanced open source VO methods with different performance level were used as an input to the fusion and the results show that monocular VO still faced severe localization error in straight roads due to the scale error. However, its localization performance in residential drive was greatly improved, almost on par with fusion output of stereo VO methods for both lateral and longitudinal positioning error. The performance for both stereo VO methods by Libviso2 and ORB-SLAM2 in highway drive with long straight road and higher vehicle speed achieved sub-meter accuracy with longitudinal error reduced to around 2m. This is a significant achievement, considering the low-cost GPS contained severe noise, no high-precision sensor was used, and the vehicle speed was around 60 to 70 km/h.

With the difference in VO methods performance, in Chapter 5, we extended our research scope by attempting to detect lane-changing events from the VO trajectories. Since the detection was conducted on posterior poses of vehicle, it was a deterministic task which is useful for lane update in lane-level localization, to complement our RPDF approach. Two methods were adopted for trajectory curve analysis by using CUSUM chart to detect small change in yaw rate and 2nd order curve fitting of the heading angle. Again, monocular VO failed to detect lane-changing correctly in one of the three datasets that has shorter drive range

while it detected lane-changing events successfully in longer drives. Meanwhile, stereo VO methods managed to detect all lane-changing events correctly in all drives, due to smaller drift and less noise in the trajectories. This concluded that it is possible to determine when a vehicle changes lane or manoeuvers, provided that detailed information on any change of road lane distribution on map is known.

Our proposed method generally fuses low-cost GPS data as a reference for global position with VO trajectory. Then, the multi-level data fusion also integrated information from digital map to obtain probabilistic position of the vehicle on the map. It was a straightforward approach that did not require complex computation and yet it was able to achieve high localization accuracy with an average performance VO method as the fusion input.

6.2 Future works

Despite good overall achievement in localization performance from our fusion technique, there are still room for improvement especially in monocular VO. Due to the scale error, monocular VO suffers from severe drift in its trajectory. This affects the fusion performance where particles are displaced based on VO trajectory vector with randomized noise. Although we also utilize GPS as position reference, the scale error is accumulated by time which worsens the result in road segment candidates. Therefore, it would be desirable to have a reliable scale factor estimation technique in monocular VO systems to, at least, have an accuracy almost equivalent to stereo system with average performance (relative translation error of less than 10%). On top of that, if this can be achieved, we can further reduce the localization system cost with only utilizing a single camera.

In terms of fusion efficiency, noisy GPS data actually affected the overall localization performance. This should be solved by reducing the dependency of particles weight on probability distribution with distance from GPS position or we can also only refer to GPS points at a longer time interval. Since this study modelled GPS noise from a high precision GPS data, it would be more realistic to use a real low-cost GPS data for validation.

In Chapter 5, challenges that we faced for the lane-changing implementation were also mentioned. Indeed, VO trajectory with acceptable accuracy can be used to determine lane-changing events for lane-level positioning system. However, in real life implementation where number of lanes on the road changes frequently, we need further information to identify the current lane position from time-to-time in order to apply RPDF approach. Thus, future work might include lane position identification by image processing from the road line detection to be matched with OSM data. Besides, since OSM is an open source service with a growing community involved in its development, we can propose an additional information in their data to specify detailed information when there is a change in number of lanes. This, would be beneficial for all.

Although this study focused on accuracy improvement in challenging drive sequence, the proposed system should be validated in other sequences and datasets as well. From this, we can further analyze its robustness in different environment and with its overall performance, we can join the leaderboard ranking of public dataset.

Finally, we also look forward to hardware implementation of the localization system which is also included in the research future plan. Upon successful validation on the software end, the real performance can only be evaluated on a hardware platform. For hardware implementation, we can test the system with more options of driving environment with different road network, weather conditions, and urban canyon scenarios. This will be executed once the tasks above are completed and the localization performance is validated with satisfying accuracy.

Bibliography

Abuhadrous, I, F Nashashibi, C Largeau, and M Chinchole (2003). "Multi-sensor data fusion for land vehicle localization using/sup RT/MAPS". In: *IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No.03TH8683)*, pp. 339–344. DOI: 10.1109/IVS.2003.1212933.

Alcantarilla, Pablo Fernández (2011). "Vision based localization: from humanoid robots to visually impaired people". PhD thesis. Electronics (University of Alcala, 2011).

Ali, Farman, Sajid Ullah Khan, Muhammad Zarrar Mahmudi, and Rahmat Ullah (2016). "A comparison of FAST, SURF, Eigen, Harris, and MSER features". In: *International Journal of Computer Engineering and Information Technology* 8 (6): p. 100. ISSN: 2412-8856.

Aly, H., A. Basalamah, and M. Youssef (2014). "Map++: A crowd-sensing system for automatic map semantics identification". In: *2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pp. 546–554. DOI: 10.1109/SAHCN.2014.6990394.

Aly, Heba, Anas Basalamah, and Moustafa Youssef (2015). "LaneQuest: An accurate and energy-efficient lane detection system". In: *2015 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 163–171. DOI: 10.1109/PERCOM.2015.7146523.

Aqel, Mohammad OA, Mohammad H Marhaban, M Iqbal Saripan, and Napsiah Bt Ismail (2016). "Adaptive-search template matching technique based on vehicle acceleration for monocular visual odometry system". In: *IEEJ Transactions on Electrical and Electronic Engineering* 11 (6): pp. 739–752. ISSN: 1931-4981. DOI: 10.1002/tee.22299.

Arulampalam, M. S., S. Maskell, N. Gordon, and T. Clapp (2002). "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking". In: *IEEE Transactions on Signal Processing* 50 (2): pp. 174–188. ISSN: 1053-587X. DOI: 10.1109/78.978374.

Awang Salleh, Dayang Nur Salmi Dharmiza and Emmanuel Seignez (2016). "Monocular visual odometry with road probability distribution factor for lane-level vehicle localization". In: *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 1–6. ISBN: 978-1-5090-3549-6. DOI: 10.1109/ICARCV.2016.7838777.

Awang Salleh, Dayang Nur Salmi Dharmiza and Emmanuel Seignez (2018a). "Longitudinal Error Improvement by Visual Odometry Trajectory Trail and Road Segment Matching". In: *IET Intelligent Transport Systems*. Accepted. ISSN: 1751-9578. DOI: 10.1049/iet-its.2018.5272.

Awang Salleh, Dayang Nur Salmi Dharmiza and Emmanuel Seignez (2018b). "Swift Path Planning: Vehicle Localization by Visual Odometry Trajectory Tracking and Mapping". In: *Unmanned Systems*. Accepted. ISSN: 2301-3869. DOI: 10.1142/S2301385018500085.

Aynaud, Claude, Coralie Bernay-Angeletti, Romuald Aufrere, Laurent Lequievre, Christophe Debain, and Roland Chapuis (2017). "Real-Time Multisensor Vehicle Localization: A Geographical Information System-Based Approach". In: *IEEE Robotics Automation Magazine* 24 (3): pp. 65–74. ISSN: 1070-9932. DOI: 10.1109/MRA.2017.2669399.

Azartash, Haleh, Nima Banai, and Truong Q. Nguyen (Apr. 2014). "An Integrated Stereo Visual Odometry for Robotic Navigation". In: *Robot. Auton. Syst.* 62 (4): pp. 414–421. ISSN: 0921-8890. DOI: 10.1016/j.robot.2013.11.008. URL: http://dx.doi.org/10.1016/j.robot.2013.11.008.

Badino, Hernán and Takeo Kanade (2011a). "A Head-Wearable Short-Baseline Stereo System for the Simultaneous Estimation of Structure and Motion." In: *IAPR Conference on Machine Vision Applications*, pp. 185–189. ISBN: 978-4-901122-11-5.

Badino, Hernan, Daniel Huber, and Takeo Kanade (2011b). *The CMU Visual Localization Data Set*. http://3dvis.ri.cmu.edu/data-sets/localization. Accessed: 10-03-2018.

Badino, Hernán, Akihiro Yamamoto, and Takeo Kanade (2013). "Visual Odometry by Multi-frame Feature Integration". In: *2013 IEEE International Conference on Computer Vision Workshops*, pp. 222–229. DOI: 10.1109/ICCVW.2013.37.

Bandemer, H. (1978). "Bibby, J., H. Toutenburg: Prediction and improved estimation in linear models. J. Wiley & Sons, Chichester-New York-Brisbane-Toronto 1977. 201 pp., £ 7.95". In: *Biometrical Journal* 20 (7-8): pp. 826–826. ISSN: 1521-4036. DOI: 10.1002/bimj.197800029.

Barron, John L, David J Fleet, and Steven S Beauchemin (1992). "Performance of optical flow techniques". In: pp. 236–242. ISSN: 1063-6919. DOI: 10.1109/CVPR.1992.223269.

Bay, Herbert, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool (2008). "Speeded-Up Robust Features (SURF)". In: *Computer Vision and Image Understanding* 110 (3): pp. 346–359. ISSN: 1077-3142. DOI: 10.1016/j.cviu.2007.09.014.

Bayes, Mr. and Mr. Price (1763). "An Essay towards Solving a Problem in the Doctrine of Chances. By the Late Rev. Mr. Bayes, F. R. S. Communicated by Mr. Price, in a Letter to

John Canton, A. M. F. R. S." In: *Philosophical Transactions of the Royal Society of London* 53 (0): pp. 370–418. ISSN: 1364-503X. DOI: 10.1098/rstl.1763.0053.

Beall, Chris, Brian J Lawrence, Viorela Ila, and Frank Dellaert (2010). "3D reconstruction of underwater structures". In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4418–4423. DOI: 10.1109/IROS.2010.5649213.

Behley, Jens and Cyrill Stachniss (2018). "Efficient Surfel-Based SLAM using 3D Laser Range Data in Urban Environments". In: DOI: 10.15607/rss.2018.xiv.016.

Bencheikh, S., S. Ibrir, and S. Boukraa (2015). "Attitude observers for accelerated vehicles using IMU and GPS measurements". In: *2015 IEEE 12th International Multi-Conference on Systems, Signals Devices (SSD15)*, pp. 1–6. DOI: 10.1109/SSD.2015.7348192.

Benseddik, Houssem Eddine, Oualid Djekoune, and Mahmoud Belhocine (2014). "SIFT and SURF Performance evaluation for mobile robot-monocular visual odometry". In: *Journal of Image and Graphics* 2 (1): pp. 70–76. ISSN: 2301-3699. DOI: 10.12720/joig.2.1.70-76.

Bétaille, David, François Peyret, and Maxime Voyer and (2015). "Applying Standard Digital Map Data in Map-aided, Lane-level GNSS Location". In: *Journal of Navigation* 68 (05): pp. 827–847. ISSN: 1469-7785. DOI: 10.1017/s0373463315000132.

Blanco, José-Luis, Francisco-Angel Moreno, and Javier Gonzalez-Jimenez (2014). "The Málaga Urban Dataset: High-rate Stereo and Lidars in a realistic urban scenario". In: *International Journal of Robotics Research* 33 (2): pp. 207–214. ISSN: 0278-3649. DOI: 10.1177/0278364913507326. URL: http://www.mrpt.org/MalagaUrbanDataset.

Bosch radar. http://www.bosch.com. Accessed: 10-03-2018.

Boukhers, Zeyd, Kimiaki Shirahama, and Marcin Grzegorzek (2018). "Example-Based 3D Trajectory Extraction of Objects From 2D Videos". In: *IEEE Transactions on Circuits and Systems for Video Technology* 28 (9): pp. 2246–2260. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2017.2727963.

Brubaker, Marcus A, Andreas Geiger, and Raquel Urtasun (2016). "Map-Based Probabilistic Visual Self-Localization". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38 (4): pp. 652–665. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2015.2453975.

Buczko, Martin and Volker Willert (2016a). "Flow-decoupled normalized reprojection error for visual odometry". In: *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1161–1167. DOI: 10.1109/ITSC.2016.7795703.

Buczko, Martin and Volker Willert (2016b). "How to distinguish inliers from outliers in visual odometry for high-speed automotive applications". In: *2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 478–483. DOI: 10.1109/IVS.2016.7535429.

Buczko, Martin and Volker Willert (2017). "Monocular Outlier Detection for Visual Odometry". In: *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 739–745. DOI: 10.1109/IVS.2017.7995805.

Campbell, Jason, Rahul Sukthankar, and Illah Nourbakhsh (2004). "Techniques for evaluating optical flow for visual odometry in extreme terrain". In: *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*. Vol. 4, 3704–3711 vol.4. DOI: 10.1109/IROS.2004.1389991.

Cappello, Francesco, Roberto Sabatini, Subramanian Ramasamy, and Matthew Marino (2015). "Particle filter based multi-sensor data fusion techniques for RPAS navigation and guidance". In: pp. 395–400. DOI: 10.1109/MetroAeroSpace.2015.7180689.

Caron, Francois, Emmanuel Duflos, Denis Pomorski, and Philippe Vanheeghe (2006). "GPS/IMU data fusion using multisensor Kalman filtering: introduction of contextual aspects". In: *Information fusion* 7 (2): pp. 221–230. ISSN: 1566-2535. DOI: 10.1016/j.inffus.2004.07.002. URL: http://www.sciencedirect.com/science/article/pii/S156625350400065X.

Chen, Dongyao, Kyong-Tak Cho, Sihui Han, Zhizhuo Jin, and Kang G. Shin (2015). "Invisible Sensing of Vehicle Steering with Smartphones". In: *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*. MobiSys '15. Florence, Italy: ACM, pp. 1–13. ISBN: 978-1-4503-3494-5. DOI: 10.1145/2742647.2742659. URL: http://doi.acm.org/10.1145/2742647.2742659.

Choi, Min-Seok and Whoi-Yul Kim (2002). "A novel two stage template matching method for rotation and illumination invariance". In: *Pattern Recognition* 35 (1): pp. 119–129. ISSN: 0031-3203. DOI: 10.1016/S0031-3203(01)00025-5.

Chong, Zhuang Jie, Baoxing Qin, Tirthankar Bandyopadhyay, Marcelo H Ang, Emilio Frazzoli, and Daniela Rus (2013). "Synthetic 2d lidar for precise vehicle localization in 3d urban environment". In: *2013 IEEE International Conference on Robotics and Automation*, pp. 1554–1559. DOI: 10.1109/ICRA.2013.6630777.

Comport, Andrew I, Ezio Malis, and Patrick Rives (2007). "Accurate Quadrifocal Tracking for Robust 3D Visual Odometry". In: *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 40–45. DOI: 10.1109/ROBOT.2007.363762.

Corke, Peter, Dennis Strelow, and Sanjiv Singh (2004). "Omnidirectional visual odometry for a planetary rover". In: *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*. Vol. 4, 4007–4012 vol.4. DOI: 10.1109/IROS.2004.1390041.

Cui, D., J. Xue, S. Du, and N. Zheng (2014). "Real-time global localization of intelligent road vehicles in lane-level via lane marking detection and shape registration". In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4958–4964. DOI: 10.1109/IROS.2014.6943267.

Cumani, Aldo (Jan. 2011). "Feature Localization Refinement for Improved Visual Odometry Accuracy". In: 5 (2): pp. 151–158. ISSN: 1998-4464.

Cvišić, Igor, Josip Ćesić, Ivan Marković, and Ivan Petrović (2017). "SOFT-SLAM: Computationally efficient stereo visual simultaneous localization and mapping for autonomous

unmanned aerial vehicles". In: *Journal of Field Robotics* 35 (4): pp. 578–595. DOI: 10.1002/rob.21762. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.21762. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21762.

Cvišić, Igor and Ivan Petrović (2015). "Stereo odometry based on careful feature selection and tracking". In: *2015 European Conference on Mobile Robots (ECMR)*, pp. 1–6. DOI: 10.1109/ECMR.2015.7324219.

Deigmoeller, Joerg and Julian Eggert (2016). "Stereo visual odometry without temporal filtering". In: *German Conference on Pattern Recognition*. Springer, pp. 166–175. ISBN: 978-3-319-45886-1.

Dellaert, Frank and Robert Collins (1999). "Fast image-based tracking by selective pixel integration". In: *Proceedings of the ICCV Workshop on Frame-Rate Vision*. Academic Press, pp. 1–22.

Deschaud, Jean-Emmanuel (2018). "IMLS-SLAM: scan-to-model matching based on 3D data". In: *CoRR* abs/1802.08633. arXiv: 1802.08633. URL: http://arxiv.org/abs/1802.08633.

Doucet, Arnaud, Nando De Freitas, and Neil Gordon (2001). "An introduction to sequential Monte Carlo methods". In: *Sequential Monte Carlo methods in practice*. Springer, pp. 3–14. ISBN: 978-1-4757-3437-9. DOI: 10.1007/978-1-4757-3437-9_1.

Du, X. and K. K. Tan (2016). "Comprehensive and Practical Vision System for Self-Driving Vehicle Lane-Level Localization". In: *IEEE Transactions on Image Processing* 25 (5): pp. 2075–2088. ISSN: 1057-7149. DOI: 10.1109/TIP.2016.2539683.

Durrant-Whyte, H. and T. Bailey (2006). "Simultaneous localization and mapping: part I". In: *IEEE Robotics Automation Magazine* 13 (2): pp. 99–110. ISSN: 1070-9932. DOI: 10.1109/MRA.2006.1638022.

Engel, Jakob, Jöirg Sturm, and Daniel Cremers (2013). "Semi-dense Visual Odometry for a Monocular Camera". In: *2013 IEEE International Conference on Computer Vision*, pp. 1449–1456. DOI: 10.1109/ICCV.2013.183.

Engel, Jakob, Jörg Stückler, and Daniel Cremers (2015). "Large-scale direct SLAM with stereo cameras". In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1935–1942. DOI: 10.1109/IROS.2015.7353631.

Esparza-Jiménez, Jorge, Michel Devy, and José Gordillo (2016). "Visual EKF-SLAM from Heterogeneous Landmarks". In: *Sensors* 16 (4): p. 489. ISSN: 1424-8220. DOI: 10.3390/s16040489.

Fanani, Nolang, Alina Stürck, Marc Barnada, and Rudolf Mester (2017a). "Multimodal scale estimation for monocular visual odometry". In: *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1714–1721. DOI: 10.1109/IVS.2017.7995955.

Fanani, Nolang, Alina Strck, Matthias Ochs, Henry Bradler, and Rudolf Mester (Dec. 2017b). "Predictive Monocular Odometry (PMO)". In: *Image Vision Comput.* 68 (C): pp. 3–13. ISSN: 0262-8856. DOI: 10.1016/j.imavis.2017.08.002.

Fanfani, Marco, Fabio Bellavia, and Carlo Colombo (2016). "Accurate keyframe selection and keypoint tracking for robust visual odometry". In: *Machine Vision and Applications* 27 (6): pp. 833–844. ISSN: 1432-1769. DOI: 10.1007/s00138-016-0793-3.

Fernandez, David and Andrew Price (2004). "Visual odometry for an outdoor mobile robot". In: *IEEE Conference on Robotics, Automation and Mechatronics, 2004.* Vol. 2, 816–821 vol.2. ISBN: 0-7803-8645-0. DOI: 10.1109/RAMECH.2004.1438023.

Floros, G., B. van der Zander, and B. Leibe (2013). "OpenStreetSLAM: Global vehicle localization using OpenStreetMaps". In: *2013 IEEE International Conference on Robotics and Automation*, pp. 1054–1059. DOI: 10.1109/ICRA.2013.6630703.

Forster, Christian, Matia Pizzooli, and Davide Scaramuzza (2014). "SVO: Fast semi-direct monocular visual odometry". In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 15–22. DOI: 10.1109/ICRA.2014.6906584.

Forster, Christian, Zichao Zhang, Michael Gassner, Manuel Werlberger, and Davide Scaramuzza (Apr. 2017). "SVO: Semidirect Visual Odometry for Monocular and Multicamera Systems". In: *Trans. Rob.* 33 (2): pp. 249–265. ISSN: 1552-3098. DOI: 10.1109/TRO.2016.2623335.

Frost, Duncan P, Olaf Kähler, and David W Murray (2016). "Object-aware bundle adjustment for correcting monocular scale drift". In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4770–4776. DOI: 10.1109/ICRA.2016.7487680.

Gade, Kenneth (2005). *Introduction to Inertial Navigation. Tutorial for Geodesi- og Hydrografidagene*. https://www.navlab.net/Publications/Introduction_to_Inertial_Navigation.pdf. Accessed: 10-03-2018. Hoenefoss, Norway.

Gao, Yanbin, Shifei Liu, Mohamed M Atia, and Aboelmagd Noureldin (2015). "INS/GPS/LiDAR integrated navigation system for urban and indoor environments using hybrid scan matching algorithm". In: *Sensors* 15 (9): pp. 23286–23302. ISSN: 1424-8220.

Geiger, Andreas, Julius Ziegler, and Christoph Stiller (2011). "StereoScan: Dense 3d reconstruction in real-time". In: *2011 IEEE Intelligent Vehicles Symposium (IV)*, pp. 963–968. DOI: 10.1109/IVS.2011.5940405.

Geiger, Andreas, Philip Lenz, and Raquel Urtasun (2012). "Are we ready for autonomous driving? The KITTI vision benchmark suite". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361. DOI: 10.1109/CVPR.2012.6248074.

Gomez-Ojeda, Ruben and Javier Gonzalez-Jimenez (2016). "Robust stereo visual odometry through a probabilistic combination of points and line segments". In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2521–2526. DOI: 10.1109/ICRA.2016.7487406.

González, Ramón, Francisco Álvarez Rodriguez, José Luis Guzmán, Cédric Pradalier, and Roland Siegwart (2012). "Combined visual odometry and visual compass for off-road mobile robots localization". In: *Robotica* 30 (6): 865–878. DOI: 10.1017/S026357471100110X.

Gordon, Neil, David Salmond, and Craig Ewing (1995). "Bayesian state estimation for tracking and guidance using the bootstrap filter". In: *Journal of Guidance, Control, and Dynamics* 18 (6): pp. 1434–1443. ISSN: 1533-3884. DOI: 10.2514/3.21565.

Goshtasby, Arthur, Stuart H Gage, and Jon F Bartholic (1984). "A Two-Stage Cross Correlation Approach to Template Matching". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-6 (3): pp. 374–378. ISSN: 0162-8828. DOI: 10.1109/TPAMI.1984.4767532.

Graeter, Johannes, Tobias Schwarze, and Martin Lauer (2015). "Robust scale estimation for monocular visual odometry using structure from motion and vanishing points". In: pp. 475–480. ISSN: 1931-0587. DOI: 10.1109/IVS.2015.7225730.

Graeter, Johannes, Alexander Wilczynski, and Martin Lauer (2018). "LIMO: Lidar-Monocular Visual Odometry". In: *arXiv preprint arXiv:1807.07524*.

Groves, Paul D (2013). *Principles of GNSS, inertial, and multisensor integrated navigation systems, Second Edition*. Artech house. ISBN: 978-1608070053.

Guerrero, Maridalia (2011). "A Comparative Study of Three Image Matcing Algorithms: Sift, Surf, and Fast". PhD thesis. Utah State University.

Guo, Shude and Cai Meng (2012). "Monocular Visual Odometry and Obstacle Detection System Based on Ground Constraints". In: *Proceedings of the 4th International Conference on Social Robotics*. ICSR'12. Chengdu, China: Springer-Verlag, pp. 516–525. ISBN: 978-3-642-34102-1. DOI: 10.1007/978-3-642-34103-8_52. URL: http://dx.doi.org/10.1007/978-3-642-34103-8_52.

Hara, Kosuke and Hideo Saito (2015). "Vehicle Localization Based on the Detection of Line Segments from Multi-Camera Images." In: *Journal of Robotics and Mechatronics* 27 (6): pp. 617–626. ISSN: 0915-3942.

Hartley, R. I. (1997). "In defense of the eight-point algorithm". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19 (6): pp. 580–593. ISSN: 0162-8828. DOI: 10.1109/34.601246.

Hartley, Richard I (1995). *An investigation of the essential matrix*. Tech. rep.

Hata, A. Y., F. S. Osorio, and D. F. Wolf (2014). "Robust curb detection and vehicle localization in urban environments". In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pp. 1257–1262. DOI: 10.1109/IVS.2014.6856405.

Hofmann, Sabine and Claus Brenner (2009). "Quality Assessment of Automatically Generated Feature Maps for Future Driver Assistance Systems". In: *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. GIS '09. Seattle, Washington: ACM, pp. 500–503. ISBN: 978-1-60558-649-6. DOI: 10.1145/1653771.1653854. URL: http://doi.acm.org/10.1145/1653771.1653854.

Hoover, Adam (2016). *Lecture Notes: Particle Filter*. Accessed: 01-08-2016.

Horn, Berthold KP and Brian G Schunck (Aug. 1981). "Determining Optical Flow". In: *Artif. Intell.* 17 (1-3): pp. 185–203. ISSN: 0004-3702. DOI: 10.1016/0004-3702(81)90024-2. URL: http://dx.doi.org/10.1016/0004-3702(81)90024-2.

Hoshiya, Masaru and Etsuro Saito (1984). "Structural identification by extended Kalman filter". In: *Journal of engineering mechanics* 110 (12): pp. 1757–1770. ISSN: 1943-7889. DOI: 10.1061/(ASCE)0733-9399(1984)110:12(1757).

Houenou, Adam, Philippe Bonnifait, Véronique Cherfaoui, and Wen Yao (2013). "Vehicle trajectory prediction based on motion model and maneuver recognition". In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4363–4369. DOI: 10.1109/IROS.2013.6696982.

Howard, Andrew (2008). "Real-time stereo visual odometry for autonomous ground vehicles". In: *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3946–3952. DOI: 10.1109/IROS.2008.4651147.

Huai, Jianzhu, Charles K Toth, and Dorota A Grejner-Brzezinska (2015). "Stereo-inertial odometry using nonlinear optimization". In: *International Technical Meeting of the Satellite Division of the Institute of Navigation 2015*, pp. 2087–2097. ISBN: 978-1-5108-1725-8.

Işık, Şahin (2014). "A comparative evaluation of well-known feature detectors and descriptors". In: *International Journal of Applied Mathematics, Electronics and Computers* 3 (1): pp. 1–6. ISSN: 2147-8228.

Jiang, Yunliang, Yunxi Xu, and Yong Liu (2014). "Performance evaluation of feature detection and feature matching for stereo visual odometry using SIFT and SURF". In: pp. 200–203. DOI: 10.1109/TENCONSpring.2014.6863025.

Johansen, Adam M and Arnaud Doucet (2008). "A note on auxiliary particle filters". In: *Statistics & Probability Letters* 78 (12): pp. 1498–1504. ISSN: 0167-7152. DOI: 10.1016/j.spl.2008.01.032.

Johnson, Andrew E, Steven B Goldberg, Yang Cheng, and Larry H Matthies (2008). "Robust and Efficient Stereo Feature Tracking for Visual Odometry". In: *2008 IEEE International Conference on Robotics and Automation*, pp. 39–46. DOI: 10.1109/ROBOT.2008.4543184.

Julier, Simon J and Jeffrey K Uhlmann (2004). "Unscented filtering and nonlinear estimation". In: *Proceedings of the IEEE* 92 (3): pp. 401–422. ISSN: 0018-9219. DOI: 10.1109/JPROC.2003.823141.

Kaess, Michael, Kai Ni, and Frank Dellaert (2009). "Flow separation for fast and robust stereo odometry". In: *2009 IEEE International Conference on Robotics and Automation*, pp. 3539–3544. DOI: 10.1109/ROBOT.2009.5152333.

Kaipio, Jari and Erkki Somersalo (2006). *Statistical and computational inverse problems*. Vol. 160. Springer Science & Business Media. ISBN: 978-0-387-27132-3. DOI: 10.1007/B138659.

Kalman, R. E. (1960). "A New Approach to Linear Filtering And Prediction Problems". In: *ASME Journal of Basic Engineering* 82 (1): pp. 35–45. DOI: 10.1115/1.3662552.

Kicman, Pawel and J Narkiewicz (2013). "Concept of integrated INS/visual system for autonomous mobile robot operation". In: *TRANSNAV 2013: Marine Navigation and Safety of Sea Transportation*. CRC Press. ISBN: 9781315882987. DOI: 10.1201/b14962-6.

Kilinc, Ahmet Sami and Tamer Baybura (2012). "Determination of Minimum Horizontal Curve Radius Used in the Design of Transportation Structures". In: *FIG Working Week*, pp. 1–11. ISBN: 97887-90907-98-3.

Kitt, Bernd, Andreas Geiger, and Henning Lategahn (2010). "Visual odometry based on stereo image sequences with RANSAC-based outlier rejection scheme". In: *2010 IEEE Intelligent Vehicles Symposium*, pp. 486–492. DOI: 10.1109/IVS.2010.5548123.

Kitt, Bernd Manfred, Joern Rehder, Andrew D. Chambers, Miriam Schonbein, Henning Lategahn, and Sanjiv Singh (2011). "Monocular Visual Odometry using a Planar Road Model to Solve Scale Ambiguity". In: *Proc. European Conference on Mobile Robots*, pp. 43–48.

Kong, Augustine, Jun S Liu, and Wing Hung Wong (1994). "Sequential imputations and Bayesian missing data problems". In: *Journal of the American statistical association* 89 (425): pp. 278–288. ISSN: 1537-274X. DOI: 10.1080/01621459.1994.10476469.

Kos, T., I. Markezic, and J. Pokrajcic (2010). "Effects of multipath reception on GPS positioning performance". In: *Proceedings ELMAR-2010*, pp. 399–402.

Krešo, Ivan and Siniša Šegvic (2015). "Improving the Egomotion Estimation by Correcting the Calibration Bias". In: *Proceedings of the 10th International Conference on Computer Vision Theory and Applications - Volume 3: VISAPP, (VISIGRAPP 2015)*. INSTICC. SciTePress, pp. 347–356. ISBN: 978-989-758-091-8. DOI: 10.5220/0005316103470356.

Lee, B., K. Daniilidis, and D. D. Lee (2015a). "Online self-supervised monocular visual odometry for ground vehicles". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5232–5238. DOI: 10.1109/ICRA.2015.7139928.

Lee, Byung-Hyun, Jong-Hwa Song, Jun-Hyuck Im, Sung-Hyuck Im, Moon-Beom Heo, and Gyu-In Jee (2015b). "GPS/DR error estimation for autonomous vehicle localization". In: *Sensors* 15 (8): pp. 20779–20798. ISSN: 1424-8220. DOI: 10.3390/s150820779.

Lee, S E, E C B Olsen, and W W Wierwille (2004). *A comprehensive examination of naturalistic lane changes*. Tech. rep. United States. National Highway Traffic Safety Administration, p. 236.

Lin, Xiaoyan, Thai Kirubarajan, Yaakov Bar-Shalom, and Simon Maskell (2002). "Comparison of EKF , Pseudomeasurement and Particle Filters for a Bearing-only Target Tracking Problem". In: *Proceedings of SPIE - The International Society for Optical Engineering Journal*. Vol. 4728, pp. 240–250. ISBN: 9780819444783. DOI: 10.1117/12.478508.

Liu, Jun S and Rong Chen (1998). "Sequential Monte Carlo methods for dynamic systems". In: *Journal of the American statistical association* 93 (443): pp. 1032–1044. ISSN: 1537-274X. DOI: 10.1080/01621459.1998.10473765.

Longuet-Higgins, H. C. (1987). "Readings in Computer Vision: Issues, Problems, Principles, and Paradigms". In: ed. by Martin A. Fischler and Oscar Firschein. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. Chap. A Computer Algorithm for Reconstructing a Scene from Two Projections, pp. 61–62. ISBN: 0-934613-33-8. URL: http://dl.acm.org/citation.cfm?id=33517.33523.

Lovegrove, Steven, Andrew J. Davison, and Javier Ibanez Guzman (2011). "Accurate visual odometry from a rear parking camera". In: pp. 788–793. ISSN: 1931-0587. DOI: 10.1109/IVS.2011.5940546.

Lowe, David G (1999). "Object recognition from local scale-invariant features". In: *Proceedings of the Seventh IEEE International Conference on Computer Vision*. Vol. 2, pp. 1150–1157. DOI: 10.1109/ICCV.1999.790410.

Lowe, David G (Nov. 2004). "Distinctive Image Features from Scale-Invariant Keypoints". In: *Int. J. Comput. Vision* 60 (2): pp. 91–110. ISSN: 0920-5691. DOI: 10.1023/B:VISI.0000029664.99615.94.

Lu, Wei, Zhiyu Xiang, and Jilin Liu (2013). "High-performance visual odometry with two-stage local binocular BA and GPU". In: *2013 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1107–1112. DOI: 10.1109/IVS.2013.6629614.

Lu, Wenjie, Emmanuel Seignez, F Sergio A Rodriguez, and Roger Reynaud (2014). "Lane marking based vehicle localization using particle filter and multi-kernel estimation". In: *2014 13th International Conference on Control Automation Robotics Vision (ICARCV)*, pp. 601–606. DOI: 10.1109/ICARCV.2014.7064372.

Lucas, Bruce D, Takeo Kanade, et al. (1981). "An Iterative Image Registration Technique with an Application to Stereo Vision". In: IJCAI'81: pp. 674–679. URL: http://dl.acm.org/citation.cfm?id=1623264.1623280.

Maimone, Mark, Yang Cheng, and Larry Matthies (2007). "Two years of visual odometry on the Mars Exploration Rovers". In: *Journal of Field Robotics, Special Issue on Space Robotics* 24 (3): pp. 169–186. ISSN: 1556-4967. DOI: 10.1002/rob.20184.

Mankowitz, Daniel J and Ehud Rivlin (2015). "CFORB: Circular FREAK-ORB Visual Odometry". In: *arXiv preprint arXiv:1506.05257*.

Mao, Ailin, Christopher G. A. Harrison, and Timothy H. Dixon (1999). "Noise in GPS coordinate time series". In: *Journal of Geophysical Research: Solid Earth* 104 (B2): pp. 2797–2816. ISSN: 2169-9356. DOI: 10.1029/1998jb900033.

Mattern, N., R. Schubert, and G. Wanielik (2010). "High-accurate vehicle localization using digital maps and coherency images". In: *2010 IEEE Intelligent Vehicles Symposium*, pp. 462–469. DOI: 10.1109/IVS.2010.5547974.

Maybeck, Peter S (1982). *Stochastic models, estimation, and control*. Vol. 3. Academic press. ISBN: 978-0124807037.

McManus, Colin, Paul Furgale, and Timothy D Barfoot (2013). "Towards lighting-invariant visual navigation: An appearance-based approach using scanning laser-rangefinders".

In: *Robotics and Autonomous Systems* 61 (8): pp. 836–852. ISSN: 0921-8890. DOI: 10.1016/j.robot.2013.04.008.

Miksik, Ondrej and Krystian Mikolajczyk (2012). "Evaluation of local detectors and descriptors for fast feature matching". In: *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pp. 2681–2684. ISBN: 978-4-9906441-0-9.

Mirabdollah, M Hossein and Bärbel Mertsching (2014). "On the second order statistics of essential matrix elements". In: *German conference on pattern recognition*. Springer International Publishing, pp. 547–557. ISBN: 978-3-319-11752-2.

Mirabdollah, M Hossein and Bärbel Mertsching (2015). "Fast techniques for monocular visual odometry". In: *German Conference on Pattern Recognition*. Springer, pp. 297–307. ISBN: 978-3-319-24947-6. DOI: 10.1007/978-3-319-24947-6_24.

Montgomery, Douglas C (2009). *Introduction to statistical quality control*. John Wiley & Sons (New York). ISBN: 978-0470169926.

Moosmann, Frank and Christoph Stiller (2011). "Velodyne SLAM". In: *2011 IEEE Intelligent Vehicles Symposium (IV)*, pp. 393–398. DOI: 10.1109/IVS.2011.5940396.

Moravec, Hans Peter (1980). "Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover". AAI8024717. PhD thesis. Stanford, CA, USA.

Moreno, Francisco-Angel, Jose-Luis Blanco, and Javier Gonzalez-Jimenez (2016). "A constant-time SLAM back-end in the continuum between global mapping and submapping: application to visual stereo SLAM". In: *The International Journal of Robotics Research* 35 (9): pp. 1036–1056. DOI: 10.1177/0278364915619238.

Mur-Artal, Raul, J. M. M. Montiel, and Juan D. Tardós (2015). "ORB-SLAM: a Versatile and Accurate Monocular SLAM System". In: *CoRR* abs/1502.00956. arXiv: 1502.00956. URL: http://arxiv.org/abs/1502.00956.

Mur-Artal, Raul and Juan D Tardós (2017). "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras". In: *IEEE Transactions on Robotics* 33 (5): pp. 1255–1262. ISSN: 1552-3098. DOI: 10.1109/TRO.2017.2705103.

Naroditsky, Oleg, Xun S Zhou, Jean Gallier, Stergios I Roumeliotis, and Kostas Daniilidis (Apr. 2012). "Two Efficient Solutions for Visual Odometry Using Directional Correspondence". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 34 (4): pp. 818–824. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2011.226. URL: http://dx.doi.org/10.1109/TPAMI.2011.226.

Nguyen, Dinh-Van, Fawzi Nashashibi, Trung-Kien Dao, and Eric Castelli (2017). "Improving poor GPS area localization for intelligent vehicles". In: *2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 417–421. DOI: 10.1109/MFI.2017.8170356.

Nistér, David (2003). "An efficient solution to the five-point relative pose problem". In: *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.* Vol. 2, pp. II–195. DOI: 10.1109/CVPR.2003.1211470.

Nistér, David, Oleg Naroditsky, and James Bergen (2004). "Visual odometry". In: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* Vol. 1, pp. I–I. DOI: 10.1109/CVPR.2004.1315094.

Nistér, David, Oleg Naroditsky, and James Bergen (2006). "Visual odometry for ground vehicle applications". In: *Journal of Field Robotics* 23 (1): pp. 3–20. DOI: 10.1002/rob.20103.

Norton, Harry N. (1989). *Handbook of transducers*. Prentice Hall. ISBN: 9780133825992. URL: https://books.google.fr/books?id=1SdTAAAAMAAJ.

Nourani-Vatani, N., J. Roberts, and M. V. Srinivasan (2009). "Practical visual odometry for car-like vehicles". In: *2009 IEEE International Conference on Robotics and Automation*, pp. 3551–3557. DOI: 10.1109/ROBOT.2009.5152403.

Nourani-Vatani, Navid and Paulo Vinicius Koerich Borges (Sept. 2011). "Correlation-based Visual Odometry for Ground Vehicles". In: *J. Field Robot.* 28 (5): pp. 742–768. ISSN: 1556-4959. DOI: 10.1002/rob.20407. URL: http://dx.doi.org/10.1002/rob.20407.

Orlande, Helcio RB, George S Dulikravich, and Marcelo J Colaço (2008). "Application of Bayesian filters to heat conduction problem". In: *EngOpt*: pp. 1–5.

Page, E. S. (1954). "Continuous Inspection Schemes". In: *Biometrika* 41 (1-2): pp. 100–115. ISSN: 1464-3510. DOI: 10.1093/biomet/41.1-2.100.

Pandey, Gaurav, James R McBride, and Ryan M Eustice (2011). "Ford Campus vision and lidar data set". In: *The International Journal of Robotics Research* 30 (13): pp. 1543–1552. DOI: 10.1177/0278364911400640.

Parra, Ignacio, Miguel Angel Sotelo, David Fernández Llorca, and Manuel Ocaña (May 2010). "Robust Visual Odometry for Vehicle Localization in Urban Environments". In: *Robotica* 28 (3): pp. 441–452. ISSN: 0263-5747. DOI: 10.1017/S026357470900575X. URL: http://dx.doi.org/10.1017/S026357470900575X.

Peker, Ali Ufuk, Oguz Tosun, Huseyin Levent Akin, and Tankut Acarman (2014). "Fusion of map matching and traffic sign recognition". In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pp. 867–872. DOI: 10.1109/IVS.2014.6856536.

Pereira, Fabio, Joel Luft, Gustavo Ilha, Arthur Sofiatti, and Altamiro Susin (2017). "Backward Motion for Estimation Enhancement in Sparse Visual Odometry". In: *2017 Workshop of Computer Vision (WVC)*, pp. 61–66. DOI: 10.1109/WVC.2017.00018.

Persson, Mikael, Tommaso Piccini, Michael Felsberg, and Rudolf Mester (2015). "Robust stereo visual odometry from monocular techniques". In: *2015 IEEE Intelligent Vehicles Symposium (IV)*, pp. 686–691. DOI: 10.1109/IVS.2015.7225764.

Pire, Taihú, Thomas Fischer, Gastón Castro, Pablo De Cristóforis, Javier Civera, and Julio Jacobo Berlles (2017). "S-ptam: Stereo parallel tracking and mapping". In: *Robotics and Autonomous Systems* 93: pp. 27–42. ISSN: 0921-8890. DOI: 10.1016/j.robot.2017.03.019.

Pololu Corporation 2016. https://www.pololu.com/blog/414/new-products-magnetic-quadrature-encoders-for-micro-metal-gearmotors. Accessed: 09-03-2018.

Qu, Xiaozhi, Bahman Soheilian, and Nicolas Paparoditis (2018). "Landmark based localization in urban environment". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 140: pp. 90–103. ISSN: 0924-2716. DOI: 10.1016/j.isprsjprs.2017.09.010.

Quddus, Mohammed, Robert Noland, and Washington Y. Ochieng (Sept. 2006). "A High Accuracy Fuzzy Logic Based Map Matching Algorithm for Road Transport". In: *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations* 10 (3): pp. 103–115. DOI: 10.1080/15472450600793560.

Rabe, Johannes, Martin Meinke, Marc Necker, and Christoph Stiller (2016). "Lane-level map-matching based on optimization". In: *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1155–1160. DOI: 10.1109/ITSC.2016.7795702.

Rigatos, Gerasimos G. (Nov. 2010). "Extended Kalman and Particle Filtering for Sensor Fusion in Motion Control of Mobile Robots". In: *Math. Comput. Simul.* 81 (3): pp. 590–607. ISSN: 0378-4754. DOI: 10.1016/j.matcom.2010.05.003. URL: http://dx.doi.org/10.1016/j.matcom.2010.05.003.

Ristic, Branko, Sanjeev Arulampalam, and Neil Gordon (2004). "Beyond the kalman filter - Book Review". In: *IEEE Aerospace and Electronic Systems Magazine* 19 (7): pp. 37–38. ISSN: 0885-8985. DOI: 10.1109/MAES.2004.1346848.

Rosten, Edward and Tom Drummond (2006). "Machine Learning for High-speed Corner Detection". In: *Proceedings of the 9th European Conference on Computer Vision - Volume Part I*. ECCV'06. Graz, Austria: Springer-Verlag, pp. 430–443. ISBN: 3-540-33832-2, 978-3-540-33832-1. DOI: 10.1007/11744023_34. URL: http://dx.doi.org/10.1007/11744023_34.

Salvucci, Dario D. (2004). "Inferring Driver Intent: A Case Study in Lane-Change Detection". In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 48 (19): pp. 2228–2231. ISSN: 1541-9312. DOI: 10.1177/154193120404801905.

Sanfourche, Martial, Vincent Vittori, and Guy Le Besnerais (2013). "eVO: A realtime embedded stereo odometry for MAV applications". In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2107–2114. DOI: 10.1109/IROS.2013.6696651.

Sasiadek, J.Z. and Q. Wang (2001). "Fuzzy Adaptive Kalman Filtering for INS/GPS Data Fusion and Accurate Positioning". In: *IFAC Proceedings Volumes* 34 (15). 15th IFAC Symposium on Automatic Control in Aerospace, Bologna/Forlì, Italy, September 2-7, 2001: pp. 410 –415. ISSN: 1474-6670. DOI: 10.1016/S1474-6670(17)40762-2. URL: http://www.sciencedirect.com/science/article/pii/S1474667017407622.

Satzoda, Ravi Kumar and Mohan Manubhai Trivedi (2015). "Drive Analysis Using Vehicle Dynamics and Vision-Based Lane Semantics". In: *IEEE Transactions on Intelligent Transportation Systems* 16 (1): pp. 9–18. ISSN: 1524-9050. DOI: 10.1109/TITS.2014.2331259.

Scaramuzza, D. and F. Fraundorfer (2011). "Visual Odometry [Tutorial]". In: *IEEE Robotics Automation Magazine* 18 (4): pp. 80–92. ISSN: 1070-9932. DOI: 10.1109/MRA.2011.943233.

Scaramuzza, Davide and Roland Siegwart (2008). "Appearance-Guided Monocular Omnidirectional Visual Odometry for Outdoor Ground Vehicles". In: *IEEE Transactions on Robotics* 24 (5): pp. 1015–1026. ISSN: 1552-3098. DOI: 10.1109/TRO.2008.2004490.

Scaramuzza, Davide, Friedrich Fraundorfer, and Roland Siegwart (2009). "Real-time monocular visual odometry for on-road vehicles with 1-point RANSAC". In: *2009 IEEE International Conference on Robotics and Automation*, pp. 4293–4299. DOI: 10.1109/ROBOT.2009.5152255.

Schlegel, Dominik, Mirco Colosi, and Giorgio Grisetti (2017). "ProSLAM: Graph SLAM from a Programmer's Perspective". In: *CoRR* abs/1709.04377. arXiv: 1709.04377. URL: http://arxiv.org/abs/1709.04377.

Schreiber, M., A. Hellmund, and C. Stiller (2015). "Multi-drive feature association for automated map generation using low-cost sensor data". In: *2015 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1140–1147. DOI: 10.1109/IVS.2015.7225837.

Sensors equipped on a Tesla model S. https://spectrum.ieee.org/cars-that-think/transportation/self-driving/teslas-model-s-will-offer-360degree-sonar. Accessed: 10-3-2018.

Sensors equipped on a Waymo autonomous car. http://www.businessinsider.fr/us/how-does-googles-waymo-self-driving-car-work-graphic-2017-1. Accessed: 10-03-2018.

Song, Shiyu and Manmohan Chandraker (2014). "Robust Scale Estimation in Real-Time Monocular SFM for Autonomous Driving". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1566–1573. DOI: 10.1109/CVPR.2014.203.

Sorenson, Harold W (1970). "Least-squares estimation: from Gauss to Kalman". In: *IEEE Spectrum* 7 (7): pp. 63–68. ISSN: 0018-9235. DOI: 10.1109/MSPEC.1970.5213471.

Stewénius, Henrik, Christopher Engels, and David Nistér (2006). "Recent developments on direct relative orientation". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 60 (4): pp. 284 –294. ISSN: 0924-2716. DOI: 10.1016/j.isprsjprs.2006.03.005. URL: http://www.sciencedirect.com/science/article/pii/S092427160600030X.

Suhr, Jae Kyu, Jeungin Jang, Daehong Min, and Ho Gi Jung (2017). "Sensor Fusion-Based Low-Cost Vehicle Localization System for Complex Urban Environments". In: *IEEE Transactions on Intelligent Transportation Systems* 18 (5): pp. 1078–1086. ISSN: 1524-9050. DOI: 10.1109/TITS.2016.2595618.

Sukkarieh, Salah, Eduardo Mario Nebot, and Hugh F Durrant-Whyte (1999). "A high integrity IMU/GPS navigation loop for autonomous land vehicle applications". In: *IEEE Transactions on Robotics and Automation* 15 (3): pp. 572–578. ISSN: 1042-296X. DOI: 10.1109/70.768189.

Sünderhauf, Niko and Peter Protzel (2007). *Stereo odometry—a review of approaches*. Tech. rep.

Tao, Zui, Philippe Bonnifait, Vincent Fremont, and Javier Ibanez-Guzman (2013). "Lane marking aided vehicle localization". In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pp. 1509–1515. DOI: 10.1109/ITSC.2013.6728444.

Tardif, Jean-Philippe, Michael George, Michel Laverne, Alonzo Kelly, and Anthony Stentz (2010). "A new approach to vision-aided inertial navigation". In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4161–4168. DOI: 10.1109/IROS.2010.5651059.

Toledo, Tomer and David Zohar (2007). "Modeling Duration of Lane Changes". In: *Transportation Research Record: Journal of the Transportation Research Board* 1999 (1): pp. 71–78. ISSN: 0361-1981. DOI: 10.3141/1999-08.

Toledo-Moreo, R., D. Betaille, and F. Peyret (2010). "Lane-Level Integrity Provision for Navigation and Map Matching With GNSS, Dead Reckoning, and Enhanced Maps". In: *IEEE Transactions on Intelligent Transportation Systems* 11 (1): pp. 100–112. ISSN: 1524-9050. DOI: 10.1109/TITS.2009.2031625.

Tomasi, Carlo and Takeo Kanade (1991). "Detection and tracking of point features". In: *International Journal of Computer Vision* 9 (3): pp. 137–154. ISSN: 1573-1405.

Toro, Federico Grasso, Damian Eduardo Diaz Fuentes, Debiao Lu, Uwe Becker, Hansjorg Manz, and Baigen Cai (2015). "Particle Filter technique for position estimation in GNSS-based localisation systems". In: *2015 International Association of Institutes of Navigation World Congress (IAIN)*, pp. 1–8. DOI: 10.1109/IAIN.2015.7352236.

Van Hamme, David, Werner Goeman, Peter Veelaert, and Wilfried Philips (2015). "Robust monocular visual odometry for road vehicles using uncertain perspective projection". In: *EURASIP Journal on Image and Video Processing* 10: pp. 1–18. ISSN: 1687-5281. DOI: 10.1186/s13640-015-0065-6.

Velas, Martin, Michal Spanel, Michal Hradis, and Adam Herout (2018). "CNN for IMU assisted odometry estimation using velodyne LiDAR". In: *2018 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pp. 71–77. DOI: 10.1109/ICARSC.2018.8374163.

Vivacqua, Rafael, Raquel Vassallo, and Felipe Martins (2017). "A Low Cost Sensors Approach for Accurate Vehicle Localization and Autonomous Driving Application". In: *Sensors* 17 (10): p. 2359. ISSN: 1424-8220. DOI: 10.3390/s17102359.

Wan, Eric A and Rudolph Van Der Merwe (2000). "The unscented Kalman filter for nonlinear estimation". In: *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pp. 153–158. DOI: 10.1109/ASSPCC.2000.882463.

Wang, Daobin, Huawei Liang, Hui Zhu, and Shuai Zhang (2014). "A Bionic Camera-Based Polarization Navigation Sensor". In: vol. 14. 7, pp. 13006–13023. DOI: 10.3390/s140713006. URL: http://www.mdpi.com/1424-8220/14/7/13006.

Wang, Rui, Martin Schwörer, and Daniel Cremers (2017). "Stereo DSO: Large-Scale Direct Sparse Visual Odometry with Stereo Cameras". In: *CoRR* abs/1708.07878. arXiv: 1708.07878. URL: http://arxiv.org/abs/1708.07878.

Weidl, Galia, Anders L. Madsen, Viacheslav Tereshchenko, Dietmar Kasper, and Gabi Breuel (2015). "Early Recognition of Maneuvers in Highway Traffic". In: *Lecture Notes in Computer Science*. Springer International Publishing, pp. 529–540. DOI: 10.1007/978-3-319-20807-7_48.

Weiss, H and J Moore (1980). "Improved extended Kalman filter design for passive tracking". In: *IEEE Transactions on Automatic Control* 25 (4): pp. 807–811. ISSN: 0018-9286. DOI: 10.1109/TAC.1980.1102436.

Winkler, Robert L (1972). *An introduction to Bayesian inference and decision/by Robert L. Winkler.* Tech. rep.

Witt, Jonas and Uwe Weltin (2013). "Robust stereo visual odometry using iterative closest multiple lines". In: pp. 4164–4171. ISSN: 2153-0858. DOI: 10.1109/IROS.2013.6696953.

Wolcott, Ryan W and Ryan M Eustice (2014). "Visual localization within lidar maps for automated urban driving". In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 176–183. DOI: 10.1109/IROS.2014.6942558.

Wold, H.O.A. (1938). *A Study in the Analysis of Stationary Time Series*. A Study in the Analysis of Stationary Time Series. Almqvist & Wiksells boktrycheri-a.-b. URL: https://books.google.fr/books?id=0BnvAAAAMAAJ.

Woo, Hanwool, Yonghoon Ji, Hitoshi Kono, Yusuke Tamura, Yasuhide Kuroda, Takashi Sugano, Yasunori Yamamoto, Atsushi Yamashita, and Hajime Asama (2017). "Lane-Change Detection Based on Vehicle-Trajectory Prediction". In: *IEEE Robotics and Automation Letters* 2 (2): pp. 1109–1116. ISSN: 2377-3766. DOI: 10.1109/LRA.2017.2660543.

Woodman, Oliver J. (2007). *An introduction to inertial navigation*. Tech. rep.

Worrall, R. D. and A.G.R. Bullen (1970). *A comprehensive examination of naturalistic lane changes*. Tech. rep. Highway Research Record, pp. 30–43.

Wu, F. C., Z. Y. Hu, and F. Q. Duan (2005). "8-point algorithm revisited: factorized 8-point algorithm". In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Vol. 1, 488–494 Vol. 1. DOI: 10.1109/ICCV.2005.3.

Wu, Meiqing, Siew-Kei Lam, and Thambipillai Srikanthan (2017). "A Framework for Fast and Robust Visual Odometry". In: *IEEE Transactions on Intelligent Transportation Systems* 18 (12): pp. 3433–3448. ISSN: 1524-9050. DOI: 10.1109/TITS.2017.2685433.

Yang, Nan, Rui Wang, Jörg Stückler, and Daniel Cremers (2018). "Deep Virtual Stereo Odometry: Leveraging Deep Depth Prediction for Monocular Direct Sparse Odometry". In: *arXiv preprint arXiv:1807.02570*.

Yao, Wen, Huijing Zhao, Franck Davoine, and Hongbin Zha (2012). "Learning lane change trajectories from on-road driving data". In: pp. 885–890. ISSN: 1931-0587. DOI: 10.1109/IVS.2012.6232190.

Ye, Mao, Peifeng Yin, and Wang-Chien Lee (2010). "Location Recommendation for Location-based Social Networks". In: *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*. GIS '10. San Jose, California: ACM, pp. 458–461. ISBN: 978-1-4503-0428-3. DOI: 10.1145/1869790.1869861. URL: http://doi.acm.org/10.1145/1869790.1869861.

Yoneda, Keisuke, Hossein Tehrani, Takashi Ogawa, Naohisa Hukuyama, and Seiichi Mita (2014). "Lidar scan feature for localization with highly precise 3-D map". In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pp. 1345–1350. DOI: 10.1109/IVS.2014.6856596.

Yu, Y., C. Pradalier, and G. Zong (2011). "Appearance-based monocular visual odometry for ground vehicles". In: *2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 862–867. DOI: 10.1109/AIM.2011.6027050.

Zair, S., S. Le Hégarat-Mascle, and E. Seignez (2016). "A-Contrario Modeling for Robust Localization Using Raw GNSS Data". In: *IEEE Transactions on Intelligent Transportation Systems* 17 (5): pp. 1354–1367. ISSN: 1524-9050. DOI: 10.1109/TITS.2015.2502279.

Zeng, Zhe, Tong Zhang, Qingquan Li, Zhongheng Wu, Haixiang Zou, and Chunxian Gao (2016). "Curvedness feature constrained map matching for low-frequency probe vehicle data". In: *International Journal of Geographical Information Science* 30 (4): pp. 660–690. ISSN: 1365-8816. DOI: 10.1080/13658816.2015.1086922.

Zhang, Ji and Sanjiv Singh (2014a). "LOAM: Lidar Odometry and Mapping in Real-time." In: *Robotics: Science and Systems*. Vol. 2, p. 9. ISBN: 978-0-9923747-4-7.

Zhang, Ji, Michael Kaess, and Sanjiv Singh (2014b). "Real-time depth enhanced monocular odometry". In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4973–4980. DOI: 10.1109/IROS.2014.6943269.

Zhang, Ji and Sanjiv Singh (2015). "Visual-lidar odometry and mapping: low-drift, robust, and fast". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2174–2181. DOI: 10.1109/ICRA.2015.7139486.

Zhang, L. and B. K. Ghosh (2000). "Line segment based map building and localization using 2D laser rangefinder". In: *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*. Vol. 3, 2538–2543 vol.3. DOI: 10.1109/ROBOT.2000.846410.

Zhou, Dingfu, Y. Dai, and Hongdong Li (2016). "Reliable scale estimation and correction for monocular Visual Odometry". In: *2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 490–495. ISBN: 978-1-5090-1821-5. DOI: 10.1109/IVS.2016.7535431.

Zhu, Jianke (2017). "Image Gradient-based Joint Direct Visual Odometry for Stereo Camera". In: *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. IJCAI'17. Melbourne, Australia: AAAI Press, pp. 4558–4564. ISBN: 978-0-9992411-0-3. URL: http://dl.acm.org/citation.cfm?id=3171837.3171924.

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Fusion de données pour la localisation de véhicule par suivi de trajectoire provenant de l'odométrie visuelle

Mots clés : odométrie visuelle, localisation, fusion des données

Résumé : Au sein des systèmes avancés d'aide à la conduite (Advanced Driver Assistance Systems - ADAS) pour les systèmes de transport intelligents (Intelligent Transport Systems - ITS), les systèmes de positionnement, ou de localisation, du véhicule jouent un rôle primordial. Le système GPS (Global Positioning System) largement employé ne peut donner seul un résultat précis à cause de facteurs extérieurs comme un environnement contraint ou l'affaiblissement des signaux. Ces erreurs peuvent être en partie corrigées en fusionnant les données GPS avec des informations supplémentaires provenant d'autres capteurs. La multiplication des systèmes d'aide à la conduite disponibles dans les véhicules nécessite de plus en plus de capteurs installés et augmente le volume de données utilisables. Dans ce cadre, nous nous sommes intéressés à la fusion des données provenant de capteurs bas cout pour améliorer le positionnement du véhicule.

Parmi ces sources d'information, en parallèle au GPS, nous avons considérés les caméras disponibles sur les véhicules dans le but de faire de l'odométrie visuelle (Visual Odometry - VO), couplée à une carte de l'environnement. Nous avons étudié les caractéristiques de cette trajectoire reconstituée dans le but d'améliorer la qualité du positionnement latéral et longitudinal du véhicule sur la route, et de détecter les changements de voies possibles.

Après avoir été fusionnée avec les données GPS, cette trajectoire générée est couplée avec la carte de l'environnement provenant d'Open-StreetMap (OSM). L'erreur de positionnement latérale est réduite en utilisant les informations de distribution de voie fournies par OSM, tandis que le positionnement longitudinal est optimisé avec une correspondance de courbes entre la trajectoire provenant de l'odométrie visuelle et les routes segmentées décrites dans OSM.

Pour vérifier la robustesse du système, la méthode a été validée avec des jeux de données KITTI en considérant des données GPS bruitées par des modèles de bruits usuels. Plusieurs méthodes d'odométrie visuelle ont été utilisées pour comparer l'influence de la méthode sur le niveau d'amélioration du résultat après fusion des données. En utilisant la technique d'appariement des courbes que nous proposons, la précision du positionnement connait une amélioration significative, en particulier pour l'erreur longitudinale. Les performances de localisation sont comparables à celles des techniques SLAM (Simultaneous Localization And Mapping), corrigeant l'erreur d'orientation initiale provenant de l'odométrie visuelle.

Nous avons ensuite employé la trajectoire provenant de l'odométrie visuelle dans le cadre de la détection de changement de voie. Cette indication est utile dans pour les systèmes de navigation des véhicules. La détection de changement de voie a été réalisée par une somme cumulative et une technique d'ajustement de courbe et obtient de très bon taux de réussite. Des perspectives de recherche sur la stratégie de détection sont proposées pour déterminer la voie initiale du véhicule.

En conclusion, les résultats obtenus lors de ces travaux montrent l'intérêt de l'utilisation de la trajectoire provenant de l'odométrie visuelle comme source d'information pour la fusion de données à faible coût pour la localisation des véhicules. Cette source d'information provenant de la caméra est complémentaire aux données d'images traitées qui pourront par ailleurs être utilisées pour les différentes taches visée par les systèmes d'aides à la conduite.

Title : Study of Vehicle Localization Optimization with Visual Odometry Trajectory Tracking

Keywords : visual odometry, localization, data fusion

Abstract : With the growing research on Advanced Driver Assistance Systems (ADAS) for Intelligent Transport Systems (ITS), accurate vehicle localization plays an important role in intelligent vehicles. The Global Positioning System (GPS) has been widely used but its accuracy deteriorates and susceptible to positioning error due to factors such as the restricting environments that results in signal weakening. This problem can be addressed by integrating the GPS data with additional information from other sensors. Meanwhile, nowadays, we can find vehicles equipped with sensors for ADAS applications. In this research, fusion of GPS with visual odometry (VO) and the digital map is proposed as a solution to localization improvement with low-cost data fusion.

From the published works on VO, it is interesting to know how the generated trajectory can further improve vehicle localization. By integrating the VO output with GPS and OpenStreetMap (OSM) data, estimates of vehicle position on the map can be obtained. The lateral positioning error is reduced by utilizing lane distribution information provided by OSM while the longitudinal positioning is optimized with curve matching between VO trajectory trail and segmented roads.

To observe the system robustness, the method was validated with KITTI datasets tested with different common GPS noise. Several published VO methods were also used to compare the improvement level after data fusion. Validation results show that the positioning accuracy achieved significant improvement especially for the longitudinal error with curve matching technique. The localization performance is on par with Simultaneous Localization and Mapping (SLAM) SLAM techniques despite the drift in VO trajectory input.

The research on the employability of VO trajectory is extended for a deterministic task in lane-change detection. This is to assist the routing service for lane-level direction in navigation. The lane-change detection was conducted by CUSUM and curve fitting technique that resulted in 100% successful detection for stereo VO. Further study for the detection strategy is however required to obtain the current true lane of the vehicle for lane-level accurate localization.

With the results obtained from the proposed low-cost data fusion for localization, we see a bright prospect of utilizing VO trajectory with information from OSM to improve the performance. In addition, to obtain VO trajectory, the camera mounted on the vehicle can also be used for other image processing applications to complement the system. This research will continue to develop with future works concluded in the last chapter of this thesis.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France