TECHNISCHE UNIVERSITÄT MÜNCHEN
Lehrstuhl für Echtzeitsysteme und Robotik

Data Fusion for Advanced Driver Assistance Systems Based on Random Finite Set Statistic

Feihu Zhang

Vollständiger Abdruck der von der Fakultät der Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:	Prof. Dr. Uwe Baumgarten
Prüfer der Dissertation:	1. Prof. Dr.-Ing. habil. Alois Knoll
	2. Prof. Dr.-Ing. Ren C. Luo

Die Dissertation wurde am 24.03.2016 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 02.08.2016 angenommen.

Abstract

The primary aim of Advanced Driver Assistance Systems (ADAS) is to aid in improving the safety and performance of assisted driving. This is achieved using a suite of integrated sensors to detect, identify and track targets within the environment. The map of target objects is then used to form the situational awareness for highly assisted (and in the future autonomous) driving operations. Since traffic participants are monitored by heterogeneous sensors with different capabilities and different fields of view, target tracking algorithms within ADAS are required to deal with randomly varying numbers of targets with various properties in presence of unknown data association and spatial registration. A rigorous mathematical foundation for dealing such scenarios has been developed, which is called 'Random Finite Set statistic (RFS)'. The RFS framework eliminates the need for pedigree tracking and data association required by traditional Bayesian techniques, but achieves this at the expense of being a conservative (or sub-optimal) estimate.

This thesis has considered the traditional tracking problems in ADAS, which essentially concerns the 'Improve the performance of heterogeneous sensor integration for highly assisted driving functionality based on random finite set framework'. Several applications are developed as individual subsystems including localization, visual odometry and lane detection. In contrast to the traditional approaches, the proposed framework avoids the requirements of explicit data association and high precision calibration (i.e. registration) of the sensors.

All the implementations developed for this thesis were published in international conferences and journals, indicating the credibility and novelty of the work.

Zusammenfassung

Das wichtigste Ziel von Fahrerassistenzsystemen ist die Verbesserung der Sicherheit und Leistungsfähigkeit des assistierten Fahrens. Möglich wird dies durch die Verwendung integrierter Sensor-Sets, mit denen Ziele in der Umgebung detektiert, identifiziert und verfolgt werden können. Eine so generierte Karte, die identifizierten Zielobjekte enthält, ist die Basis für die in der Zukunft für hoch automatisierte Fahrerassistenzsysteme notwendige Umgebungswahrnehmung. Da Verkehrsteilnehmer durch heterogene Sensorsets detektiert werden, welche über unterschiedliche Stärken, Schwächen und Sichtfelder verfügen, muss die Umgebungserfassung in der Lage sein die unterschiedlichen Messwerte der verschiedenen Sensoren in Einklang zu bringen. Dies beinhaltet die Auflösung scheinbarer Widersprüche in Messdaten, die korrekte Zuordnung von Messdaten zu den jeweiligen Zielobjekten, sowie die Eliminierung systematischer Fehler. So könnten unterschiedliche Sensoren unterschiedliche zufällig variierte Mengen und Positionen von Zielen melden und unterschiedliche Sensortypen unterschiedliche Zieleigenschaften erkennen.

Eine rigorose mathematische Basis für den Umgang mit solchen Szenarien sind 'Random Finite Set Statistic (RFS)'. Das RFS-Framework beseitigt die Notwendigkeit für die Verfolgung von Objekttrajektorien und Datenzuordnung, die für traditionelle Bayesianische Methoden, zum Preis von konservativen statt optimalen Ergebnissen.

In dieser Arbeit wurden traditionelle Szenarien für Objekt-Verfolgung für Fahrerassistenzsysteme untersucht, mit dem Ziel die Leistungsfähigkeit heterogener Sensor-Sets durch die Verwendung des Random Finite Set Frameworks zu verbessern. Mehrere Anwendungskonzepte wurden entwickelt und demonstriert. Die entwickelten Konzepte umfassen Szenarien zur Lokalisierung, visuellen Odometrie und Fahrspurerkennung. Im Unterschied zu traditionellen Ansätzen, beseitigen die neu entwickelten Ansätze die Notwendigkeit korrekter Messdatenzuordnung und hoch präziser Sensor Kalibrierung (Sensor Registrierung). Die entwickelten Konzepte und Implementierungen wurden jeweils auf internationalen Konferenzen und in Journals publiziert.

Acknowledgements

There are lots of people I would like to thank for a huge variety of reasons.

First and foremost, I would like to express my gratitude to my supervisor Prof. Dr. Alois Knoll for his support during my studies. His open door policy and academic guidance, together with his willingness to share his experience, have tremendously benefited myself during the period.

I would also like to thank the member of my defense committee, Prof. Ren C. Luo. His reviews, comments and fruitful suggestions helped me a lot to improve the quality of this thesis.

Further, I would like to thank the Dr. Christian Buckl and Dr. Daniel Clarke for their regular guiding and valuable advices from the very beginning of my PhD study. They also have been great friends and I am very glad to have an opportunity to work with them.

During my period as a PhD student, I have met a lot of friendly colleagues who need a special mention. I would like to thank Gereon Hinz, Guang Chen, Dhiraj Gulati, Chao Chen and Caixia Cai.

Finally, I would like thank to my family, first of all to my parents, for their loving support through all the ups and downs. And a special thanks to my wife Fanfan Li for providing substantial support and encouragement during the period of my research work. Without their consistent love and support, I could not have finished this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Traffic accidents are a major cause of death worldwide. A study by the World Health Organization (WHO) reports that, an estimated 1.2 million people are dead in traffic accidents every year, as well as up to 50 million people are injured [2]. Highly assisted and Autonomous driving thus becomes significantly important to prevent accident in traffic scenarios.

In 2004, the U.S. Defense Advanced Research Projects Agency (DARPA) launched a series of autonomous car races, which offered a modern, uniform testing opportunities to examine the state-of-the-art in autonomous driving [3]. Recently, the Google car has also attracted the public eye to the development of autonomous vehicles, which demonstrates that autonomous vehicles can be successful in most traffic scenarios [4]. However, it is still quite challenging for autonomous driving in all possible scenarios [5]. For automotive manufacturers, the technology behind the autonomous driving has been continually refined as a long-term goal, whereas the Advanced Driver Assistance Systems (ADAS) has been proposed as the short-term developments to gradually improve road safety.

Numerous ADAS functions have been developed to help drivers avoid accidents, improve driving efficiency, and reduce driver fatigue, e.g. lane departure warning, vehicle/pedestrian detection and autonomous parking. Adaptive Cruise Control (ACC) plays an important role as a sub-function, to aid in control of the speed of the vehicle and distance to the ego-car in relation to the front vehicle [6]. Lane Departure Warning (LDW) is important for reminding the driver when the vehicle is out of the lane or, in case of the turning indicator light is switched off [7]. Emergency Braking System (EBS) is utilized to avoid the collision or mitigating the impact during critical situations by applying the braking system automatically, which has become mandatory for new heavy vehicles starting from 2013 [8]. As a conclusion, Table 1.1 summaries the different ADAS functions which have already been developed in market (though this list is non-exhaustive and new functions are being introduced on a regular basis).

For a typical autonomous or highly assisted driving scenario, dynamic obstacles are often combined with complex conditions such as driving in narrow streets, turning to avoid pedestrians or braking for traffic lights. To achieve these tasks, a number of heterogeneous sensors are used to monitor the surrounding environment deriving estimates of the physical state of targets. Multi-sensor fusion is used to reduce the uncertainty of state estimates and infer state properties which cannot be directly estimated. For instance, image sensors have drawbacks such as low ability to detect range information but are very efficient in appearance description. Radar provides precise range information but is limited in the representation of shape. LiDAR

Table 1.1: Overview of the ADAS related functions

ADAS Task	Description	Level	Impact
Adaptive Cruise Control	Automatic control speed and distance in relation to the heading vehicles	C	Lo
Traffic Sign Recognition	Recognizing the current road status to control the vehicle more suitable to the environment	I/S/W	Lo+La
Emergency Braking	Suddenly brake if any objects emerges into the minimum safety distance requirement	C	Lo
Pedestrian Detection	By utilizing radar or vision sensors to detect the locations of the pedestrians near the vehicle	W/S	Lo+La
Collision Avoidance	Warn driver in case of any collisions cause to the vehicle	C	Lo
Lane Departure Warning	Warn driver to stay in lane and providing information for potential collision when vehicle in adjacent lanes	W/S	La
Parking Assist	Assist driver parking the vehicle in the right place	I/W/S	La
Rear Collision Warning	Warn driver to avoid collision from backside	W/S	Lo
Surrounding View	Describe surrounding situation when vehicle in a non-dynamic environment	I/S	Lo+La
Intersection Negotiation	regulate vehicle traffic at intersections based on the Internet of vehicle	C	Lo+La
Fully Autonomous Driving	controlled by computer at all scenarios	C	Lo+La
Curve Assistance	Assist driver in keeping certain speed during curve maneuverer for safety issue	W/S	Lo
Navigation system	Provide vehicle position and calculate the optimal to the goal place	I/S/W	Lo+La
Vision Enhancement	Minimum and maximum operating temperature	I/W/S	Lo+La
Driver Monitoring	Limit for vibrations before damage or major fatigue occurs	I/W	Lo

Source: Partly based on [9].
Level: I=information, W=warn, C=control, S=support
Impact: Lo=longitudinal, La=lateral

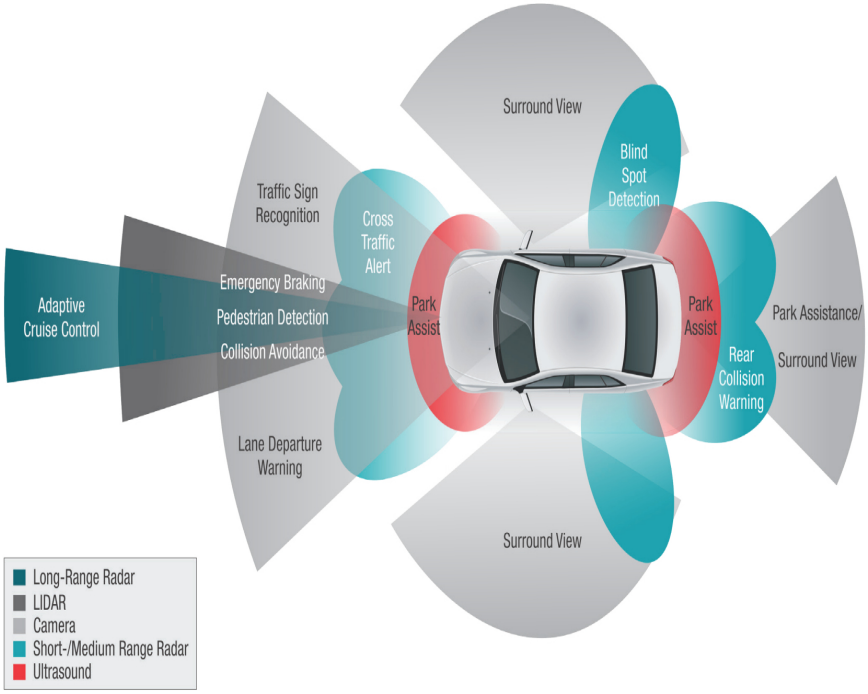


Figure 1.1: TI's data fusion solution for Advanced Driver Assistance Systems (ADAS).

provides both appearance and range information but has issues with object segmentation. Due to the proprieties of the various sensors, a fusion based ADAS system becomes significantly important for the research activities. Figure 1.1 illustrates the ADAS functions with respect to the corresponding sensors in Texas Instruments [10].

By fusing data from heterogeneous sources, it is assumed to have the ability to perceive the whole environment. However, there are still challenges. For a better representation of the environment, the system is required to know the association of each measurement with its target object. In a single-sensor-single-target environment, this association process is known explicitly. However, in a multi-target-multi-sensor environment, it becomes a challenge to update the states with the correct measurements. To solve the problem, traditional solutions formulate the multi-target tracking task in two separate steps, an initial data association step and a subsequent filter to perform the state estimate (e.g. using a Kalman Filter [11]). Under this framework, multiple solutions to the Data Association problem have been proposed, e.g. Nearest Neighbor filter (NN) [12], Probabilistic Data Association filter (PDA) [12], Joint Probabilistic Data Association filter (JPDA) [12] and Multiple Hypothesis Tracking filter (MHT) [13]. However, such data association processes suffer from the combinatorial growth in the number of hypotheses with the number of measurements and targets, specially influenced by the clut-

ters. Furthermore, it requires that the sensor transformation model should be well constructed. For instance, in order to process the whole information, all data should be transformed into a public coordinate system due to the non-uniform distributed fields of view. If the systematic error exists (which is considered as a fixed value originated from the calibration process), the transformed measurements would be definitely influenced. This is called spatial registration, which also increases the challenges of the data association [14].

In this thesis, a new architecture is proposed based on the Random Finite Set (RFS) statistic to integrate data from heterogeneous sources with distributed fields of view, for purpose of improving the performance of environment perception in highly advanced driving assistance systems.

1.2 Motivation and Scope

Autonomous or highly assisted driving is a growing research trend in the automotive domain where the utilization of sensor fusion plays an important role for providing the technical solution. The concept of data fusion is to combine the observed data (uncertain and noisy measurements) and the priori knowledge to provide a reliable, accurate and precise estimate of an unknown quantity or outcome (state). Earlier research of data fusion focused on the single-target tracking and estimation issue, which was formulated and solved in a Bayes setting by representing the target state probability and incorporating statistical methodologies for the sensing action and the target state transition [15]. To convert the single-target tracking problem to a multi-target scenario, an efficient measurement-to-target association combined with the standard Bayes filtering techniques is required. This data association approach suffers from large numbers of technical difficulties in Bayes framework, due to the combinatorial growth in the hypotheses with the corresponding targets and measurements. Further, linearized models and Gaussian noise approximations are often made in order to apply the Kalman filter, which performs poorly under non-linear dynamical models or non-Gaussian distributions [16]. In such case, the inaccuracies further increase the difficulties in the association procedure.

Let us consider a simplified example: A vehicle platform often consists of various sensors that provides data with disparate physical characteristics , e.g. lidar, camera, radar. Also, the urban environment contains multiple dynamic targets with different physical properties, including pedestrians, vehicles, lane markings, traffic signs and other objects, etc. From the sensor side, an unknown number of measurements are received which consists of the true detection and false alarms. The standard Bayes filtering techniques cannot be utilized since the data association (measurement-to-target) is unknown. Also, each sensor have different fields of view which may detect multiple 'ghosts' if target is presented in the overlapping views. It is difficult for track management as the fusion system may consider such 'ghosts' as multiple 'new' targets. In addition, for the perception system, targets should remain in tracks even when they cannot be observed continuously. Traditional solutions have subsequently addressed the above problems by hypothesizing associations between measurements and targets. However, estimating whole environment is a chicken-and-egg problem: knowing measurement-to-target association, it is rather easy to predict and estimate interesting targets, but in cluttered scenes, it is often difficult to associate closely targets without further knowledge about targets.

The recently emerging Finite Set Statistics (FISST), proposed by Mahler as Random Finite Set (RFS), is considered as a systematic and rigorous approach to the Bayes multi-target filtering, which does not require the data association procedure [17]. In the RFS formulation, the collection of individual targets is treated as a set-valued state, and the collection of individual measurements is treated as a set-valued measurement. Modeling set-valued representations as RFS allows the multi-target posterior distribution to be propagated using the Bayes recursion,

as well as the single-target Bayes filter [17, 18, 19, 20, 21].

The focus of this thesis is to develop an effective architecture, to investigate how improved performance for highly assisted driving functionality can be achieved for integration of heterogeneous sensors using random finite set statistics. In particular, attempts have been made to the existing implementations of the Probability Hypothesis Density (PHD) filter, based on the Sequential Monte Carlo and the Gaussian Mixture implementation [22, 23, 24, 25]. Within this work, two main challenges are addressed:

- Avoidance of data association from heterogeneous sources.

  One of the most important tasks of data fusion for ADAS functions is that of data association, in which measurements from heterogeneous sources are correctly associated to their corresponding targets in order to estimate the physical state. By modeling a collection of observations as set-valued observation and a collection of states as set-valued state, the RFS approach provides a natural framework for handling the problem of association uncertainty in Bayes framework. Analogous to the Kalman filter, which provides a computationally efficient approximation to the single-target Bayes filter, the probability hypothesis density filter provides a suboptimal but computationally tractable alternative solution to the multi-target Bayes filter.

- Solution for spatial registration from sensors.

  For modern driving assistance systems, data is processed from heterogeneous sources with distributed fields of view. It is necessary to transform data from different platforms to a uniform reference system, in which spatial registration plays an key important role. Once the measurement is acquired, two kinds of errors are possibly generated. One belongs to the random noise described as the Gaussian distribution, the other is called as bias (systematic error) resulting from the calibration process. The spatial registration is to eliminate the bias before the filtering phase. In the RFS framework, the bias is also considered as an additional state and jointly estimated by the probability hypothesis density filter.

1.3 Key Contributions

This thesis adopts the random finite set statistics to address the data fusion challenges, with respect to the advanced driver assistance systems. The demonstration illuminates that the random finite set framework for multi-object filtering is a theoretically sound and practically viable methodology for developing modern ADAS functions. The major contributions are concluded as follows:

1. A generic framework for driving assistance systems is proposed.

   A mathematical solution for multi-object filtering theoretically addresses the problems of information fusion in the advanced driver assistance systems. In the proposed framework, data is processed using finite sets of observations, rather than just individual point or vector from a single sensor. The processed data is then utilized to detect, identify and classify the interesting objects in advanced driver assistance systems.

2. Several applications have been developed and evaluated for ADAS functions.

   Under the proposed framework, multiple ADAS functions have been implemented including Lane Detection System (LDS), Vehicle Localization System (VLS) and Visual Odometry System (VOS). The PHD filter has been proposed as the key component to

track an unknown number of targets in the presence of unknown data association environment. Traditional fusion challenges are thus addressed in both data association and spatial registration, where the filtering process becomes much easier compared to the state-of-the-art approaches. All implemented applications are evaluated by the benchmark datasets, where the high reliability and flexibility are guaranteed in complex urban environments.

3. It has been demonstrated that the proposed framework is a suitable architecture for future ADAS.

   In the proposed framework, information from heterogeneous sources is collected as a finite set of observations, which is called set-valued measurement. Within the set-valued measurement, data is unordered and constitutes with disparate physical characteristics, whereas the random finite set statistic is utilized to generalize a generic likelihood function according to the statistical models of the sensors. The same principle is also applied in the set-valued state, where the motion models and the likelihood functions are constructed. Therefore, the challenge of adding or replacing sensors has been greatly reduced in such modular data fusion architecture. Adding sensors only requires specifying a specific configuration in set-valued measurement, e.g. sensor range, field of view and statistical parameters.

1.3.1 Publications

First author published papers related to this thesis are listed below.

- **Reviewed Conference Proceedings**

  1. Feihu Zhang, et. al. 'A sensor fusion approach for localization with cumulative error elimination'. 2012 IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI).
  2. Feihu Zhang, et. al. 'Single camera visual odometry based on random finite set statistics'. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).
  3. Feihu Zhang, et. al. 'Visual odometry based on random finite set statistics in urban environment'. Intelligent Vehicles Symposium (IV), 2012 IEEE, 69-74.
  4. Feihu Zhang, et. al. 'Multiple vehicle cooperative localization under random finite set framework'. 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).
  5. Feihu Zhang, et. al. 'A lane marking extraction approach based on Random Finite Set Statistics'. Intelligent Vehicles Symposium (IV), 2013 IEEE, 1143-1148.
  6. Feihu Zhang, et. al. 'Cumulative error estimation from noisy relative measurements'. 2013 International IEEE Conference on Intelligent Transportation Systems (ITSC), 1422-1429.

- **Reviewed Journal Articles**

  1. Feihu Zhang, et. al. 'Multiple Vehicle Cooperative Localization with Spatial Registration Based on a Probability Hypothesis Density Filter'. Sensors 14 (1), 995-1009.

1.4 Thesis organization

The rest of the proposed thesis is organized as follows:

Chapter 2 provides an overview of the mathematical background in the proposed thesis. The Kalman filter (KF), the Particle filter (PF) and the Probability Hypothesis Density (PHD) filter are introduced.

Chapter 3 presents the thesis in scenario I, where the proposed solution is demonstrated by localization applications. Furthermore, a detailed performance evaluation is discussed, in contrast to the state-of-the-art methodologies.

Chapter 4 describes the thesis in scenario II, where the proposed solution is performed by visual odometry applications. Meanwhile, the proposed solution also exhibits the super performance in complex urban environments.

Chapter 5 expresses the thesis in scenario III, where the proposed solution is exhibited by lane detection application.

Chapter 6 includes the concluding remarks of the proposed thesis. In particular, it summarizes the main contributions and outlines the potential research directions in future.

Chapter 2

Theoretical Foundations and Methods

The purpose of this chapter is to present the mathematical foundation of data fusion techniques for this thesis and is organized as follows: Section 2.1 reviews the general Bayesian filter. In Section 2.2, we provide more details for the random finite set statistic, as well as the probability hypothesis density filter. Finally, Section 2.3 compares the random finite set framework with the traditional approaches.

2.1 Bayes Filter

By applying Bayesian statistics and Bayes rule to the stochastic filtering problem, Bayesian filtering is developed to recursively estimate the state of a dynamic system from observations.

The objective of the Bayes filter, in combination with timing information, is to estimate the state of the object at a particular time step, using a sequence of observations. Consider the discrete-time-state-space approaches to the dynamical systems, the target state is represented as a n_x dimensional random vector x_k in a state space $\mathcal{X} \in \mathbb{R}^{n_x}$ at step k. The target state is indirectly observed according to the noisy measurement z_k which takes values from the observation space $\mathcal{Z} \in \mathbb{R}^{n_x}$. The dynamic state estimation issue is therefore concerned with dynamically estimating x_k using the measurement observation during the whole period $z_{1:k} = (z_1, ..., z_k)$.

In the Bayes filter, the state vector x_k is represented by a dynamic model in the form of a Markov transition as follows:

$$x_k = F_k(x_{k-1}, v_{k-1}) \tag{2.1}$$

which represents the transformation of the previous state x_{k-1} and the system noise v_{k-1} at time $k-1$ to the current state x_k at time k. The evolution of the state interval is described by a target transition density

$$f_{k|k-1}(x_k|x_{k-1}) \tag{2.2}$$

The measurement vector is subsequently modeled by

$$z_k = H_k(x_k, w_k) \tag{2.3}$$

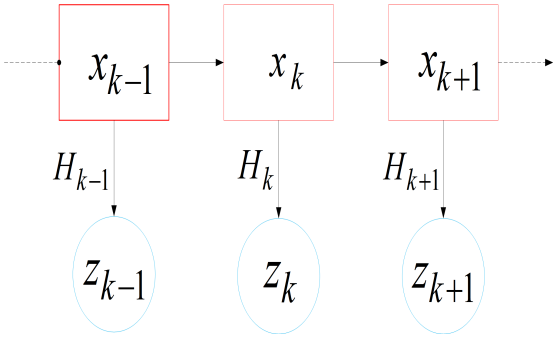


Figure 2.1: A Hidden Markov Model represented by a Bayesian Network. The arrows represent conditional dependencies, such that the state x_k is conditionally dependant upon the state x_{k-1}. The observation of the state z_k is itself conditionally dependent upon the state x_k and modeled the transition function H_k.

which represents the transformation of the current state x_k and the measurement noise w_k at time k to the current measurement z_k. Furthermore, measurement is modeled by the likelihood function

$$g_k(z_k|x_k) \tag{2.4}$$

which is a probability density function.

The problem of estimating a single state modeled by (2.1) and (2.3) is essentially that of a dynamic state space estimation, where the Bayes approach provides a theoretical filtering solution (e.g. [17, 26, 27]). It is assumed that the probability density of the measurement history $z_{1:k}$ conditioned on the system trajectory $x_{1:k}$ given by

$$\pi_{1:k}(z_{1:k}|x_{1:k}) = g_k(z_k|x_k)g_{k-1}(z_{k-1}|x_{k-1})...g_1(z_1|x_1) \tag{2.5}$$

The above equation is called "Hidden Markov Model"; the Bayesian network representation of this is shown in Fig. 2.1.

In Bayesian filtering, $\pi_k(x_k|z_{1:k})$ is the posterior density which contains all information relating to the state vector up to time k, where the whole measurement information is also captured. It is noticed that the posterior density can be recursively propagated using

$$\pi_{k|k-1}(x_k|z_{1:k-1}) = \int f_{k|k-1}(x_k|x_{k-1})\pi_{k-1}(x_{k-1}|z_{1:k-1})\mathrm{d}x_{k-1} \tag{2.6}$$

$$\pi_k(x_k|z_{1:k}) = \frac{g_k(z_k|x_k)\pi_{k|k-1}(x_k|z_{1:k-1})}{\int g_k(z_k|x)\pi_{k|k-1}(x|z_{1:k-1})\mathrm{d}x} \tag{2.7}$$

Notice that the Bayes recursion does not admit closed form solution in general. However, simple approaches such as state space discretization or numerical integration are possible. The Kalman filter provides the optimal solution to the the Bayes filter [16]. However, it is limited to linear systems governed by Gaussian statistics. The Extended Kalman filter (EKF) uses a first order approximation to linearise about the estimate of the current mean and covariance, allowing non-linear systems to be considered [28, 29]. The Unscented Kalman filter (UKF) approximates the first and second moments of the densities by utilizing sampling principles of the unscented transform (UT) [30]. In cases of non-linear and/or non-Gaussian scenarios,

numerical solutions have been investigated by utilizing Sequential Monte-Carlo (SMC) methods [31, 32, 26, 33].

More details about the implementation process could be found in Appendix A.

2.2 Random Finite Set

Although the traditional solutions enable the task of state estimation, however, a data association procedure is required which tends to be ambiguous in several scenarios, e.g. extended objects, closely spaced objects and cluttered environments. The random finite set statistic (RFS) is proposed to address the association issues based on the point process theory and has been investigated by many authors (e.g.[18, 17, 19, 20, 21].)

The key concept of the RFS approach, to multi-sensor-multi-target scenarios, is to treat the collection of targets as a set-valued state, called multi-target state. Also, the collection of measurements is treated as a set-valued observation, called multi-observation. The RFS filtering techniques are utilized to model those set-valued entities as random finite sets to handle the association uncertainties.

In formulation, the input of RFS is defined as a unordered finite set valued random variable. The essential difference between an RFS and a random vector is: the number of constituent points in an RFS is random and the points themselves are also random, distinct and unordered; a random vector is exactly one constituent point which is random. Finite set statistics (FISST) facilitates an intuitive application of the random finite set theory to multi-object tracking applications by casting the problem into the Bayesian framework [17].

A RFS

$$X = \{x^{(1)}, ..., x^{(n)}\} \subset \mathcal{X} \tag{2.8}$$

consists of n unordered points with random object states $x^{(1)}, ..., x^{(n)}$ each taking values in a state space $\mathcal{X} \subseteq \mathbb{R}^{n_x}$, where $n \geq 0$ is a random number which considered as the cardinality of $|X|$. In RFS framework, the randomness of $|X|$ is described by a discrete probability distribution and an appropriate density characterizes the joint distribution of all the elements in X. By representing as this form, RFS X contains the complete state of the individual objects and the corresponding numbers.

Similarly, the RFS

$$Z = \{z^{(1)}, ..., z^{(m)}\} \subset \mathcal{Z} \tag{2.9}$$

is utilized to describe the measurement collection which returns a random number of measurements whose values $z^{(1)}, ..., z^{(m)}$ are also random (each measurement takes value in an observation space $\mathcal{Z} \subseteq \mathbb{R}^{n_z}$). The randomness in the measurement RFS representation is due to the missed detections and the false alarms.

Fig.2.2 illustrates the concept of the RFS where the multi-target tracking problem is transformed into a single target problem. All the measurements are bundled into a "set-valued measurement" which retains all the characteristics of the original sensors. All the targets are bundled into a "set-valued state" which retains all the characteristics of the individual targets. Hence, the RFS provides a natural framework for handling the problem of multi-object tracking while the FISST enables the use of RFS to formulate the multi-object tracking in Bayesian framework.

During the filtering phase, targets can take various statuses: die, survive or evolve to a new state whereas the new target may appear soon. For the surviving and new born target, there is no guarantee on generating measurements. Further measurements received at the sensor may also be obscured by false detections or clutter. Thus, the evolution of the targets' dynamic

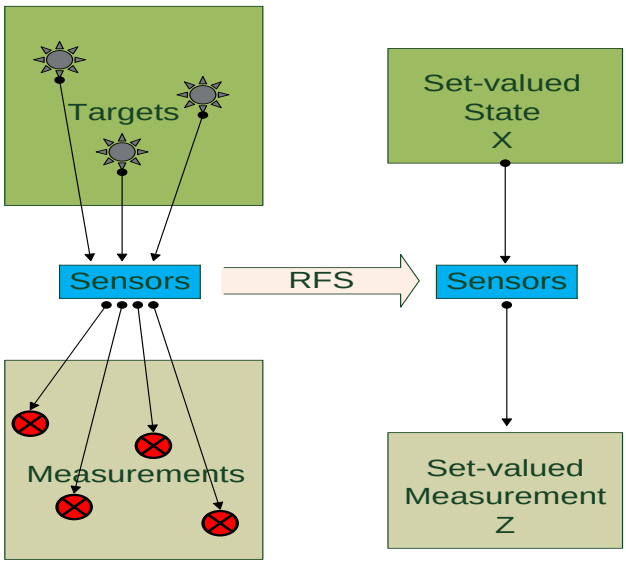


Figure 2.2: Concept of the random finite set in multi-target tracking. All of the measurements and targets are bundled into a "set-valued measurement" and "set-valued state", respectively.

status and the origin of the measurements are unknown. RFS constructs stochastic models for the dynamics of the multi-object state to address the randomness and uncertainty as follows:

The collection of the targets at time $k-1$ modeled by the RFS X_{k-1}, target $x_{k-1} \in X_{k-1}$ may either exist at time k with probability $P_{S,k}(x_{k-1})$ and move to a new state x_k with probability density $f_{k|k-1}(x_k|x_{k-1})$, or die with probability $1 - P_{S,k}(x_{k-1})$ and take on the value $\emptyset$. Thus, the behavior of target with state $x_{k-1} \in X_{k-1}$ is modeled by the RFS

$$S_{k|k-1}(x_{k-1}) \tag{2.10}$$

Since a new target can appear due to spontaneous target birth independent of the existing objects or by spawning from an existing target, the multi-object state model is given by

$$X_k = \left(\bigcup_{x \in X_{k-1}} S_{k|k-1}(x) \right) \cup \left(\bigcup_{x \in X_{k-1}} B_{k|k-1}(x) \right) \cup \Gamma_k \tag{2.11}$$

where $B_{k|k-1}(x)$ denotes the target spawned at time k from the target with state x at time $k-1$. Γ_k denotes the spontaneous birth at time k. It should be noticed that the RFS $S_{k|k-1}(\cdot)$, $B_{k|k-1}(\cdot)$ and Γ_k are independent of each other, which are related to the tracked targets.

Similarly, the RFS observation model incorporates the measurement likelihood and the detection uncertainty with respect to each sensor. The observation at time k is modeled by the RFS as follows

$$Z_k = \left(\bigcup_{x \in X_{k-1}} \Theta_k(x) \right) \cup K_k \tag{2.12}$$

where $\Theta_k(x)$ denotes the object with state x at time k whereas K_k denotes the clutter. A target $x_k \in X_k$, is either detected with probability $P_{D,k}(x_k)$ and contributes a measurement z_k, or missed with probability $1 - P_{D,k}(x_k)$ and generates the value $\emptyset$. The generation of z_k from state x_k is modeled by the likelihood $g_k(\cdot|x_k)$. It is also noticed that the RFS $\Theta_k(\cdot)$ and K_k are independent of each other.

The transition density $f_{k|k-1}(X_k|X_{k-1})$ and the measurement likelihood $g_k(Z_k|X_k)$ represent the uncertainties of X_k and the generation of Z_k, respectively. In RFS framework, the transition density incorporates all the motions such as target birth, spawning, interaction and death. Meanwhile, the measurement likelihood incorporates the characteristics of all the sensors, e.g. detection probability, clutter density and the measurement likelihood.

It is a challenge to calculate the multi-target transition density and the likelihood as Radon-Nikodym derivatives of the appropriate probability measures, however, FISST provides a potential solution which can be utilized to construct the transition density and likelihood from the physical models of sensors and the dynamical models of targets.

- Multi-target Filtering

The RFS framework was proposed by Mahler in the finite set statistics in [17, 34], and is considered as the first systematic and rigorous approach to multi-target filtering and estimation.

Given the posterior density $p_{k-1}(\cdot|Z_{1:k-1})$ at time $k-1$, the multi-target density predicted at time k is given as

$$p_{k-1}(X_k|Z_{1:k-1}) = \int f_{k|k-1}(X_k|X_{k-1}) p_{k-1}(X_{k-1}|Z_{1:k-1}) \mu_s(\mathrm{d}X_{k-1}) \tag{2.13}$$

where μ_s denotes an appropriate measure described in [35].

The updated posterior density $p_k(\cdot|Z_{1:k})$ is acquired from the predicted multi-target density by utilizing the measurement Z_k at time k as follows

$$p_k(X_k|Z_{1:k}) = \frac{g_k(Z_k|X_k) p_{k|k-1}(X_{k|k-1}|Z_{1:k-1})}{\int g_k(Z_k|X) p_{k|k-1}(X|Z_{1:k-1}) \mu_s(\mathrm{d}X)} \tag{2.14}$$

The multi-target prediction (2.13) and estimation (2.14) avoids the inconsistencies caused by errors during the data associations. However, the multi-object Bayes recursion is intractable in practice due to the combinatorial nature of densities and the multiple integrations in Eq. (2.13) and Eq. (2.14). Therefore, the Probability Hypothesis Density (PHD) recursion is proposed as a first order moment approximation to the multi-target Bayes [19, 20, 36, 37, 38, 39].

2.2.1 Probability Hypothesis Density (PHD) Filter

The probability hypothesis density filter is a suboptimal but computationally tractable alternative to the multi-target Bayes filter. It is noted that the intensity function is defined as a recursion that only propagates the first order moments of the objects. The PHD recursion operates on the set-valued state space by propagating the posterior first order moment, where the data association challenge is solved.

For a RFS X on $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ with a probability distribution $\mathcal{P}$, the non-negative function v is considered as the first order moment on $\mathcal{X}$ called the PHD function or intensity. For any closed subset $S \subseteq \mathcal{X}$ the following is given

$$\int_S v(x)dx = \int |X \cap S| \mathcal{P}(\mathrm{d}X) \tag{2.15}$$

With the intensity function v, any integral over region S gives an estimate for the amounts of objects in X that are present in S.

Let $v_{k|k-1}$ and v_k denote the intensities of the predicted and posterior states at time k. Then, with the following assumptions

- Each target evolves and generates measurements independently of one another;
- The birth RFS is a Poisson RFS and independent of the surviving object RFS;
- The clutter RFS is a Poisson RFS and independent of the target generated measurement RFS;
- The predicted and posterior multi-object RFS are approximated by Poisson RFS.

the PHD recursion is defined as

$$v_{k|k-1}(x_k) = \int P_{S,k}(x_{k-1}) f_{k|k-1}(x_k|x_{k-1}) v_{k-1}(x_{k-1}) \mathrm{d}x_{k-1} + \gamma_k(x_k) \tag{2.16}$$

$$v_k(x_k) = [1 - P_{D,k}(x_k)] v_{k|k-1}(x_k) + \sum\nolimits_{z \in Z_k} \frac{P_{D,k}(x_k) g_k(z|x_k) v_{k|k-1}(x_k)}{\kappa_k(z) + \int P_{D,k} g_k(z|\zeta) v_{k|k-1}(\zeta) \mathrm{d}\zeta} \tag{2.17}$$

where $\kappa_k(\cdot)$ denotes the intensity of the clutter, $\gamma_k(\cdot)$ denotes the intensity of spontaneous target birth. It has to be noticed that $N = \int v(x) \mathrm{d}x$ is considered as cardinality and elements of X are independent and identically distributed.

Since the approximation of the posterior based on its first order moment, the equations (B.1) and (B.2) do not incorporate set integrals which significantly reduces the computational complexity. However, it still requires solving multi-dimensional integrals which do not have closed form solutions in general. On one hand, the PHD filter utilizes Sequential Monte Carlo (SMC) implementation which has been introduced in [22, 23, 35]. On the other hand, a Gaussian mixture implementation of the PHD filter (GM-PHD) has been proposed based on the Gaussian sum filters in [24, 25].

Although the PHD filter provides an association free solution for multi-target tracking, the estimated states of each individual objects have no labels associated with them. The lack of temporal association renders the PHD filter unable to provide the whole trajectories of each individual targets. This issue has been investigated in [40, 41]. A potential solution is combines the PHD filter with a multiple hypothesis tracker to associate the states. An efficient labeling PHD filter for the SMC implementation has been presented in [42, 43] whereas the similar scheme for the GM-PHD tracker has been proposed in [44].

2.3 Comparison of Data Fusion Techniques

For ADAS, most fusion techniques are applied in multi-target-multi-sensor scenarios which results in significant challenges:

- Target System
  - Target movements (behaviors) are uncertain.
  - Target status are uncertain (appear, disappear and spawn other targets).
- Sensor System
  - Measurements made are often only a partial observation of the whole state

- Different sensors measuring different properties. Hence the measurement nosies are utilized to describe the confidence of the sensors.
- Measurements contain uncertainties (systematic error, random error).
- Measurements contain miss detection, false detection.

Traditional approaches cannot be directly employed due to the ambiguity issues, which are often solved as follows:

- **Data Association**-in which measurements are correctly associated with targets. In case of an uncertain (or even missing) measurement-to-target association, it is a significant challenge to correctly estimate true states. Also, measurements are often obscured or cluttered (measurements may not only originate from the interesting targets, but also from outliers), further increasing the estimation uncertainty.

As peviously discussed, the key challenge is to associate measurements to the most likely targets. In traditional approaches, a gate or association ellipse is often used to predict the state vector. Data association approaches are thus proposed, which process the output of the receiving measurements within the gate and helps to partition measurements into targets' tracks. As exhibited in Fig. 2.3, given that the state and covariance prediction on a track hypothesis, only measurements falling into the gate are utilized to update the state, the rest are either deleted or utilized to initiate new tracks.

In past decades, data association approaches have been developed including: Global Nearest Neighbor (GNN) [12], Joint Probabilistic Data Association (JPDA) [12] and Multiple Hypothesis Tracking (MHT) [45]. However, it is concluded that beyond a certain level of difficulty, performance of traditional approaches might be significantly influenced. Conventional algorithms, including the GNN and the JPDA, use predefined gate to associate measurements. However, in cases of closely-spaced tracking scenarios, multiple tracks often converge into single track. With respect to the MHT approach, although it has been proven in most scenarios, an exponential explosion in computational complexity still exists during the entire phase.

In random finite set framework, the data association problem is avoided since the PHD filter directly operates on the set space for recursive Bayesian filtering. The collection of individual targets is treated as the set-valued state, and the collection of individual observations is treated as set-valued observation. Fig. 2.4 is a simple demonstration of the PHD filter during the filtering phase.

- **Spatial Registration**-in which measurements are transformed from individual platforms to a uniform reference system to improve the fusion performance. However, errors are also existed which could be mainly divided into: the random noises and the biases. It has to be noticed that the biases should be eliminated before the filtering phase.

Traditional fusion techniques use spatial registration process to compute a bias free state estimation, which is referred to deal with the calibration error induced by individual sensors. Therefore, most solutions are implemented before the filtering phase, e.g. [46, 47, 48]. However, it is noted that conventional algorithms often assume biases as constant values, whereas the fusion performance is significantly influenced when they are also drifting.

Since the PHD filter avoids the data association issue, it is thus considered as a promising method for spatial registration by adding sensor biases as additional states. Thus the sensor biases and the target states are recursively estimated based on the set-valued measurement, respectively.

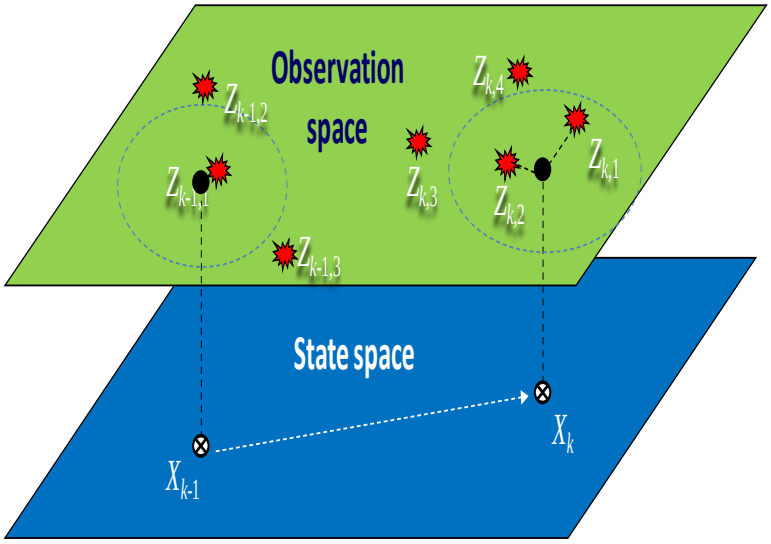


Figure 2.3: Example of measurement-to-target association. The task is to evaluate each observation in its gating region either using Euclidean or Mahalanobis distance.

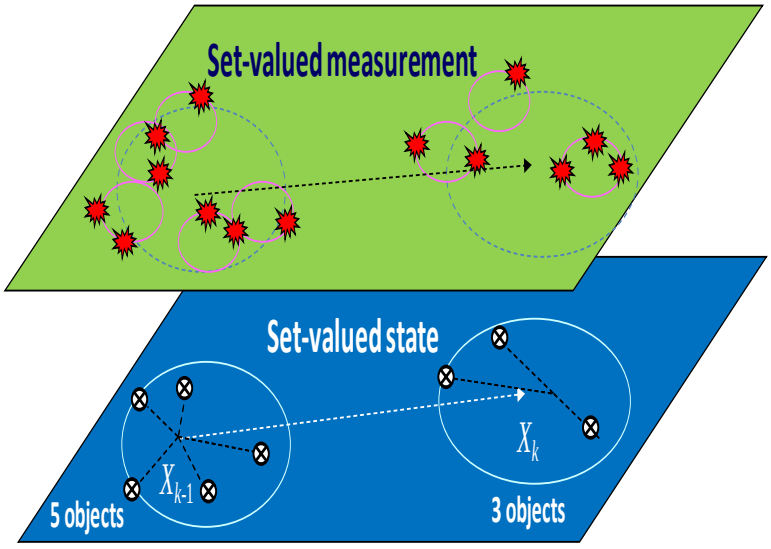


Figure 2.4: Example of set-valued states and set-valued observation in PHD filter.

- **State Update**-in which the measurement-to-target association is confirmed, standard Bayes filtering techniques are utilized to eliminate random noises from individual sensors, e.g. the Kalman filter and the particle filter.

In traditional approaches, the Bayes filter is employed to update the state by fusing the new measurements recursively. However, as previously discussed, there is no closed form in general. Thus, an analytic solution is provided in the various guises of the Kalman Filter and Particle Filter.

The PHD filter is an approximation which alleviates the computational intractability of the optimal multi-target Bayes filter. Instead of propagating the multi-target posterior density, the PHD filter propagates the posterior intensity, a first-order statistical moment of the posterior multi-target state, for extending the Bayes filter from single target into multi-target. In the PHD filter, both the states and their numbers are jointly estimated in the set-valued space, using the given information from both the birth and spawn models. Thus the estimation result is also guaranteed under Bayesian filtering framework.

In light of the issues mentioned above, it is concluded that the traditional techniques offer inadequate solutions to the fusion challenges for driving assistance systems. On the other hand, the Random Finite Set (RFS), especially the Probability Hypothesis Density (PHD) filter is proposed as a mathematically consistent and association free formulation, albeit at the expense of performance, to extend further applications in following chapters.

Chapter 3

Localization

In this chapter, we focus on tackling issues of localization, so as to provide reliable position information for highly assistance driving systems. The problem is addressed by employing the Probability Hypothesis Density (PHD) filter, to detect and localize interesting targets in complex environments.

This chapter is organized as follows: Section 3.1 gives the PHD solution for single vehicle localization. In Section 3.2 and Section 3.3, we give details for cooperative localization of multiple vehicles along with the elimination of sensor biases. Section 3.4 concludes the chapter.

3.1 Single Vehicle Localization

3.1.1 Introduction

Localization plays an important role in Advanced Driver Assistance Systems (ADAS). Global Navigation Satellite Systems (GNSS) such as the United States operated Global Positioning System (GPS) can provide location within a global coordinate framework, to within an accuracy of ($\pm 5m$ [49]). However, their operation requires constant visibility of the sky (i.e. the satellites), location accuracy can decrease with reduced visibility of the sky, such as in mountainous terrain or urban canyons, and signal integrity is susceptible to interference (intentional or otherwise) [50]. Therefore, the use of other sensory data and data fusion can help to provide a localization solution solution under GPS denied situations. It has to be noted that, except GPS, most sensors have their own coordinate reference systems. Systems are thus required to frequently recalibrate themselves to eliminate the cumulative errors.

In the past decades, a variety of approaches have been proposed, using for example terrain mapping and localisation, [51, 52, 53, 54], or visual landmark recognition [55, 56, 57, 58]. Although the proposed solutions eliminate the accumulative errors, both the map building and the land-marking algorithms require huge computational and memory resources. Radio-Frequency Identification (RFID) sensors are thus considered as an alternative solution to eliminate the cumulative error [59]. In addition, RFID based solutions have low computational and memory requirements. However, there are still challenges. A precise cumulative error model is required to calculate the optimal distributions of the RFID sensors. It has been proved that the cumulative error grows super-linearly as $O(d^{3/2})$ (d denotes the length), but, a detailed description of the growth model is still missing [60]. To optimally distribute the RFID tags, a least-square method has been proposed by calibrating the odometry error [61]. But, the major drawback is that the proposed solution only works off-line. Agrawal et al. [62] presented a localiza-

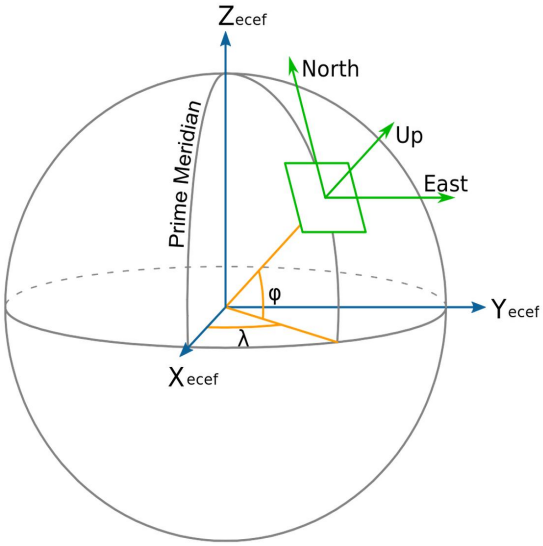


Figure 3.1: Different coordinates systems [1], e.g. Geodetic coordinate system, Earth-Centered-Earth-Fixed coordinate system and local East, North, Up coordinate system.

tion system by fusing data from both stereo cameras and GPS. In their system, an extended Kalman filter is utilized to recursively estimate the positions. Here the limitation is that all measurements are collected after coordinate transformations.

The PHD filter is proposed to recursively estimate the position by utilizing raw data from heterogeneous sources (GPS, gyroscope and velocity sensors). Furthermore, the proposed solution also investigates the statistic model of the cumulative error to improve the overall performance.

3.1.1.1 Coordinate Definition

In this section, we will systematically describe the background of the coordinate systems, in which the raw data from GPS, gyroscope and velocity sensor is considered.

- Geodetic coordinate system

Geodetic coordinate system utilizes a set of reference points to localize positions, in which the earth is considered as an ellipsoid and the corresponding positions are described in terms of latitude ϕ, longitude λ and height h.

- Earth-Centered-Earth-Fixed (ECEF) coordinate system

Earth-Centered-Earth-Fixed (ECEF) coordinate system, also known as "Conventional Terrestrial", represents reference point rotating with the earth around its spin axis. The X axis intersects the earth at 0° of latitude, as well as longitude. The Z axis passes through the north pole but it does not exactly coincide with the instantaneous earth rotational axis. The Y axis is determined by the right-hand rule passed through the equator at 90° longitude.

- Local East, North, Up (ENU) coordinate system

Local East, North, Up (ENU) coordinate system, also known as the ground coordinate system, is often used in automotive domain. The local ENU coordinates are formed from a plane tangent to the earth in a specific location. The origin and axes of the ENU coordinate system are defined as follows: The X axis points toward the ellipsoid east. The Y axis points toward the ellipsoid north, and the Z axis points upward along the ellipsoid normal.

Fig. 3.1 exhibits the proposed Geodetic, ECEF and ENU coordinate systems. To fuse data from heterogeneous sources, the following coordinate systems are also required:

- Vehicle coordinate system

Vehicle coordinate system is defined in a local Cartesian coordinate system, where the origin point is located at the middle of the rear axle. The X, Y axis are parallel with the horizontal and vertical axles, whereas the Z axis is perpendicular to the ground and towards to the sky.

- Navigation coordinate system

Navigation coordinate system is also defined in a Cartesian coordinate system, where the origin point is located at the initial position of the trajectory. The position of vehicle in the navigation coordinate system is calculated as follows:

$$\begin{aligned} X_{nav} &= \sum_{i=1}^{n} d_i \sin\theta_i \\ Y_{nav} &= \sum_{i=1}^{n} d_i \cos\theta_i \\ Z_{nav} &= 0 \end{aligned} \tag{3.1}$$

where θ_i and d_i denotes the orientation and velocity, respectively. It should be noted that the orientation angle is acquired by accumulating the angle changes as follows

$$\theta_i = \sum_{j=1}^{i} \eta_j \tag{3.2}$$

where η denotes the angle change in consecutive two frames.

Fig. 3.2 exhibits the relationship between the local ENU coordinate system, the vehicle coordinate system and the navigation coordinate system.

3.1.1.2 Coordinate Transformations

To fuse data from heterogeneous sources, the transformations among the proposed systems are described as follows:

- Geodetic to ECEF

The transformation from the Geodetic coordinate system to the ECEF coordinate system is an intermediate step in the proposed solution, which is calculated by:

$$\begin{aligned} X_{ecef} &= (N+h)\cos(\phi)\cos(\lambda) \\ Y_{ecef} &= (N+h)\cos(\phi)\sin(\lambda) \\ Z_{ecef} &= (N(1-e^2)+h)\sin(\phi) \end{aligned} \tag{3.3}$$

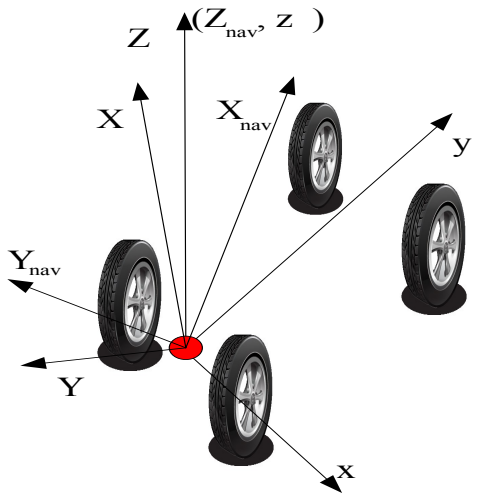


Figure 3.2: Relationship between different coordinate systems, e.g. the ENU coordinate, the vehicle coordinate and the navigation coordinate systems.

with

$$N = \frac{a}{\sqrt{1 - e^2 \sin^2(\phi)}}$$

where $e = 8.1819191 \times 10^{-2}$ and $a = 6378137$ denotes the first numerical eccentricity and the semi-major axis of the WGS84 ellipsoid, respectively.

- ECEF to ENU

A reference point is required during the transformation, which is often considered as the origin point of the ENU coordinate system. The transformation between the ECEF and the ENU coordinate systems is calculated by:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = A \times \begin{bmatrix} X_{ecef} - X^0_{ecef} \\ Y_{ecef} - Y^0_{ecef} \\ Z_{ecef} - Z^0_{ecef} \end{bmatrix} \tag{3.4}$$

with

$$A = \begin{bmatrix} -\sin(\lambda_0) & \cos(\lambda_0) & 0 \\ -\sin(\phi_0)\cos(\lambda_0) & -\sin(\phi_0)\sin(\lambda_0) & \cos(\phi_0) \\ \cos(\phi_0)\cos(\lambda_0) & \cos(\phi_0)\sin(\lambda_0) & \sin(\phi_0) \end{bmatrix}$$

where (ϕ_0, λ_0, h_0) denotes the Geodetic coordinate of the origin point $(X^0_{ecef}, Y^0_{ecef}, Z^0_{ecef})$.

3.1.1.3 Measurement collection

As a summary, raw measurements from heterogeneous sources are transformed and collected as follows:

- Transform GPS from Geodetic to ECEF

Raw data from GPS is presented in format of Geodetic coordinate system, which makes it difficult to predict future locations during the filtering phase (note that GPS provides only location and not orientation). Thus the GPS data is transformed to the ECEF coordinate system.

- Transform gyroscope and velocity data from Vehicle to ECEF

Data from gyroscope and velocity sensors should also be processed in the ECEF coordinate system. To achieve this goal, the local ENU coordinate system is considered as a natural bridge.

$$\begin{aligned} \dot{X}_i &= d_i \sin\gamma_i \\ \dot{Y}_i &= d_i \cos\gamma_i \\ \dot{Z}_i &= 0 \\ \gamma_i &= \theta_i + \delta \end{aligned} \tag{3.5}$$

where γ_i and θ_i denotes the orientation in both the ENU and the navigation coordinate systems respectively. δ denotes the bias between the vehicle coordinate system and the ENU coordinate system at the origin point. $(\dot{X}_i, \dot{Y}_i, \dot{Z}_i)$ denotes the velocity in the local ENU coordinate system. It has to be noted that the velocity in the Z direction is defined as 0.

Irrespective of the coordinate systems, the local ENU or the navigation, the topology of the trajectory remains unchanged. The bias δ is thus represented by applying Euler's rotation (Appendix. C) theorem as follows:

$$\begin{bmatrix} X_{nav} \\ Y_{nav} \\ Z_{nav} \end{bmatrix} = \begin{bmatrix} \cos\delta & -\sin\delta & 0 \\ \sin\delta & \cos\delta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{3.6}$$

Based on Eq. (3.6), the Singular Value Decomposition (SVD) method is utilized to calculate the bias δ [63]. More details about the SVD implementation are introduced in Appendix E.

Until now, raw data from both the gyroscope and velocity sensors is represented as the velocity in the ENU coordinate system. To acquire the data in the reference system, Eq. (3.4) is utilized as follows:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = A \times \begin{bmatrix} \dot{X}_{ecef} \\ \dot{X}_{ecef} \\ \dot{X}_{ecef} \end{bmatrix} \tag{3.7}$$

with

$$A = \begin{bmatrix} -\sin(\lambda_0) & \cos(\lambda_0) & 0 \\ -\sin(\phi_0)\cos(\lambda_0) & -\sin(\phi_0)\sin(\lambda_0) & \cos(\phi_0) \\ \cos(\phi_0)\cos(\lambda_0) & \cos(\phi_0)\sin(\lambda_0) & \sin(\phi_0) \end{bmatrix}$$

where the data in the ECEF coordinate system is acquired by rearranging the above equation.

Measurements from heterogeneous sensors are thus transformed into the ECEF coordinate system, in which the PHD filter is utilized to estimate the corresponding state.

3.1.2 PHD Filter Implementation

The PHD implementation is described as follows:

- State Model

The state is comprised of the poses of the vehicle including the position and velocity. In this scenario, the state is represented in the ECEF coordinate system as

$$\mathbf{x}_k = [X, Y, Z, \dot{X}, \dot{Y}, \dot{Z}]^T \tag{3.8}$$

where k denotes the steps.

- Process Model

To predict the future poses, the process model is used to describe the translation of the state. In practice, the movement of the vehicle is unknown and the constant velocity model in discrete space is often utilized for simplification.

$$\mathbf{x}_{k+1} = F_k \mathbf{x}_k + \mathbf{w}_k \tag{3.9}$$

where the linear Gaussian dynamics matrix F_k and covariance matrix Q_k are represented as

$$F_k = \begin{bmatrix} \mathbf{I}_3 & \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix}, Q_k = \begin{bmatrix} \mathbf{I}_3/4 & \mathbf{I}_3/2 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix}, \tag{3.10}$$

where $\mathbf{I}_n$ and $\mathbf{O}_n$ denote the $n \times n$ identity and zero matrices, respectively.

- Measurement Model

As previously mentioned, the measurement is represented in the ECEF coordinate system. Thus the measurement model can be described as follows:

$$\mathbf{z}_k = H_k \mathbf{x}_k + \mathbf{v}_k \tag{3.11}$$

To map the state to the observation space, the measurement matrix H_k is described as

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & A \end{bmatrix}$$

where A denotes the transformation matrix in Eq. (3.7).

It should be noted that the proposed solution would also work either GPS or Odometry not be available to the algorithm. For instance, when GPS is not available, the position is estimated by replacing $\mathbf{I}$ to $\mathbf{0}$. Similarly, A is replaced by $\mathbf{0}$ when the odometry is not available.

- PHD Recursion

Finally, the PHD filter is utilized as follows:

$$v_{k|k-1}(x_k) = \int P_{S,k}(x_{k-1}) f_{k|k-1}(x_k|x_{k-1}) v_{k-1}(x_{k-1}) \mathrm{d}x_{k-1} + \gamma_k(x_k) \tag{3.12}$$

$$v_k(x_k) = [1 - P_{D,k}(x_k)] v_{k|k-1}(x_k) + \sum\nolimits_{z \in Z_k} \frac{P_{D,k}(x_k) g_k(z|x_k) v_{k|k-1}(x_k)}{\kappa_k(z) + \int P_{D,k} g_k(z|\zeta) v_{k|k-1}(\zeta) \mathrm{d}\zeta} \tag{3.13}$$

where x_k equals to the state $\mathbf{x}_k$ in this scenario.

In general, the PHD filter has no closed form formulas. In this scenario, the Gaussian Mixture (GM) PHD filter is utilized since both the process and measurement models are linearly represented. More details about the GMPHD implementation can be found in Appendix. B.

- Further Extension

Using the PHD filter, the location is estimated by fusing information from heterogeneous sources. The proposed approach still needs to be re-calibrated frequently, due to the cumulative error in velocity. To address this issue, we further extend the proposed approach as follows:

Assuming measurements from both the gyroscope and velocity sensors (d_i, η_i) could also be represented as

$$\begin{aligned} \eta_i &= \bar{\eta}_i + \tilde{\eta}_i \\ d_i &= \bar{d}_i + \tilde{d}_i \end{aligned} \tag{3.14}$$

where $\tilde{\eta}_i$ and $\tilde{d}_i$ denote the independent Gaussian distributions with zero mean, deviations δ_η and δ_d. $\bar{\eta}_n$ and $\bar{d}_n$ denote the corresponding true values.

According to Eq. (3.5), the corresponding velocity in the local ENU coordinate system is represented as:

$$\dot{X}_i = d_i \sin\gamma_i = d_i \sin(\sum_{j=1}^{i} \eta_j + \delta)$$

$$\dot{Y}_i = d_i \cos\gamma_i = d_i \cos(\sum_{j=1}^{i} \eta_j + \delta)$$

$$\dot{Z}_i = 0$$

Rearranging the above equation, we have

$$\begin{aligned} \dot{X}_i &= \dot{X}_i^t + \dot{X}_i^e = (\bar{d}_i + \tilde{d}_i)\sin(\sum_{j=1}^{i}(\bar{\eta}_j + \tilde{\eta}_j) + \delta) \\ &= [\bar{d}_i + \tilde{d}_i] \cdot \left\{ \sin[\sum_{j=1}^{i}\bar{\eta}_j + \sum_{j=1}^{i}\tilde{\eta}_j] \cdot \cos\delta + \cos[\sum_{j=1}^{i}\bar{\eta}_j + \sum_{j=1}^{i}\tilde{\eta}_j] \cdot \sin\delta \right\} = [\bar{d}_i + \tilde{d}_i] \cdot \mathbf{M} \end{aligned} \tag{3.15}$$

$$\begin{aligned} \dot{Y}_i &= \dot{Y}_i^t + \dot{Y}_i^e = (\bar{d}_i + \tilde{d}_i)\cos(\sum_{j=1}^{i}(\bar{\eta}_j + \tilde{\eta}_j) + \delta) \\ &= [\bar{d}_i + \tilde{d}_i] \cdot \left\{ \cos[\sum_{j=1}^{i}\bar{\eta}_j + \sum_{j=1}^{i}\tilde{\eta}_j] \cos\delta - \sin[\sum_{j=1}^{i}\bar{\eta}_j + \sum_{j=1}^{i}\tilde{\eta}_j] \sin\delta \right\} = [\bar{d}_i + \tilde{d}_i] \cdot \mathbf{N} \end{aligned} \tag{3.16}$$

where $\mathbf{M}$ and $\mathbf{N}$ represent the combination between the true values and the Gaussian noises (More information is introduced in [64]).

The ground-truth of the velocity is represented as

$$\dot{X}_i^t = \bar{d}_i \sin(\sum_{j=1}^{i} \bar{\eta}_j + \delta)$$

$$\dot{Y}_i^t = \bar{d}_i \cos(\sum_{j=1}^{i} \bar{\eta}_j + \delta)$$

and the cumulative errors are represented as

$$\dot{X}_i^e = \bar{d}_i \cdot (\mathbf{A_1} + \mathbf{A_2}) + \tilde{d}_i \cdot (\mathbf{A_3} + \mathbf{A_4}) \tag{3.17}$$

$$\dot{Y}_i^e = \bar{d}_i \cdot (\mathbf{B_1} - \mathbf{B_2}) + \tilde{d}_i \cdot (\mathbf{B_3} - \mathbf{B_4}) \tag{3.18}$$

where $\mathbf{A_i}$ and $\mathbf{B_i}$ $(i = 1, 2, 3, 4)$ represent the combination between the true values and the noises (Details are introduced in [64]).

As noises from gyroscope and velocity sensors are Gaussian distributions, we have

$$\tilde{\eta} \sim N(0, \delta_\eta^2),\ \tilde{d} \sim N(0, \delta_d^2) \tag{3.19}$$

The statistical proprieties are acquired as

$$\sum_{j=1}^{i} \tilde{\eta}_j \sim \mathcal{N}(0, i\delta_\eta^2),\ \sum_{j=1}^{i} \tilde{d}_i \sim \mathcal{N}(0, i\delta_d^2)$$

where the first order moment is given by

$$E(\cos \sum_{j=1}^{i} \tilde{\eta}_j) = e^{-i\frac{\delta_\eta^2}{2}},\ E(\sin \sum_{j=1}^{i} \tilde{\eta}_j) = 0$$

Hence the expectation of the cumulative error is represented as

$$\begin{aligned} E[\dot{X}_i^e|\bar{\eta}, \bar{d}] &= \bar{d}_i[\sin \sum_{j=1}^{i} \bar{\eta}_j (e^{-\frac{i\delta_\eta^2}{2}} - 1) \cos\delta + \cos \sum_{j=1}^{i} \bar{\eta}_j (e^{-\frac{i\delta_\eta^2}{2}} - 1) \sin\delta] \\ E[\dot{Y}_i^e|\bar{\eta}, \bar{d}] &= \bar{d}_i[\cos \sum_{j=1}^{i} \bar{\eta}_j (e^{-\frac{i\delta_\eta^2}{2}} - 1) \cos\delta - \sin \sum_{j=1}^{i} \bar{\eta}_j (e^{-\frac{i\delta_\eta^2}{2}} - 1) \sin\delta] \end{aligned} \tag{3.20}$$

Equation (3.20) is an explicit expression for the expectation of the cumulative error in the 2D Cartesian coordinate system, however, it still requires the ground-truth. In practice, the expectation is calculated by using the conditional measurements as follows:

$$E[\mu_t|\eta_i, d_i] = \begin{bmatrix} d_i(e^{-i\delta_\eta^2} - e^{\frac{-i\delta_\eta^2}{2}})(\sin \sum_{j=1}^{i} \eta_j \cos\delta + \cos \sum_{j=1}^{i} \eta_j \sin\delta) \\ d_i(e^{-i\delta_\eta^2} - e^{\frac{-i\delta_\eta^2}{2}})(\cos \sum_{j=1}^{i} \eta_j \cos\delta - \sin \sum_{j=1}^{i} \eta_j \sin\delta) \end{bmatrix} \tag{3.21}$$

And the cumulative error is eliminated as follows

$$\begin{bmatrix} \dot{X}_i \\ \dot{Y}_i \\ \dot{Z}_i \end{bmatrix} = \begin{bmatrix} d_i \sin(\sum_{j=1}^{i} \eta_j + \delta) - E[E(\dot{X}_i^e)|d_i, \eta_i] \\ d_i \cos(\sum_{j=1}^{i} \eta_j + \delta) - E[E(\dot{Y}_i^e)|d_i, \eta_i] \\ 0 \end{bmatrix}$$

Thus the PHD filter is proposed to estimate the locations.

3.1.3 Simulation

In this section, the performance of the proposed solution is demonstrated qualitatively and quantitatively. The proposed solution is evaluated by using the KITTI benchmark dataset [65]. The ground-truth is acquired by raw data from GPS/IMU, whereas the measurements are acquired by manually adding Gaussian noises. For GPS uncertainties, the distribution follows $\mathcal{N}(0, 30)$ in height and $\mathcal{N}(0, 0.000075^o)$ in both latitude and longitude, respectively. For gyroscope uncertainty, we have $\mathcal{N}(0, 0.002^o)$ in radian while the velocity error follows $\mathcal{N}(0, 0.15)$ in meters. As the PHD filter relies information from GPS, there is no kidnap problem during the whole phase. The proposed approach is implemented on a Core 2 Duo 3.0Ghz computer, with the average processing time 5ms per step.

Fig. 3.3 demonstrates the fusion results in four sequences. To better evaluate the proposed approach, a quantitative performance evaluation is given. As there is no requirement for data association, the estimated result from Kalman filter is also compared with the PHD filter. Although the PHD filter fuses data from heterogeneous sources, the cumulative error still affects the performance. It is observed that the proposed solution performs better than without considering the cumulative error.

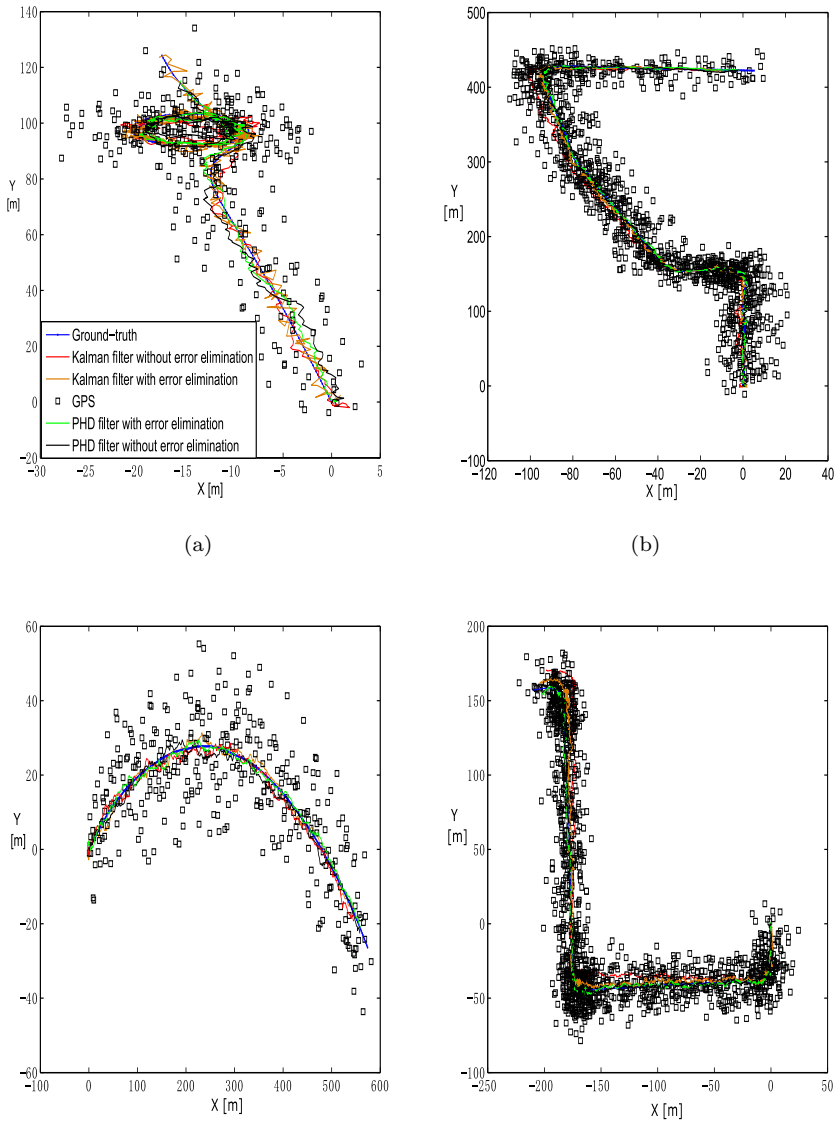


Figure 3.3: Localization performance in urban environments. Each figure shows a different scenario in complex urban environments.

Table 3.1: Performance of the single vehicle localization

Index	Distance	Frames	PHD filter without error elimination	PHD filter with error elimination	Kalman filter without error elimination	Kalman filter with error elimination
a	310m	340	3.5m	0.9m	3.3	0.9m
b	640m	1249	10m	6.5m	9.9m	6.4m
c	533m	355	3.2m	3.1m	3.1m	3.0m
d	408m	1424	11.1m	7.2m	10.9m	7.2m

Table 3.1 analyzes the proposed solution by calculating distance between the ground-truth and the estimated position. The index illustrates the sequences of experiments in Fig. 3.3 and the distance denotes the length of each scenario. As both the process model and the measurement model are linearly represented, the Kalman filter guarantees to provide the optimal filtering result. For PHD filter, instead of propagating the posterior density, it propagates the first-order statistical moment of the posterior density. Although the difference of both propagations is negligible, the PHD filter still achieves a suboptimal performance compared to the Kalman filter.

3.1.4 Summary

In this section, a single vehicle localization approach is proposed to process data from heterogeneous sensors. The benefits of the proposed approach are summarized as follows:

First and foremost, it provides a flexible and reliable framework to improve the localization performance. Second, it eliminates the influences from cumulative errors in large scale environments. Last but not least, the performance is guaranteed under the Bayes framework, in contrast to the Kalman filter.

Although the Kalman filter is considered the optimal solution to the Bayes filter for linear cases, it often suffers the data association challenge to confirm the measurement-to-target association. In this section, the task was focused on single vehicle localization. We have proven that the estimated result has comparable performance to the Kalman filter. In the next section, the PHD filter is utilized for multi-target, multi-source scenarios.

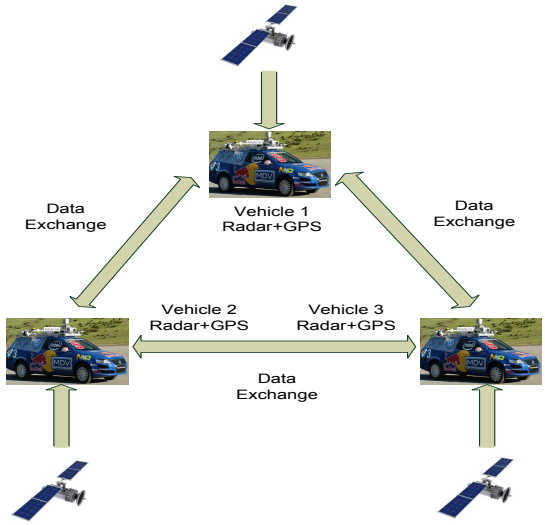


Figure 3.4: Multiple vehicle cooperative localization. Each vehicle is able to localize itself by using both internal sensors (e.g. GPS) and external sensors (e.g. radar).

3.2 Multiple Vehicle Cooperative Localization

In Sec. 3.1, the PHD filter is proposed to estimate the position in a scenario of single vehicle localization. With the improvement in the Car-2-Car (C2C) and Car-2-Infrastructure (C2I) techniques, methodologies using both internal and external sensors are being developed, called cooperative localization.

In cooperative localization, vehicles not only localize themselves but also assist in localizing the nearby vehicles. It has been concluded that cooperative localization achieves better performance than self-localization in terms of accuracy and coverage [66]. Essentially, this is due to the introduction of additional information to the system.

The aim of this section is to estimate the position of vehicles through the PHD filter by utilizing both the internal and external sensors.

3.2.1 Introduction

The concept of cooperative localization is illustrated in Fig. 3.4, by utilizing a dynamic heterogeneous network with following assumptions:

- Each vehicle is able to localize itself by using internal sensors in an absolute reference system. Here we assume that the measurements are given in a 2D global Cartesian coordinate system.
- Each vehicle is able to measure the relative position of the others by using external sensors. Here we assume that the measurements are given in a 2D local coordinate system in form of range and bearing.
- The communication network exchanges information among vehicles. Here we assume that the communication could be affected by delays or breaks in the network.

- The communication protocol cannot identify vehicles. Each vehicle only observes its surrounding and transmits the information to the others. It is possible to have miss detections (false positives).

Cooperative localization uses data from all the networks to estimate positions of all the vehicles. In practice, there are several challenges [67, 68, 69, 70]:

- The data association challenge.

One of the most important challenge is Data Association [71]. The development of C2C and C2I techniques supports vehicles in localizing and identifying other traffic participants correctly. However, in case of uncertain measurement-to-track association, it is a significant challenge to correctly estimate states. Also, measurements are often obscured or cluttered, further diluting the strength of association and increasing the localization uncertainty.

- The bandwidth challenge.

The bandwidth challenge was introduced in [72]. Typically, cooperative localization requires each node to transmit both the state and its covariance estimations. Systems with a high dimensional state (e.g. position, velocity, acceleration) and a high update rate (many times per second) present a significant amount of data. This in turn requires a high bandwidth multiplexed signal carrier. As the number of vehicles increases, the network is likely to become overloaded and thus unusable.

- The over-convergence challenge.

Over-convergence or rumor propagation was introduced in [73]. Each vehicle estimates the relative poses of the surrounding vehicles and broadcasts them to the network. In Fig. 3.4, vehicle 1 detects vehicle 2 first, the local pose on vehicle 2 can be fused with the observation from vehicle 1. Similarly, vehicle 3 can also be estimated from the observation on vehicle 2. However, the corresponding observation cannot be utilized by vehicle 1. This is due to the stochastic interdependence among vehicles, where the observation of vehicle 3 is partly from vehicle 1, called the over-convergence challenge.

Much work has been done for cooperative localization in both centralized and decentralized solutions [67, 73, 74, 75, 76]. In the centralized solution, vehicles are considered as a single system where positions are calculated based on the Bayes estimator. The communication bandwidth is satisfied. However, as the number of vehicles increases, the computation of data association grows exponentially. In decentralized solution, multiple fusion centers exist and each fusion center handles parts of the local information. However, it often exceeds the bandwidth load since both the states and covariances are transmitted. In addition, decentralized solution often suffers from the over-convergence problem.

It is still a challenge to address the aforementioned issues, either centralized solution or decentralized solution. In this section, the Probability Hypothesis Density (PHD) filter is proposed for cooperative localization. In PHD filter, the states are augmented as a set-valued state, whereas the measurements are augmented as a set-valued measurement. Modeling set-valued state and set-valued measurement provides a potential solution in multiple vehicle cooperative localization.

3.2.2 PHD Filter Implementation

The data flow of the proposed approach is introduced as follows:

Measurements from both internal and external sensors are collected as a set-valued measurement to recursively estimate the set-valued state.

- State Model

In this scenario, the state of each vehicle is represented as

$$\mathbf{x}_k = [p_{x,k}, p_{y,k}, \dot{p}_{x,k}, \dot{p}_{y,k}]^T \tag{3.22}$$

which contains the position $(p_{x,k}, p_{y,k})$ and velocity $(\dot{p}_{x,k}, \dot{p}_{y,k})$. And the set-valued state is represented as

$$X_k = \{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,N(k)}\} \tag{3.23}$$

where $N(k)$ represents the estimated amount of states at time k.

- Process Model

To predict the future state, the process model is utilized to represent the state translations.

$$\dot{\mathbf{x}}_k = f(\mathbf{x}_k, \mathbf{w}_k) \tag{3.24}$$

In general, the more complex the process model, the more precise the vehicle dynamics could be represented. In this scenario, the discrete time space is used.

$$\mathbf{x}_{k+1} = F_k \mathbf{x}_k + \mathbf{w}_k \tag{3.25}$$

where F_k and $Q_k = \text{cov}(\mathbf{w}_k)$ are represented as

$$F_k = \begin{bmatrix} \mathbf{I}_2 & \mathbf{I}_2 \\ \mathbf{0}_2 & \mathbf{I}_2 \end{bmatrix}, Q_k = \delta^2 \begin{bmatrix} \mathbf{I}_2/4 & \mathbf{I}_2/2 \\ \mathbf{0}_2 & \mathbf{I}_2 \end{bmatrix}, \tag{3.26}$$

in which $\mathbf{I}_2$ and $\mathbf{O}_2$ denote, the 2×2 identity and zero matrices, respectively. δ denotes the standard deviation of the process noise.

- Measurement Model

As previously discussed, measurements are collected from both internal and external sensors as a set-valued measurement, given by

$$Z_k = \{\mathbf{z}_k^1, \mathbf{z}_k^2, \cdots, \mathbf{z}_k^n\} \tag{3.27}$$

For each individual component, the measurement model is represented as

$$\mathbf{z}_k = H_k \mathbf{x}_k + \mathbf{v}_k \tag{3.28}$$

To map the state to the observation space, the measurement matrix is

$$H_k = [\mathbf{I}_2, \mathbf{0}_2] \tag{3.29}$$

It should be noted that $\mathbf{z}_k$ is acquired from both the internal sensors as

$$Z_k^1 = \begin{bmatrix} x \\ y \end{bmatrix} \tag{3.30}$$

and the external sensors as

$$Z_k^2 = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} r\cos\phi \\ r\sin\phi \end{bmatrix} \tag{3.31}$$

where r and ϕ denote the range and bearing, respectively.

- PHD Recursion

During the estimation process, the PHD filter is implemented as follows:

$$v_{k|k-1}(x_k) = \int P_{S,k}(x_{k-1}) f_{k|k-1}(x_k|x_{k-1}) v_{k-1}(x_{k-1}) \mathrm{d}x_{k-1} + \gamma_k(x_k) \tag{3.32}$$

$$v_k(x_k) = [1 - P_{D,k}(x_k)] v_{k|k-1}(x_k) + \sum\nolimits_{z \in Z_k} \frac{P_{D,k}(x_k) g_k(z|x_k) v_{k|k-1}(x_k)}{\kappa_k(z) + \int P_{D,k} g_k(z|\zeta) v_{k|k-1}(\zeta) \mathrm{d}\zeta} \tag{3.33}$$

where $P_{D,k}$ and $P_{S,k}$ denote the detection probability and the survival probability, respectively. Intensity function $\gamma_k(\,\cdot\,)$ describes the new born target, where the intensity $\kappa_k(\,\cdot\,)$ describes the clutter rate. It should be noted that x_k equals to the state $\mathbf{x}_k$ in this scenario.

As already stated, the PHD filter has no closed form solution. Hence in this scenario, the Gaussian Mixture Probability Hypothesis Density (GMPHD) filter is utilized since both the process and measurement models are linear.

- Implementation Issues

There are still issues for PHD implementation: the statistical properties of the internal and external sensors are inconsistent with respect to each other. To solve this issue, a converted measurement is proposed.

Measurements from both internal and external sensors are collected as Z_k^1 and Z_k^2. It is noted that Z_k^1 is solely acquired from the internal sensor, whereas Z_k^2 is represented as the combination of both the internal and external sensors. Meanwhile, it is assumed that the uncertainties in Z_k^1 and Z_k^2 are described as $\mathcal{N}(\mathbf{0}, \mathbf{R_1})$ and $\mathcal{N}(\mathbf{0}, \mathbf{R_2})$, respectively. Assuming cooperative localization contains n vehicles, the size of the measurement set Z_k^1 and Z_k^2 are calculated as n and $n(n-1)$, respectively.

At time $k-1$, the update process is represented as

$$v_{k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k-1}} \omega_{k-1}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{k-1}^{(i)}, P_{k-1}^{(i)}) \tag{3.34}$$

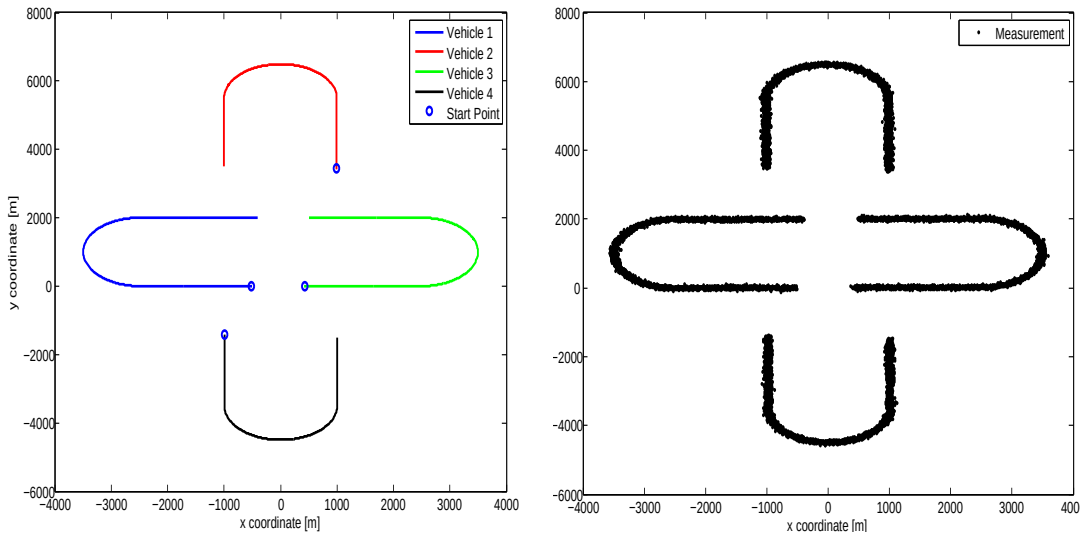
Using equation (Appendix.B.12), equation (Appendix.B.13) and equation (3.34), the update function $v_k^1(\mathbf{x})$ is calculated by using Z_k^1. Furthermore, since $v_{k-1}(\mathbf{x})$ is a Gaussian mixture, $v_k^1(\mathbf{x})$ is also a Gaussian mixture. With respect to Z_k^2, $v_k^1(\mathbf{x})$ is thus utilized as the predicted intensity function.

$$v_k(\mathbf{x}) = (1 - P_D) v_{k|k-1}^1(\mathbf{x}) + \sum_{z \in Z_k^2} v_{D,k}(\mathbf{x}; \mathbf{z}) \tag{3.35}$$

3.2.3 Simulation

In this section, the performance of the proposed solution is evaluated both quantitatively and qualitatively. We focus on demonstrating the ability of the proposed approach on dealing with all challenges including data association, communication bandwidth and over-convergence. The proposed approach is executed in Matlab on Duo CPU@3.0 Ghz, and the averaged processing time is close to 100 ms/frame.

During the simulation, both internal and external sensors are utilized to observe the surrounding environment. The measurement noise is also given by manually adding Gaussian white nosies. For GPS uncertainty, the distribution follows $\mathcal{N}(30, 30)$ in Cartesian coordinate



(a) (b)

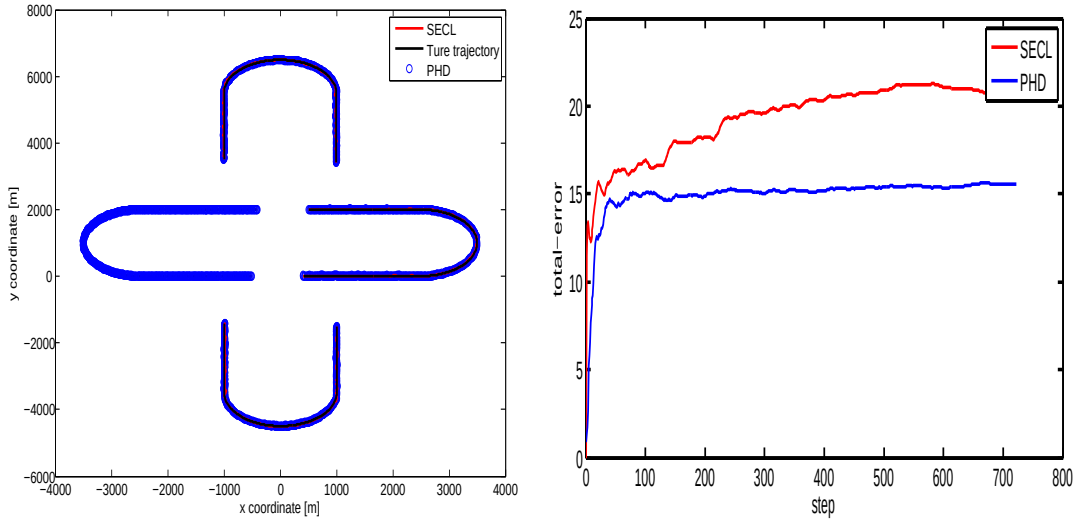
Figure 3.5: Ground-truth and the corresponding measurements. (a) Ground-truth of the vehicle. (b) Projected measurements from both internal and external sensors.

system. For radar uncertainty, the distribution follows $\mathcal{N}(10, 0.1)$ in polar coordinate system. As the PHD filter relies information from GPS, there is no kidnap problem during the whole phase. Meanwhile, information sharing is also available through the network. However, vehicles are not able to identify other vehicles in the environment. A comparative method using State Exchange Cooperative Localization (SECL) is also given, which mainly focuses on addressing the over-convergence issue among a group of communicating vehicles [73].

The ground-truth is represented in Fig. 3.5(a), and the measurements are shown in Fig. 3.5(b). Four measurements are used to represent one vehicle at each step. From these, only one is from Z_k^1 and the rest originate from Z_k^2. Figure 3.6(a) illustrates the estimated trajectories compared to the-state-of-the-art approach. The SECL estimation is exhibited in red lines, whereas the blue circles represent the PHD estimations.

In multiple vehicle cooperative localization, targets are often inconsistently observed, which is called missed objects. This is mainly due to two facts: on one hand, targets are measured in limited fields of view; on the other hand, the communication among networks may often temporarily unavailable. Hence the vehicles are not fully detected, whereas the performance of localization is influenced. The fresh and retired tracking objects also influence the performance, when targets are first detected and later disappeared in a long time. For traditional approaches, this is handled by the track management approaches to keep the whole trajectories. For PHD filter, it operates on the set space in format of multi-sensor-multi-target, meaning it is not able to provide a continuous track. Using different target birth models and system models, the PHD filter can successfully deal with the missed, fresh and retired objects during the whole process. Although it can estimate both the number of the targets and their states, the state identification is not provided. Therefore the estimated states are represented as individual circles in Figure 3.6(a), and the estimated number of states is represented in Figure 3.7.

To better analyze the performance, Fig. 3.6(b) calculates the Root Mean Square Error (RMSE) of the error in localization for each vehicle. As already stated, comparative results



(a) (b)

Figure 3.6: Evaluation of the proposed solution. (a) The estimated results with the SECL approach. (b) Error evaluation by using RMSE value.

for each vehicle are quite challenging, and a data association process is utilized to label the estimated states. The evaluation reference is then utilized to calculate the localization error by summing up the RMSE among vehicles:

$$\texttt{Error} = (\texttt{x}_{\texttt{est}} - \texttt{x}_{\texttt{true}})^2 + (\texttt{y}_{\texttt{est}} - \texttt{y}_{\texttt{true}})^2$$

$$\texttt{Total\_error} = \sqrt{\frac{\sum_{j=1}^{k} \sum_{i=1}^{4} \texttt{Error}_i^j}{k}}$$

where i and k denote the index of the vehicles and time, respectively.

Estimated errors in both the PHD or SECL techniques still exist when compared with the ground-truth. However, as time increases, the uncertainty of the PHD filter converges to the lower bound of the measurements. It is thus concluded that the overall performance of the PHD filter is better than the SECL approach.

Figure 3.7 exhibits the estimated number of vehicles. For PHD filter, since it operates on the set space, both the states and their numbers are recursively estimated. However, most approaches address this issue by assuming that some prior information was available (e.g. the structure of the topology is unchanged, the number of vehicles is given). Due to the measurement noises and the limited fields of view, the number of the vehicles is inconsistent estimated in Figure 3.7. In complex environments, the proposed approach is more flexible compared to the state-of-the-art. However, for fully autonomous driving, the continuity of trajectories (track management) should also be considered to improve the safety issue. It should be noted that track management is not the focus of the thesis and, therefore, has not been discussed here.

With respect to the bandwidth issue, the PHD filter still maintains superior performance. Assuming each measurement takes 2 bits (both internal and external sensors), the proposed approach requires at most $2n^2$ bits communication bandwidth for information sharing (there is

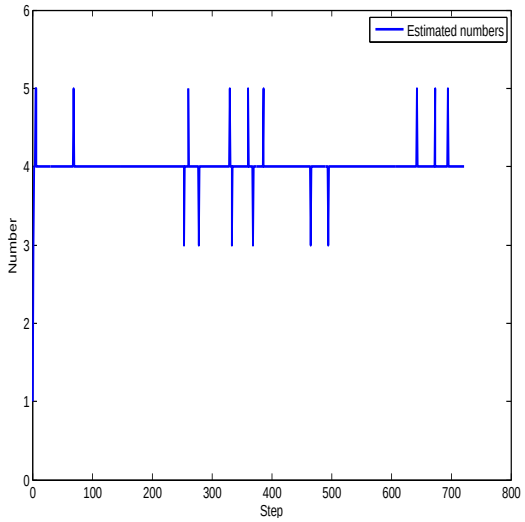


Figure 3.7: The estimated number of vehicles.

a total of n^2 measurements collected at each step, where n denotes the number of the vehicles). For the SECL approach, a huge communication bandwidth is required to transmit not only the states but also the covariances. For instance, each state is described as a random vector containing 4 components (position and velocity in 2D Cartesian coordinate system), in which 4 bits of communication bandwidth is required. Meanwhile, the covariance is described as a 4×4 matrix which takes 16 bits bandwidth. As a conclusion, it requires at least $20n^2$ bits bandwidth in contrast to the PHD filter (each vehicle is a fusion center which requires both the states and the corresponding covariances that occupying $20n$ bits. The total amount of the bandwidth requirement is thus calculated as $20n^2$ bits).

3.2.4 Summary

Cooperative localizing multiple vehicles using multiple sources of sensory data is a challenging application. The unknown data association, limited bandwidth communication and highly dynamic topology make the localization complex and often intractable. In this section, the PHD filter is proposed for multiple vehicle cooperative localization.

The primary benefit of the PHD filter is in addressing the challenge of data association. By utilizing the PHD filter, measurements are collected as set-valued measurements and used to recursively estimate the set-valued state.

Furthermore, the requirement of communication bandwidth is significantly reduced. The network only transmits the original measurements which results in minimal bandwidth consumption requirements.

Finally, the over-convergence challenge is addressed. Since the PHD filter operates on the set space, the set-valued state is thus recursively estimated where the circular situation no longer exist and common information is not double-counted across disparate nodes.

3.3 Multiple Vehicle Cooperative Localization with Sensor Bias Elimination

In Sec. 3.2, the Probability Hypothesis Density (PHD) filter is proposed to localize a group of vehicles. This section also extends the PHD filter for cooperative localization by jointly estimating sensor biases and vehicle states.

For our scenarios, two major kind of errors can exist: one belongs to the random noise which is often considered to be normally distributed; the other is represented as a bias (systematic error) in the system. Spatial registration is thus introduced to eliminate the bias before the filtering phase. Traditional solutions have already been employed in [77, 78, 79, 80, 81]. For multiple vehicle cooperative localization, it is still not fully tackled with the challenges mentioned in Sec. 3.2.

The aim of this section is to provide a framework for jointly estimating both the positions of vehicles and the biases of sensors through the PHD filter. The Sequential Monte Carlo (SMC) method is utilized to implement the PHD filter in the non-linear and non-Gaussian conditions [82].

3.3.1 Introduction

The description of multiple vehicle cooperative localization has already been introduced summarizing when operating in a cooperative network, vehicles are able to localize themselves more precisely. In this section, we extend the scenario by also estimating biases from external sensors.

As illustrated in Fig. 3.8, the projected measurements are exhibited as ghost targets in contrast to the ground-truth. To achieve a precise estimation, influences from biases should be eliminated before the filtering phase.

Methodologies have already been introduced for cooperative localization, mainly described as centralized and decentralized solutions. Centralized solutions have benefits for large bandwidth communication and limits the data association challenge. Decentralized solutions are advantageous for data association, but suffer from double-counting common information leading to over-convergence. To the best of our knowledge, neither centralized nor decentralized approaches jointly estimate the position and the sensor biases in cooperative localization.

The goal of this section is to take spatial registration into account in cooperative localization. By adopting the SMC PHD filter, both the states and the biases are jointly estimated in scenarios of data association, communication bandwidth and over-convergence challenges.

3.3.2 PHD Filter Implementation

As before, measurements from both internal and external sensors are collected as a set-valued measurement to estimate the set-valued state.

- State Model

The state is defined as a vector of random variables comprised of the position and velocity. To estimate the biases, the state model should be augmented with additional variables.

Let

$$\beta_k = [(\beta_k^1)^T, \ldots, (\beta_k^L)^T]^T$$

denotes the biases from the external sensors, the posterior density function is defined as

$$P(X_k, \beta_k | Z_{1:k})$$

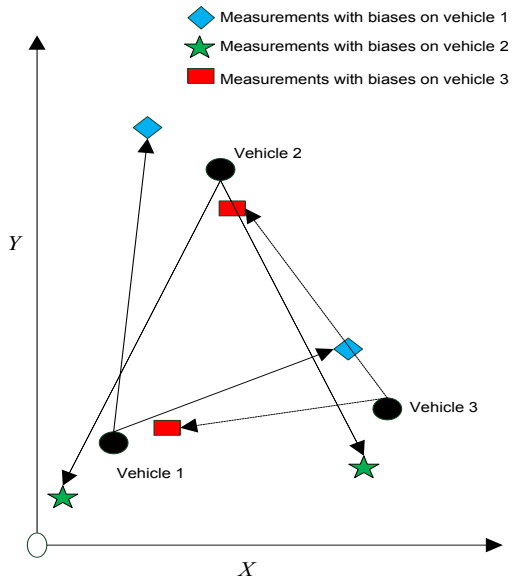


Figure 3.8: Cooperative localization with biases on external sensors. It is observed that the precision of the transformed measurements are strongly influenced by biases.

where $Z_{1:k}$ denotes the measurement sets received up to time k, X_k represented the set-valued state which is defined in Eq. (3.23). By deriving the PHD filter for bias estimation, the new set-valued state is augmented as

$$\mathbf{y}_k = (X_k, \beta_k) \tag{3.36}$$

- Process Model

The process model is represented as Markov process as follows

$$x_{k+1} = F_k x_k + w_k \tag{3.37}$$

where F_k and w_k denote the process matrix and the process noise, respectively. More details about the process model are introduced during the implementation process.

- Measurement Model

In contrast to the measurement model in Eq. (3.28), the projected measurements are affected by both the random noises and the systematic errors (biases). Assuming Z_k collects all measurements at time k

$$Z_k = \{z_k^1, z_k^2, \cdots, z_k^n\} \tag{3.38}$$

The measurement model for each singe target is expressed as:

$$z_k^l = h_k^l(x_k) + \beta_k^l + v_k^l \tag{3.39}$$

where h_k^l denotes the nonlinear transformation of sensor l, β_k^l and v_k^l denote the bias and the random noise.

- PHD Recursion

The PHD recursion is given by

$$v_{k|k-1}(\mathbf{y}_k) = \int P_{S,k}(\mathbf{y}_{k-1}) f_{k|k-1}(\mathbf{y}_k|\mathbf{y}_{k-1}) v_{k-1}(\mathbf{y}_{k-1}) \mathrm{d}\mathbf{y}_{k-1} + \gamma_k(\mathbf{y}_k) \tag{3.40}$$

$$v_k(\mathbf{y}_k) = [1 - P_{D,k}(\mathbf{y}_k)] v_{k|k-1}(\mathbf{y}_k) + \sum\nolimits_{z \in Z_k} \frac{P_{D,k}(\mathbf{y}_k) g_k(z|\mathbf{y}_k) v_{k|k-1}(\mathbf{y}_k)}{\kappa_k(z) + \int P_{D,k} g_k(z|\zeta) v_{k|k-1}(\zeta) \mathrm{d}\zeta} \tag{3.41}$$

Assuming β_k is independent of the original state x_k, the following equations (the birth intensity, the spawn intensities, the transition density and the survival probability) are represented as:

$$\begin{aligned} \gamma_k(\mathbf{y}_k) &= \gamma_k(x_k) + \gamma_k(\beta_k) \\ \rho(\mathbf{y}_k|\mathbf{y}_{k-1}) &= \rho(x_k|x_{k-1}) + \rho(\beta_k|\beta_{k-1}) \\ f_{k|k-1}(\mathbf{y}_k|\mathbf{y}_{k-1}) &= f_{x,k|k-1}(x_k|x_{k-1}) f_{\beta,k|k-1}(\beta_k|\beta_{k-1}) \\ P_{S,k}(\mathbf{y}_{k-1}) &= P_{S,k}(x_{k-1}) P_{S,k}(\beta_{k-1}) \end{aligned} \tag{3.42}$$

By assuming the biases are constant values, the birth intensity, the spawn intensity and the survival probability are given as

$$\gamma_k(\beta_k) = 0, \quad \rho(\beta_k|\beta_{k-1}) = 0, \quad P_{S,k}(\beta_{k-1}) = 1 \tag{3.43}$$

where the PHD prediction is

$$\begin{aligned} & v_{k|k-1}(\mathbf{y}_k) = v_{k|k-1}(x_k, \beta_k | Z_{1:k-1}^{l:L}) \\ &= \int [P_{S,k}(x_{k-1}) f_{x,k|k-1}(x_k|x_{k-1}) f_{\beta,k|k-1}(\beta_k|\beta_{k-1}) + \rho(x_k|x_{k-1})] \\ & \quad \cdot v_{k-1}(x_{k-1}, \beta_{k-1}|Z_{1:k-1}^{l:L}) \mathrm{d}x_{k-1} \mathrm{d}\beta_{k-1} + \gamma_k(x_k) \end{aligned} \tag{3.44}$$

The update equation is acquired by

$$v_{k|k}(\mathbf{y}_k) = v_{k|k}(x_k, \beta_k|Z_{1:k}^{l:L}) = G_k^1(Z_k^1|x_k, \beta_k^1) \cdots G_k^L(Z_k^L|x_k, \beta_k^L) \cdot v_{k|k-1}(x_k, \beta_k|Z_{1:k-1}^{l:L}) \tag{3.45}$$

with

$$\begin{aligned} & G_k^l(Z_k^l|x_k, \beta_k^l) = 1 - P_{D,k}^l(x_k, \beta_k^l) \\ & + \sum\nolimits_{z_k^l \in Z_k{}^l} \frac{P_{D,k}^l(x_k, \beta_k^l) g_k^l(z_k^l|x_k, \beta_k^l)}{\kappa_k^l(z_k^l) + \int P_{D,k}^l(x_k, \beta_k^l) g_k^l(z_k^l|x_k, \beta_k^l) v_{k|k-1}(x_k, \beta_k|Z_{1:k-1}^{l:L}) dx_k \mathrm{d}\beta_k} \end{aligned} \tag{3.46}$$

The expected number of vehicles is estimated by integrating the proposed PHD recursion on the whole region:

$$\hat{N}_{k|k} = \int v_{k|k}(\mathbf{y}_k) d\mathbf{y}_k = \int v_{k|k}(x_k, \beta_k|Z_{1:k}^{l:L}) \mathrm{d}x_k \mathrm{d}\beta_k \tag{3.47}$$

where the bias is derived by

$$\hat{\beta}_{k|k} = \frac{\int_S \beta_k v_{k|k}(x_k, \beta_k|Z_{1:k}^{l:L}) \mathrm{d}x_k \mathrm{d}\beta_k}{\hat{N}_{k|k}} \tag{3.48}$$

As already shown, the PHD recursion has multiple integrals where a closed form implementation is still missing. In this section, the Sequential Monte Carlo (SMC) method is utilized to jointly estimate the states and the biases due to the nonlinear issues. More details of the SMCPHD implementation could be found in Appendix. B.

- Implementation Issues

In process model, the original state $\mathbf{x}_k = [p_{x,k}, p_{y,k}, \dot{p}_{x,k}, \dot{p}_{y,k}]^T$ is utilized from Sec. 3.2 and the transition probability density is modeled as

$$f_{k|k-1}(\mathbf{x}|\zeta) = \mathcal{N}(\mathbf{x}; F_{k-1}\zeta, Q_{k-1}) \tag{3.49}$$

where Q_k and F_k are given by the constant velocity model with

$$F_k = \begin{bmatrix} \mathbf{I}_2 & \mathbf{I}_2 \\ \mathbf{0}_2 & \mathbf{I}_2 \end{bmatrix}, Q_k = \delta^2 \begin{bmatrix} \mathbf{I}_2/4 & \mathbf{I}_2/2 \\ \mathbf{0}_2 & \mathbf{I}_2 \end{bmatrix}, \tag{3.50}$$

$\mathbf{I}_n$ and $\mathbf{O}_n$ denote the $n \times n$ identity and zero matrices, respectively. δ is the standard deviation of the process noise.

Furthermore, the bias is comprised of the range and orientation as

$$\beta_k^l = [\Delta\rho_k^l, \Delta\theta_k^l]^T \tag{3.51}$$

and the random noise is

$$v_k^l = [\delta\rho_k^l, \delta\theta_k^l]^T \tag{3.52}$$

Assuming biases do not drift during the whole process, the dynamic model is thus described as a Gauss-Markov process with the transition density

$$f_{\beta,k|k-1}(\beta_k|\beta_{k-1}) = \mathcal{N}(\beta_k^1|\beta_{k-1}^1, B_{k-1}^1) \cdots \times \mathcal{N}(\beta_k^L|\beta_{k-1}^L, B_{k-1}^L)$$

where B_k^l denotes covariance for the bias β_k^l.

Similar to Sec. 3.2, measurements from both internal and external sensors are projected to the ground plane. Suppose a number of L vehicles are monitored over the whole space, a number of $L(L-1)$ observations are collected at each step.

3.3.3 Simulation

This section aims to demonstrate the performance of the proposed approach by assuming that the external sensors have biases in both range and orientation.

The proposed approach was evaluated over $[-6000, 6000] \times [-6000, 6000]$ m^2, where the measurements are acquired by manually adding Gaussian random noises. During the simulation, biases are given as follows: $\beta^1 = [-25\text{m}, 75\text{mrad}]^T, \beta^2 = [55\text{m}, -60\text{mrad}]^T, \beta^3 = [-40\text{m}, 25\text{mrad}]^T, \beta^4 = [35\text{m}, -45\text{mrad}]^T$. For random noise, it is distributed as i.i.d. with zero mean, covariance $\mathrm{R}_{GPS} = \mathrm{diag}[15\text{m}^2, 15\text{m}^2]$ for internal sensors, and zero mean, $\mathrm{R}_{Radar} = \mathrm{diag}[5\text{m}^2, 5\text{mrad}^2]$ for external sensors. The proposed approach is executed in Matlab on Duo CPU@3.0 Ghz, and the processing time is averaged as 2.9 second/step for the PHD filter.

To evaluate the proposed solution qualitatively as well as quantitatively, the Gaussian Mixture PHD filter is also utilized which has already been exhibited in Sec. 3.2. At this stage it is important to note that for the GMPHD filter, the spatial registration issue has not been considered yet.

Fig. 3.9 exhibits the ground-truth, the GMPHD estimation and the SMCPHD estimation. The solid line denotes the ground-truth, the symbol '+' and '$\circ$' denotes the estimation of the SMCPHD filter and the GMPHD filter, respectively. Since the PHD filter operates on the set space, both the states and their cardinalities are jointly estimated whereas the identification of a single track is quite challenging. The GMPHD filter treats the uncertainties as random noises, whereas the SMCPHD filter treats the uncertainties as the combination of both the biases and

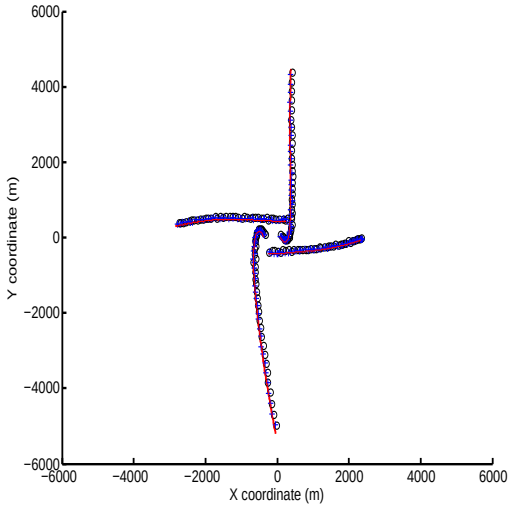


Figure 3.9: Multiple vehicle cooperative localization based on the probability hypothesis density filter. Symbol '+' denotes the estimation of the SMCPHD filter and the symbol '∘' denotes the corresponding result from the GMPHD filter.

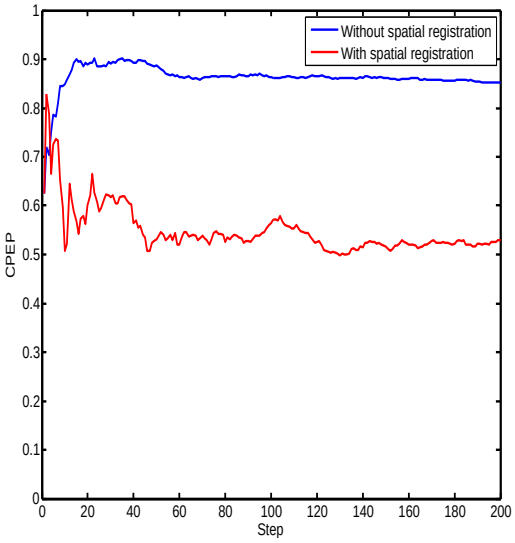


Figure 3.10: Performance of the Circular Position Error Probability (CPEP) against time

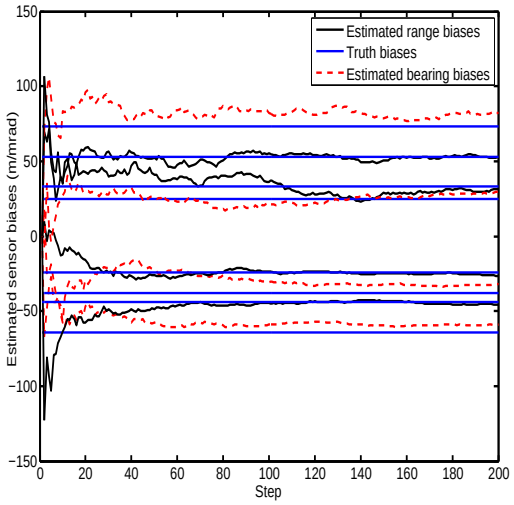


Figure 3.11: Estimated biases from external sensors

the random noises, respectively. By taking spatial registration into account, the estimations of the SMCPHD filter are more close to the ground-truth, in contrast to the GMPHD filter.

The Circular Position Error Probability (CPEP) is also utilized as a benchmark to analyze the performance of the proposed approach. Given a set of true values

$$\mathcal{X}_k = \{\mathbf{x}_k^l\}_{l=1}^{L_k} \tag{3.53}$$

and the corresponding estimations

$$\hat{\mathcal{X}}_{k|k} = \{\hat{\mathbf{x}}_{k|k}^l\}_{l=1}^{\hat{L}_{k|k}} \tag{3.54}$$

The CPEP is calculated by

$$\text{CPEP}_k(r) = \frac{1}{L_k} \sum_{\mathbf{x}_k \in \mathcal{X}_k} \text{Prob}\{||H_k\hat{\mathbf{x}}_{k|k} - H_k\mathbf{x}_k||_2 > r, \quad \forall \hat{\mathbf{x}}_{k|k} \in \hat{\mathcal{X}}_{k|k}\} \tag{3.55}$$

where $H_k\hat{\mathbf{x}}_{k|k}$ and $H_k\mathbf{x}_k$ denote the estimated trajectories and the ground-truths, respectively. $r = 20$ m is the given radius. In general, the CPEP value becomes smaller when the filter achieves a more precise estimation.

Fig. 3.10 illustrates the estimation of the CPEP during the whole process. Since the SMCPHD filter also considers the biases, the corresponding CPEP value is smaller than the GMPHD filter.

Fig. 3.11 also exhibits the estimated biases in contrast to the ground-truth on both bearing and orientation of vehicles. It is observed that the estimated biases converged to the ground-truth during the filtering phase. Also, the estimations are inconsistent with the ground-truth, which may be due to a number of issues (e.g. delay, clutter, random noise etc.).

Fig. 3.12 demonstrates the estimated amounts of vehicles by both the SMCPHD filter and the GMPHD filter, respectively. At beginning, the estimations are close to each other. With time increases, estimation from the SMCPHD filter is more close to the ground-truth by eliminating sensor biases.

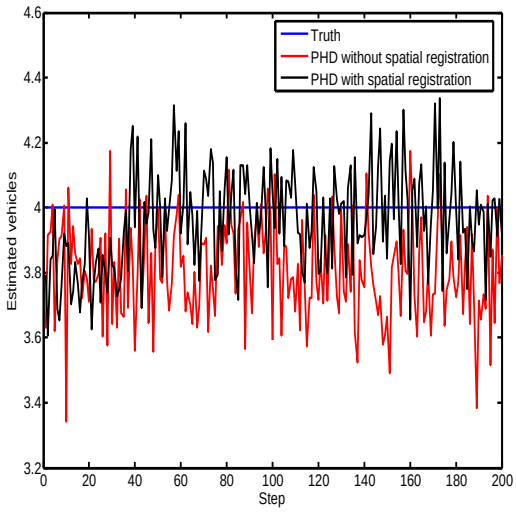


Figure 3.12: The estimated number of vehicles.

3.3.4 Summary

For multiple vehicle cooperative localization, the potential of using the PHD filter has already been demonstrated in complex environments. In this scenario presented herewith, the PHD filter is extended by also taking spatial registration into account. The advantages of the proposed approach are summarized as follows:

First, it works under complex situations which often exist in practice. Due to the limited information, both the data association and the number of vehicles are unknown. Also, sensors may include biases which make it difficult to calculate a bias free state estimation.

Second, the communication network only transmits the original measurements which take the minimal requirement.

Third, it is also flexible in highly dynamic environments. The locations are recursively estimated, which the over-convergence challenge addresses.

3.4 Conclusion

This chapter provides an insight into the Probability Hypothesis Density (PHD) filter in applications of localization. An implementation process is constructed off-line, by defining the state model, process model and the measurement model. And frame-by-frame estimation is performed by the proposed approach to significantly improve the performance of localization.

Experimental results over complex urban scenarios show that our proposed approach deals fairly well with the outlined challenges including data association, sensor biases, communication bandwidth and so on. In single-target environments, the proposed solution achieves almost the same performance as the Kalman filter. In next chapter, the proposed approach is employed to the visual odometry applications.

Chapter 4

Visual Odometry

In Chapter 3, the Probability Hypothesis Density (PHD) filter has been introduced as a framework for efficient and robust localization in both single and multi-target scenarios. Traditional Bayes filtering (e.g. the Kalman Filter) provides an optimal solution, but suffers from the track-measurement association challenge. The PHD filter operates on the set space in which this data association challenge is avoided.

In this chapter, we focus on tackling issues of visual odometry, so as to provide reliable pose information of the ego-vehicle. Challenges are addressed by employing the PHD filter to extract and track features in both monocular camera and stereo camera scenarios.

The remainder of this chapter is organized as follows: Section 4.1 gives a brief introduction to visual odometry. In Section 4.2 and Section 4.3, we give more details to single camera visual odometry and stereo camera visual odometry respectively. Finally, Section 4.4 concludes the chapter.

4.1 Introduction

Accurate localization plays an important role in Intelligent Transportation Systems (ITS). Events such as the Defense Advanced Research Projects Agency (DARPA) Grand Challenge and the Urban Challenge have significantly increased the research interests in autonomous vehicles. Numbers of applications are being developed mainly relying on the locations of vehicles, in which GPS plays an important role for the localization. However, GPS has a number of limitations as it relies on good visibility of the sky and is susceptible to radio frequency interference. Methods of using camera as the primary sensor for localization are thus investigated, called visual odometry. Assuming the transformation between camera and environment is known, the trajectory of the ego-vehicle is calculated by searching the changes in consecutive frames.

Reasons for choosing cameras are summarized as follows: cameras are not only cheaper but also provide a more reliable result in contrast to the Inertial Measurement Units (IMUs) [83]. Furthermore, cameras are utilized without any prior knowledge of the environment. In past few years, a large number of approaches have been developed which could be mainly divided into optical flow, and feature matching.

Optical flow (also called dense algorithm) focuses on the changes in brightness of the image, which originate from the apparent motion in consecutive images [84] [85]. Optical flow is computationally cheaper compared to the feature matching, but has a lower precision. Corke et al.[86] and Benoit et al. [87] claimed that optical flow could not be widely used due to the difficulties of estimating the dense structure in unknown environments.

Feature matching (also called structure-from-motion) appears to be a current trend in research communities. By matching features in consecutive frames, the displacement (also called ego-motion vector) between associated features is calculated. The movement of the camera is thus estimated by accumulating the corresponding displacements during the whole process [88]. The basic sketch of feature matching was introduced by Moravec [89] broadly as follows:

- Feature Detection

The FAST feature, considered as the most popular detector, is utilized due to the high speed with respect to an low computational cost [90]. Additionally, the Scale Invariant Feature Transform (SIFT) algorithm is also utilized to extract SIFT feature, which is invariant to uniform scaling, orientation, and partially invariant to affine distortion and illumination [91]. Similar to the SIFT feature, the Speed Up Robust Features (SURF) algorithm is proposed regarding to its fast approximations compared to the SIFT algorithm [92]. In addition to these techniques, there are other algorithms for extracting features which are often used in computer vision applications, e.g. Harris corner detector [93] and Kanade-Lucas-Tomasi detector (KLT) [94].

- Feature Matching

In matching process, features are collected between consecutive frames where data association plays an important role. By translating associated features from the image coordinate system to the camera coordinate system, the ego-motion vector is calculated. However, in practice, the performance is often affected by outliers since a perfect data association with no user-input can be extremely challenging. Therefore, eliminating the falsely associated pairs is quite important and is done using Random Sample Consensus (RANSAC) algorithm [95]. By generating a model from randomly selected minimal set of data, the RANSAC is utilized to remove outliers and the remaining features are used to estimate the motion hypothesis.

- Motion Estimation

Motion estimation is used to encode the poses. Due to the nonlinear issues, solutions are mainly proposed by using least-square method [89], Sparse Bundle Adjustment (SBA) method [96] and the Gauss-Newton minimization method [97]. Furthermore, traditional Bayes filters are also used in forms of the Kalman filter, the extended Kalman filter and the particle filter [98].

In general, visual odometry consists of the aforementioned procedures by using structure-from-motion approaches. However, in urban environments, tracking huge numbers of features is quite challenging because:

1. Outliers may exist in consecutive frames.

   In the matching process, falsely associated features should be eliminated before the estimation phase. Most algorithms use RANSAC to treat mismatches as outliers. However, such elimination may also have inconsistent performances with respect to different detectors. An efficient matching method is therefore required to eliminate outliers among features.

2. Unevenly distributed features may influence the performance.

   Features are expected to be uniformly distributed over the whole space. However, small regions may often contain large number of features. Assuming each feature contributes the same importance during the estimation phase, the aggregated features may introduce biases. An efficient extracting method is thus required to detect features.

3. Extracted features from moving objects may influence the performance.

   The ego-motion vector is calculated by using features originated from stationary objects. However, urban environments often contain a huge amount of features from moving objects, as the ego-motion vector is imprecisely estimated. A method is thus required to eliminate the influences from moving objects.

In the next sections, the PHD filter is proposed to solve the aforementioned issues in applications of both single camera and stereo camera visual odometry.

4.2 Single Camera Visual Odometry

For visual odometry, much work has been completed by utilizing a single camera (e.g. [99] [100]), stereo-camera pairs (e.g.[101, 102, 103]) and even omni-directional cameras (e.g. [104]). Assuming cameras have been well calibrated, the trajectory is reconstructed based on the homogeneous transformation. For feature matching approaches, the primary task is to estimate the displacement from the associated features. Data association methodologies are thus applied to reduce influences from the outliers. By assuming a statistical model-fitting task, the RANSAC approach is utilized to find the optimal model to explain the corresponding data. Although it eliminates a huge amount of outliers, challenges remain. For example, when features are aggregated from moving objects, correct measurements may be treated as outliers.

The PHD filter, based on the Random Finite Set (RFS) statistic is proposed as a solution for this. Instead of tracking individual features, set-valued measurements are utilized to recursively estimate the set-valued state. As a result, the association between measurements becomes part of the state which the filter estimates, thus avoiding the matching process. The extraction phase starts by searching and reconstructing features and later the estimation phase estimates the ego-motion vector. Once the ego-motion vector has been computed, dead-reckoning is used to calculate the trajectory.

4.2.1 Feature Extraction

The goal of this section is to collect interesting features as the set-valued measurement in the vehicle coordinate system.

- **Points-of-interest Detection**

In the proposed approach, the Scale Invariant Feature Transform (SIFT) feature is extracted, which is invariant to image translation, rotation, scaling and illumination [91]. For visual odometry, the SIFT feature is selected since the image sequences are acquired from different fields of view.

In Fig. 4.1, the SIFT features are uniformly distributed in the urban environment. Most approaches associate the nearest features in the matching process; however, the high dimensionality makes the association exhaustively inefficient. In this scenario, the SIFT features are only collected as a set-valued measurement in the Cartesian coordinate system.

- **Transformation**

Once the SIFT features have been extracted, the corresponding coordinates are calculated. The following step is a basic description of utilizing linearized transformation in single camera visual odometry.

Figure 4.1: Scale Invariant Feature Transform (SIFT) features in urban environments

Fig. 4.2 demonstrates the transformations in spatial coordinates. It is observed that the camera is placed at height h, with down/tilt angle ϕ oriented towards the ground plane. It should be noted that $(x, y, z)^T$ denotes the coordinate in the vehicle coordinate system, and the position of camera is represented as $(0, 0, h)^T$.

Assuming the camera has already been calibrated to eliminate the skew and distortion issues [105, 106], the linear transformation between the vehicle coordinate and image coordinate system is calculated by utilizing a homogeneous transformation as:

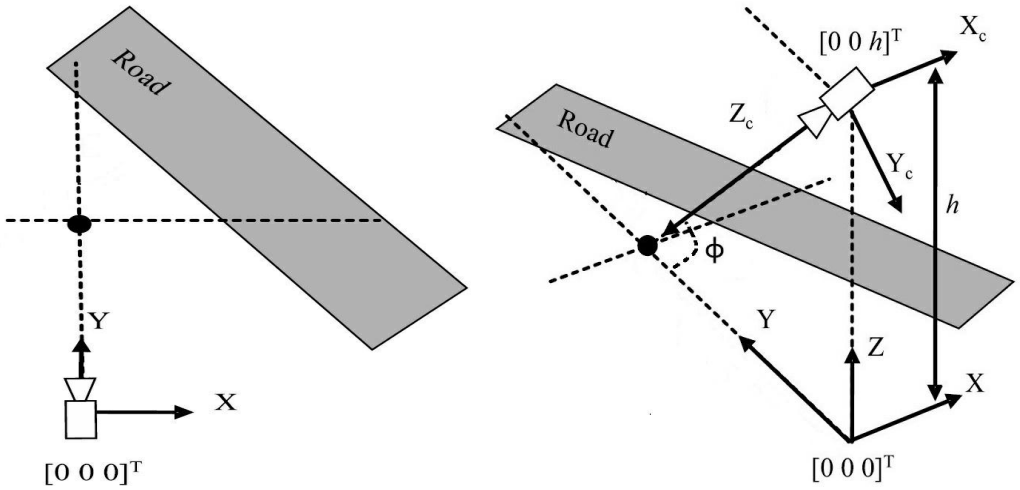
$$\begin{bmatrix} uw \\ vw \\ w \end{bmatrix} = K \cdot R \cdot [I_{3\times3}| - T] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{4.1}$$

with

$$K = \begin{bmatrix} f & 0 & u_c \\ 0 & f & v_c \\ 0 & 0 & 1 \end{bmatrix}, R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -\sin\phi & -\cos\phi \\ 0 & \cos\phi & -\sin\phi \end{bmatrix}$$

where K and R denotes the camera calibration matrix and the rotation matrix, respectively. ϕ represents the angle with respect to the X axis, $I_{3\times3}$ defines the three dimension identity matrix and $T = [0, 0, h]^T$ describes the translation vector between the camera and the vehicle coordinate systems. The coordinate of the principal point is represented as (u_c, v_c) in the image plane and w is the scale factor.

Assuming all features are on the ground plane, the corresponding coordinates are represented



(a) Top view of the scene

(b) Same scene viewed from the left side

Figure 4.2: Coordinate transformation in case of single camera visual odometry.

as

$$u = \frac{wu}{w} = \frac{fx}{y\cos\phi + h\sin\phi} + u_c \tag{4.2}$$

$$v = \frac{wv}{w} = \frac{fh\cos\phi - fy\sin\phi}{y\cos\phi + h\sin\phi} + v_c \tag{4.3}$$

Rearranging the above equations, we have

$$x = \frac{(u-u_c)(hf\cos\phi + fh\tan\phi\sin\phi) + 2hu_c\sin\phi(v_c - v)}{f(v + f\tan\phi - v_c)} \tag{4.4}$$

$$y = \frac{h(f - v\tan\phi + v_c\tan\phi)}{v + f\tan\phi - v_c} \tag{4.5}$$

Thus the features are transformed from the image coordinate system to the vehicle coordinate system.

4.2.2 PHD Filter Implementation

In the proposed solution, the ego-motion vector is considered as an extended target where the extracted features are considered as the scattering measurements.

Based on the physical characteristics, the displacements between consecutive frames should consistent among the associated features. Visual odometry is thus achieved by calculating the average movements of the set-valued state.

- State Model

To estimate the ego-motion vector, the state is required to represent the pose in the vehicle coordinate system. In this scenario, each individual state is defined as

$$\mathbf{x}_k = [x_k, y_k, \beta_k, \dot{x}_k, \dot{y}_k]^T \tag{4.6}$$

which contains the position (x_k, y_k), velocity $(\dot{x}_k, \dot{y}_k)$. Rotation β_k (β_k denotes the changes in orientation during the interval $[k-1, k]$.)

- Process Model

According to the Euler rotation theorem, the process model is proposed to predict the future poses. Suppose the previous position at step k is known, the new position at step $k+1$ is acquired as:

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} \cos\Delta\beta_k & -\sin\Delta\beta_k \\ \sin\Delta\beta_k & \cos\Delta\beta_k \end{bmatrix} \begin{bmatrix} x_k - \Delta x_k \\ y_k - \Delta y_k \end{bmatrix} \tag{4.7}$$

where $(\Delta x_k, \Delta y_k)$ and $\Delta\beta_k$ denote the translation and rotation respectively. Furthermore, the displacement of the camera is equal to the movement of the associated features.

$$(\Delta x_k, \Delta y_k, \Delta\beta_k) = [\dot{x}_k, \dot{y}_k, \beta_k]$$

Thus the process model is given by:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_k \tag{4.8}$$

where

$$\mathbf{F} = \begin{bmatrix} \cos\beta_k & -\sin\beta_k & 0 & -\cos\beta_k & \sin\beta_k \\ \sin\beta_k & \cos\beta_k & 0 & -\sin\beta_k & -\cos\beta_k \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and $\mathbf{w}_k$ denotes the process noise.

- Measurement Model

Since the coordinates are represented in the vehicle coordinate system, the measurement model is given by

$$\mathbf{z}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k \tag{4.9}$$

where

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{4.10}$$

and $\mathbf{v}_k$ denotes the measurement noise.

- Ego-motion Estimation

As the displacements of the associated features return the same values, the ego-motion vector is calculated by averaging the translation and rotation variables in the set-valued state:

$$\mu_{\mathbf{k}} = [\Delta x_k, \Delta y_k, \Delta\beta_k]^T \tag{4.11}$$

with

$$\Delta x_k = \frac{1}{N(k)} \sum_{i=1}^{N(k)} \dot{x}_k^i,$$

$$\Delta y_k = \frac{1}{N(k)} \sum_{i=1}^{N(k)} \dot{y}_k^i,$$

$$\Delta \beta_k = \frac{1}{N(k)} \sum_{i=1}^{N(k)} \beta_k^i$$

where $N(k)$ denotes the estimated amount of the states.

- PHD Recursion

Once the corresponding models are confirmed, the PHD filter is utilized to estimate the set-valued state with the following equations:

$$v_{k|k-1}(x_k) = \int P_{S,k}(x_{k-1}) f_{k|k-1}(x_k|x_{k-1}) v_{k-1}(x_{k-1}) \mathrm{d}x_{k-1} + \gamma_k(x_k)$$

$$v_k(x_k) = [1 - P_{D,k}(x_k)] v_{k|k-1}(x_k) + \sum\nolimits_{z \in Z_k} \frac{P_{D,k}(x_k) g_k(z|x_k) v_{k|k-1}(x_k)}{\kappa_k(z) + \int P_{D,k} g_k(z|\zeta) v_{k|k-1}(\zeta) \mathrm{d}\zeta}$$

where x_k denotes the corresponding state of each target, same to the state $\mathbf{x}_k$ in this scenario.

The PHD filter has no closed form. Since both the process and the measurement models are linearly described, the Gaussian Mixture (GM) implementation is utilized. Further details of the implementation process could be found in Appendix. B.

- Implementation Issues

If features fall into small regions, it influences the precision of the ego-motion vector. To solve this problem, those aggregated features are considered as a single unit to ensure the uniform distribution. Given a threshold, the corresponding states are automatically merged into a single state. This calculation is based on the Mahalanobis distance given as:

$$\mathbf{x}_k^j := \{i : (\mathbf{x}_k^i - \mathbf{x}_k^j)^T (P_k^i)^{-1} (\mathbf{x}_k^i - \mathbf{x}_k^j) \leq \tau\} \tag{4.12}$$

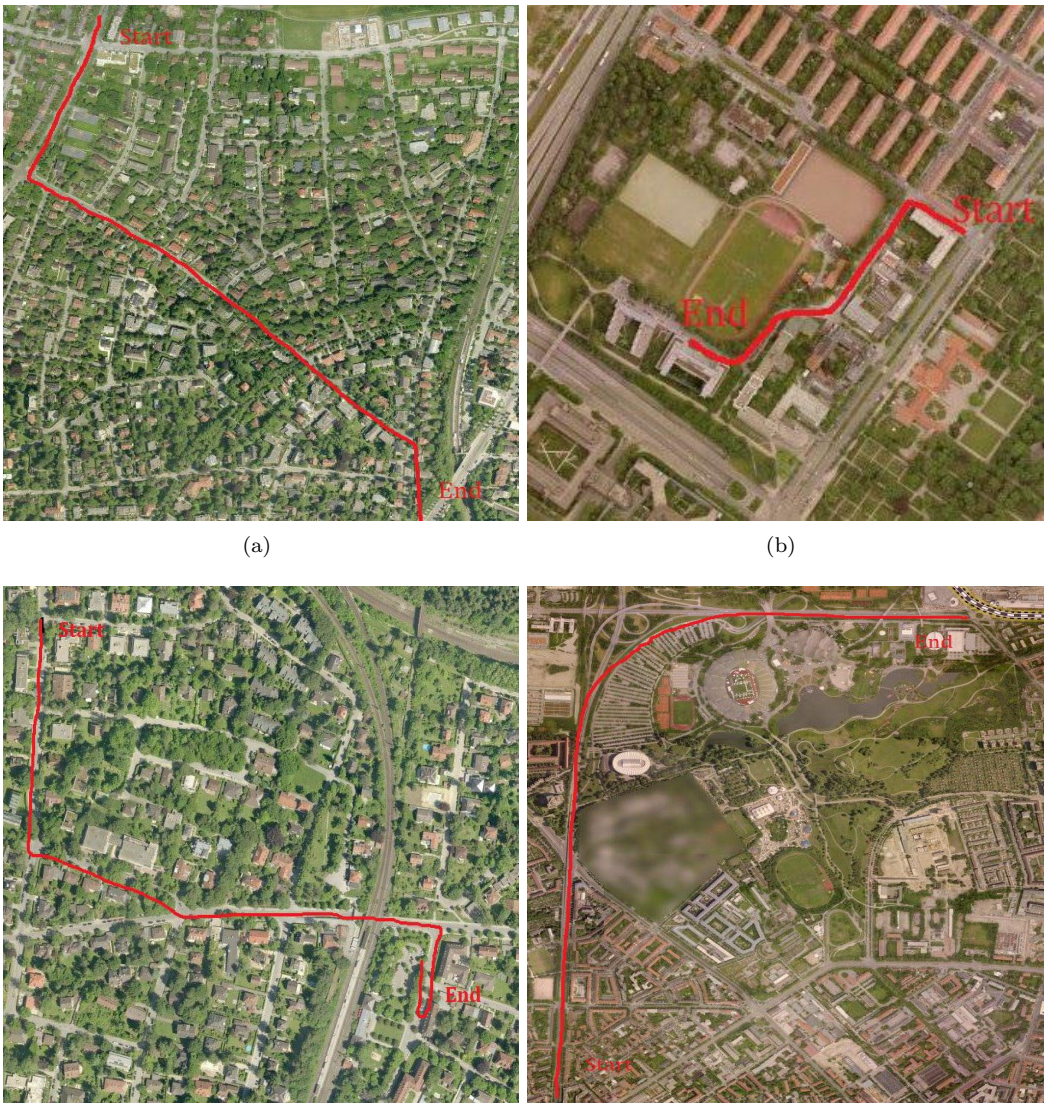
where τ denotes the given threshold, and i, j denotes the corresponding labels. Thus the states are effectively estimated over the whole space.

4.2.3 Simulation

In this section, the performance of the proposed solution is demonstrated. The PHD filter is evaluated through a large variety of images in urban environments. The data is collected from an iPhone 4 platform which equipped on the front window of the vehicle, and recorded as 30 frames/second with a resolution of 480×640. Furthermore, the GPS measurement is also utilized as the ground-truth to evaluate the performance of the proposed approach. The proposed approach is executed in Matlab on Duo CPU@3.0 Ghz, and the processing time is averaged as 1.2 second/frame for the PHD filter.

To analyze the proposed approach quantitatively and qualitatively, the RANSAC algorithm has also been implemented (more details could be found in Appendix. D). In contrast to the PHD filter, the RANSAC approach only operates on the pre-associated features to eliminate the outliers. Then, the dead reckoning method is used to reconstruct the whole trajectory (More details could be found in Appendix. F).

Fig. 4.3 illustrates the ground-truth in all scenarios by utilizing open street map [107]. During the experiments, the average length is about one kilometer which contains numbers



(a) (b)

(c) (d)

Figure 4.3: The ground-truth of all scenarios.

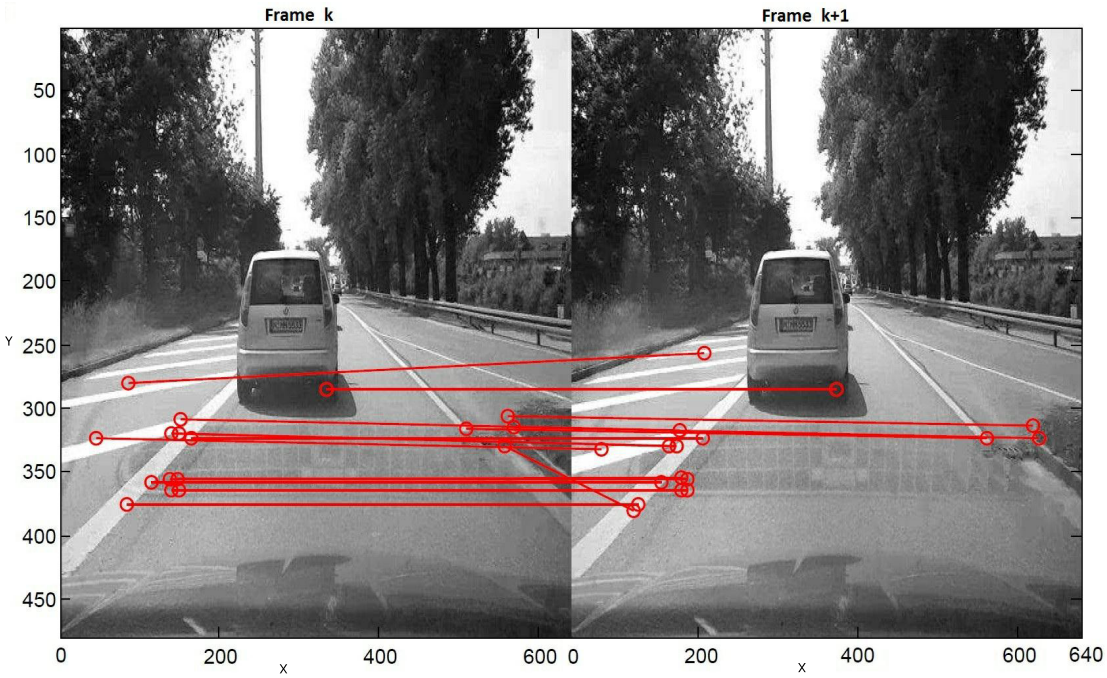
Table 4.1: Performance of the PHD filter and the RANSAC approach for single camera visual odometry

Index	Size	Distance	Frames	RANSAC error	PHD error
a	480×640	1281m	3510	24m(1.8%)	28m(2.1%)
b	480×640	413m	1500	38m(9.2%)	27m(6.5%)
c	480×640	950m	1800	135m(14.2%)	47m(4.9%)
d	480×640	4078m	6600	458m(11.2%)	376m(9.2%)

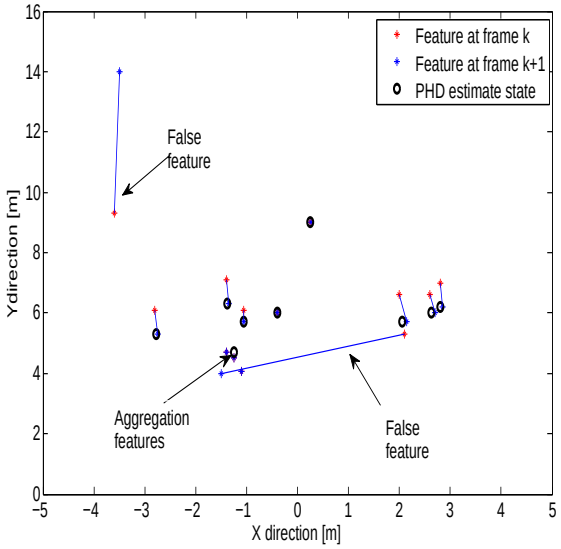
of moving objects, e.g. pedestrians, motors and bicycles. Figure 4.4(a) and 4.4(b) present the estimations in both the image and the vehicle coordinate systems. It is observed that some features are aggregated in a small region, in which the ego-motion vector is inefficiently estimated. In PHD filter, this challenge is addressed by utilizing the pruning and merging techniques. By calculating the Mahalanobis distance, the aggregated features are converted into a single state. The ego-motion vector is then calculated by averaging the remaining states. It is also observed that some features originate from moving vehicles. The RANSAC approach only relies on the statistic distribution model and there is no prior information (e.g. system process model) to describe such behaviors. However, for the PHD filter this challenge is addressed by utilizing the Bayesian inference. Based on the dynamic model, the posterior intensity is propagated and features from moving objects are thus considered as clutter.

Fig. 4.5 evaluates the performance of the proposed approach in contrast to the RANSAC approach. It is observed that both approaches contain biases, caused by the following reasons: First and foremost, the coordinate transformation issue. As previously discussed, features are assumed to have originated from the ground plane. However, in urban scenarios, features are randomly distributed over the whole space. Therefore, the transformations can be very imprecise. Since the trajectory is constructed by accumulating the ego-motion vectors, the actual error is a random walk and the cumulative error grows throughout the scenario. As a result of this, the estimated trajectories are inconsistent with the ground-truths. However, it is still observed that the performance of the PHD filter is better than the RANSAC approach.

Table 4.1 gives quantitative results for all scenarios. The index is related to the corresponding scenarios from Fig.4.3, where the distance is the length of each scenario. The performance is evaluated by calculating the distance error between the estimated position and the ground-truth. Again it is observed Fig. 4.5(b), Fig. 4.5(c) and Fig. 4.5(d) that the PHD filter provides a more precise result in, but a worse result in Fig. 4.5(a), when compared to the RANSAC approach. The reason why Fig. 4.5(a) performs a worse performance is due to the multiple moving targets. Since the PHD filter relies the system model to predict the further behavior, features from moving targets continually influence the estimation in the system model. Respect to the RANSAC, as the ego-motion vector is calculated independent with the frames, the performance is better than the PHD filter.



(a)



(b)

Figure 4.4: Interesting features and the estimation results. (a) Associated SIFT features in the image coordinate system. (b) The estimated states in the vehicle coordinate system.

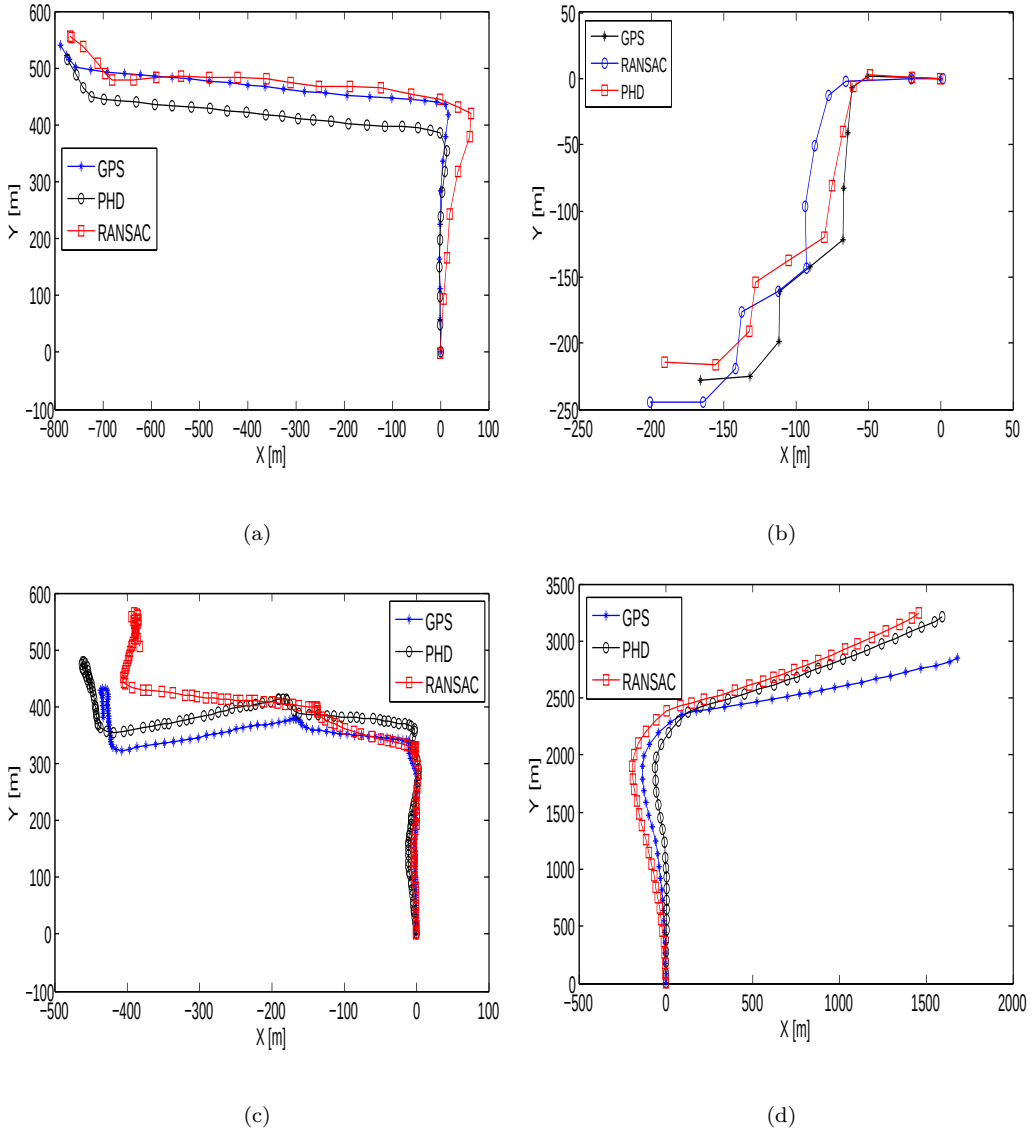


Figure 4.5: Performance of the PHD filter and the RANSAC approach for single camera visual odometry.

4.2.4 Summary

In this section, the Probability Hypothesis Density (PHD) filter is employed for the case of single camera visual odometry. The effectiveness of the proposed approach has been illustrated through quantitative comparisons with the state-of-the-art approaches. The advantages are summarized as follows:

First and foremost, the data association (matching) challenge is addressed. The traditional systems utilize the associated features to calculate the ego-motion vector. For the PHD filter, it operates on the set space to address the data association challenge.

Second, unevenly distributed features are effectively extracted. When features are aggregated into small regions, the PHD filter utilizes the pruning and merging techniques to optimally estimate the states.

Third, features stemming from moving objects are eliminated. For the PHD filter, such a challenge is addressed by utilizing the propagated posterior intensity function under the Bayesian network.

4.3 Stereo Camera Visual Odometry

In Sec. 4.2, using PHD filter for single camera visual odometry has been evaluated. The results demonstrate the high performance of utilizing PHD filter, with respect to imprecise coordinates of features. In this section, its potential for stereo camera visual odometry is investigated, to evaluate the performance with respect to precise coordinates of features.

Compared to monocular camera, stereo camera is relatively simple and more robust for visual odometry. During the extraction phase, a more accurate disparity map is acquired by utilizing simultaneously images from both the left and right cameras. By doing this, the coordinates are acquired more precisely in the vehicle coordinate system. Meanwhile, there is no requirement regarding to the distributions of features.

In this section, the proposed system uses not only the stereo camera, but also the gyroscope to measure the orientation changes. To guarantee the real-time requirement, the Speeded Up Robust Features (SURF) is thus utilized. Similar to single camera visual odometry, the stereo camera implementation is also divided into two phases: the extraction phase and the estimation phase. In first phase, the SURF features are extracted and collected in the vehicle coordinate system. The second phase mainly focuses on estimating the ego-motion vector in consecutive frames.

4.3.1 Feature Extraction

The goal of the extraction phase is to collect features as set-valued measurements. Instead of SIFT feature, as used in single camera visual odometry, the SURF feature is selected as the point-of-interest.

- **Interesting point detection**

In Sec. 4.2, the SIFT feature is utilized to calculate the ego-motion vector due to the invariant properties such as translation, rotation, scaling and illumination in urban environments. However, detecting all SIFT features under the real-time requirement is still quite challenging. In this scenario, the Speeded Up Robust Feature (SURF) is utilized to ensure the real-time requirement [92].

In contrast to the SIFT descriptor, the SURF descriptor has an ability of fast speed combined with a low computational cost. Meanwhile, the invariant properties are still almost the same when compared to the SIFT descriptor. Fig. 4.6 exhibits the SURF features with the same scenario in Sec. 4.2.

- **Coordinate Transformation**

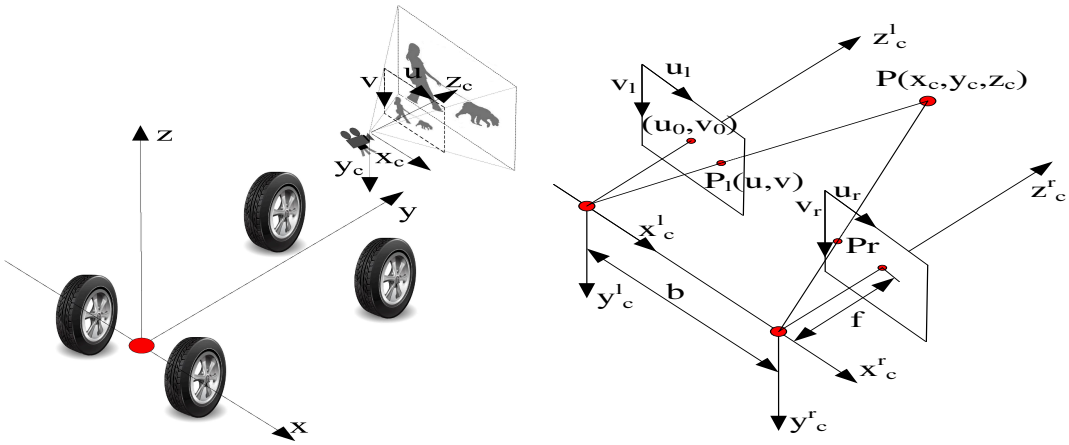
As shown in Fig. 4.7(a) and 4.7(b), cameras are aligned with their optical axis parallel to each other. The vehicle coordinate system is represented in a three-dimensional Cartesian coordinate system with origin in the middle of the rear axle. It is also assumed that point P is obtained through triangulation from the intersection of rays P_l and P_r. To reconstruct P in the vehicle coordinate system, the baseline b is utilized. As illustrated in Fig. 4.7(b), the depth of P in the left camera coordinate system is expressed as:

$$z_c = f \cdot \frac{b}{d} \tag{4.13}$$

where

$$d = x_c^r - x_c^l \tag{4.14}$$

Figure 4.6: Speeded Up Robust Features in urban environment



(a) (b)

Figure 4.7: Different coordinates systems. (a) The vehicle coordinate system. (b) The stereo camera coordinate system.

denotes the disparity of P, f is the focal length in both cameras.

Transforming Eq.(4.13), we have

$$z_c = f \cdot \frac{b}{d} = \frac{f}{d_x} \cdot \frac{b}{u_r - u_l} \tag{4.15}$$

Given the baseline, the focal length and the disparity, the depth is thus calculated. More details about the disparity map can be found in [108]. The corresponding position in the camera coordinate system is represented as

$$x_c = z_c \cdot \frac{u}{f} \tag{4.16}$$

$$y_c = z_c \cdot \frac{v}{f} \tag{4.17}$$

The transformation from the vehicle coordinate system to the camera coordinate system is given by the following transformation

$$\begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{4.18}$$

where T and R denotes the translation and the rotation matrix, respectively. Rearranging the above equation, the corresponding position in the vehicle coordinate system is acquired.

In this scenario, features are collected independent with their distributions.

4.3.2 PHD Filter Implementation

Displacements of the associated features should be the same value, where the ego-motion vector is calculated by averaging the set-valued state.

- State Model

The state is a vector which integrates the ego-motion vector. In this scenario, it is defined as

$$\mathbf{x}_k = [x_k, y_k, z_k, \beta_k, \dot{x}_k, \dot{y}_k, \dot{z}_k, \dot{\beta}_k]^T \tag{4.19}$$

where (x_k, y_k, z_k) and β_k denote the position and orientation, $(\dot{x}_k, \dot{y}_k, \dot{z}_k)$ and $\dot{\beta}_k$ denote the corresponding velocities.

- Process Model

The process model is proposed to represent and estimate the state in future. In this scenario, the proposed model operates on the 3D Cartesian coordinate system, which can be represented as follows:

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ z_{k+1} \end{bmatrix} = \begin{bmatrix} \cos\Delta\beta_k & -\sin\Delta\beta_k & 0 \\ \sin\Delta\beta_k & \cos\Delta\beta_k & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k - \Delta x_k \\ y_k - \Delta y_k \\ z_k \end{bmatrix} \tag{4.20}$$

where $(\Delta x_k, \Delta y_k, \Delta\beta_k)$ denotes the ego-motion vector (assuming experiments are conducted on the ground-plane, there is no changes with respect to the z-axis coordinate).

Rearranging the above equation, the process model is becomes:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_k \tag{4.21}$$

where

$$\mathbf{F} = \begin{bmatrix} \cos\beta_k & -\sin\beta_k & 0 & 0 & -\cos\beta_k & \sin\beta_k & 0 & 0 \\ \sin\beta_k & \cos\beta_k & 0 & 0 & -\sin\beta_k & -\cos\beta_k & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and $\mathbf{w}_k$ denotes the process noise.

- Measurement Model

The measurement model is also represented in the 3D Cartesian coordinate system as follows:

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \tag{4.22}$$

where

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and $\mathbf{v}_k$ denotes the measurement noise.

- PHD Recursion

The PHD filter is utilized to estimate the set-valued state with the following equations:

$$v_{k|k-1}(x_k) = \int P_{S,k}(x_{k-1}) f_{k|k-1}(x_k|x_{k-1}) v_{k-1}(x_{k-1}) \mathrm{d}x_{k-1} + \gamma_k(x_k)$$

$$v_k(x_k) = [1 - P_{D,k}(x_k)] v_{k|k-1}(x_k) + \sum\nolimits_{z \in Z_k} \frac{P_{D,k}(x_k) g_k(z|x_k) v_{k|k-1}(x_k)}{\kappa_k(z) + \int P_{D,k} g_k(z|\zeta) v_{k|k-1}(\zeta) \mathrm{d}\zeta}$$

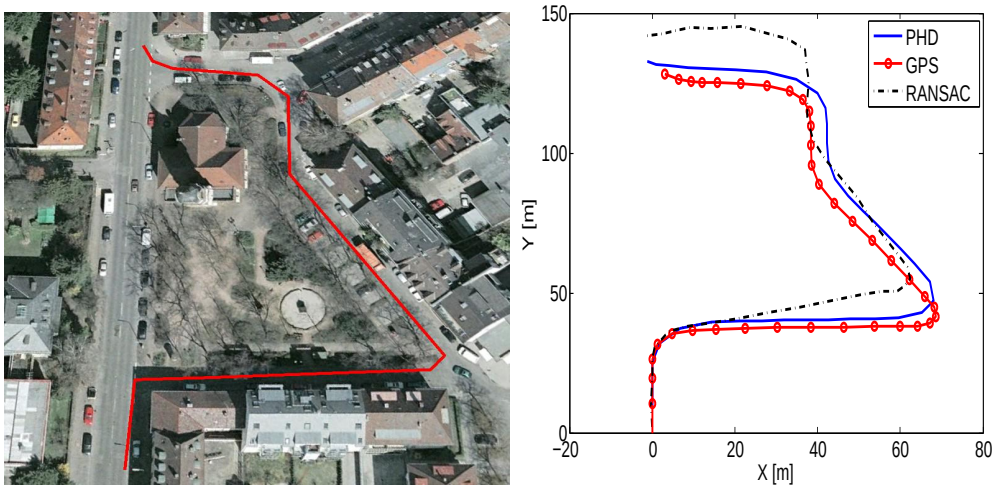
where x_k denotes the corresponding state of each target, with respect to the state $\mathbf{x}_k$ in this section.

The implementation process is the same as that presented in Sec. 4.2, in which the ego-motion vector is calculated by averaging the corresponding components among the states.

4.3.3 Simulation

In this section, the performance of the proposed solution is demonstrated in complex urban environments. To analyze the proposed solution quantitatively and qualitatively, the KITTI benchmark dataset is utilized which contains GPS, gyroscope and stereo camera at 10 frames/second with a resolution of 1344×391 [65]. Similar to single camera visual odometry, the measurement noises are given by manually adding Gaussian white noises. The proposed approach is executed in Matlab on Duo CPU@3.0 Ghz, and the processing time is averaged as 1.5 second/frame for the PHD filter.

Fig. 4.8 demonstrates the performance of the proposed approach in contrast to the ground-truth and the RANSANC approach. The conducted experiment is represented in Fig. 4.8(a)



(a) (b)

Figure 4.8: Stereo camera visual odometry. (a) Aerial view from GoogleEarth. (b) Odometry result in global coordinate system.

from aerial viewing, where the estimated trajectories are represented in Fig. 4.8(b). It is concluded that the PHD filter achieves high performance in contrast to the RANSAC approach. In order to prove this analytically, the position error is calculated at each step. As illustrated in Fig. 4.9, the error of the RANSAC approach increases significantly, which is also observed in Fig. 4.8 around the second corner. This is mainly due to the outliers of the extracted features. Fig. 4.10 exhibits the estimated amount in contrast to the extracted features in the PHD filter. A large number of features are extracted during the period, however, outliers are also included. Since the PHD filter also relies the system model, the states are effectively estimated in the whole process. In RANSAC, the outliers are eliminated based on the given threshold. Hence the position error is significantly increased in scenarios of huge amount of outliers.

Fig. 4.11 demonstrates the estimated orientation changes in contrast to the raw data from gyroscope. It is observed that the proposed solution is smooth in all steps. The filtered changes are then combined with the movements to reconstruct the trajectory in the deadlocking phase.

The proposed approach has also been executed on whole dataset, which leads to a decrease between 25% and 80% on the position error with the average as 54%, in contrast to the RANSAC approach. It is thus concluded that the proposed solution achieves high performance in visual odometry applications.

4.3.4 Summary

The benefits of utilizing the PHD filter for single camera visual odometry have already been demonstrated in Sec. 4.2. In this section, the proposed solution is extended to the stereo camera applications. The advantages are summarized as follows:

As a recurring theme within this thesis, the PHD filter addresses the data association challenge. Since the PHD filter operates on the set space, the ego-motion vector is calculated by averaging the set-valued states.

Secondly, the PHD filter is independent with the features. The proposed solution collects

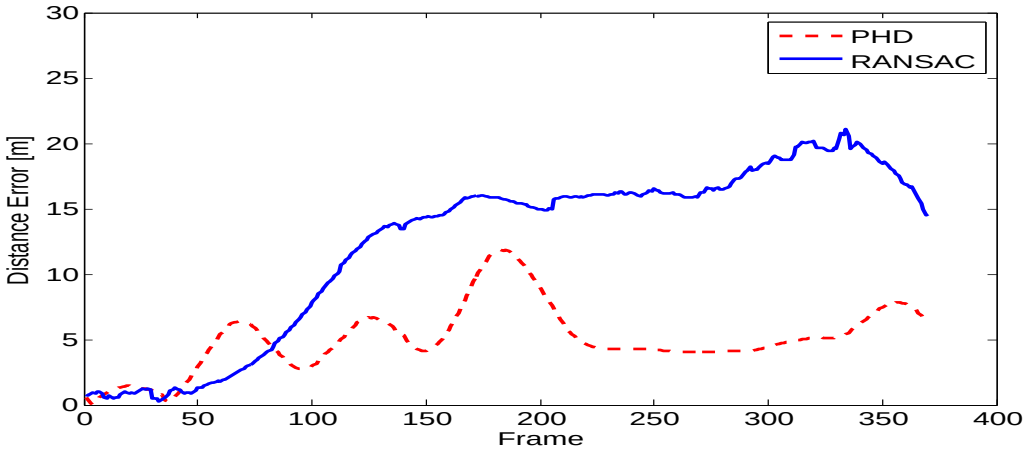


Figure 4.9: Evaluation on position error during the whole process.

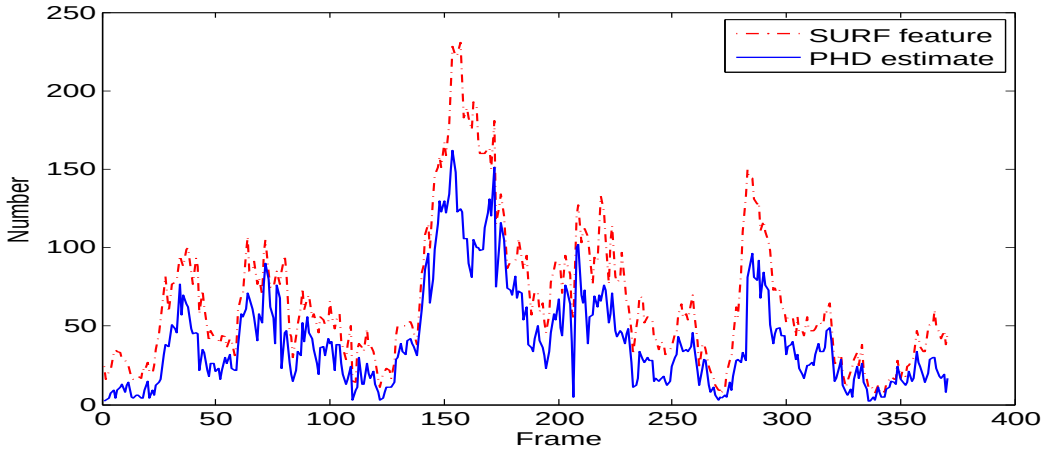


Figure 4.10: Estimated number of the set-valued state at each step.

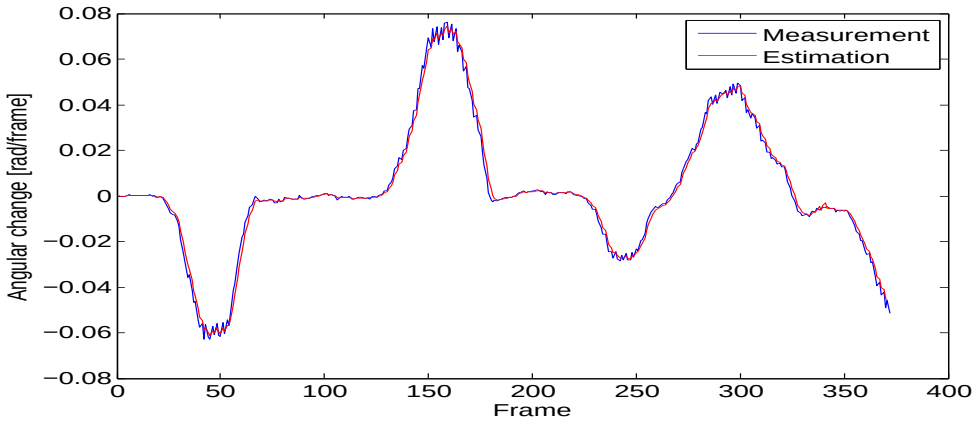


Figure 4.11: Orientation estimation based on the PHD filter.

features without considering their physical proprieties.

Last but not the least, the PHD filter provides reliable estimation in complex environments, e.g. features stemming from moving objects, or aggregating in small regions.

4.4 Conclusion

In this chapter, we have employed the Probability Hypothesis Density (PHD) filter in applications of visual odometry, which is quite challenging due to a large amount of outliers. The PHD filter is thus proposed to overcome the aforementioned challenges such as features association, features stemming from moving objects, or aggregating in small regions and so on.

Experimental results demonstrate that the proposed approach achieves high performance in contrast to the state-of-the-art approaches. In next chapter, the proposed approach is employed to the application of lane detection.

Chapter 5

Lane Detection

In Chapter 4, we have applied the Probability Hypothesis Density (PHD) filter to the visual odometry challenge. For traditional approaches, the data association step is required to calculate the track-measurement association and subsequently calculate the displacement between consecutive frames. The PHD filter avoids such issues by treating the data association as a part of the state estimation process and thus maintains high performance in complex environments.

In this chapter, we focus on tackling the issues of lane detection. The challenges are addressed by employing the PHD filter, to detect and identify interesting lane markings in urban scenarios.

This chapter is organized as follows: Section 5.1 gives brief introduction for lane detection. Section 5.2 explains in details the extraction process. Section 5.3 and Section 5.4 presents the estimation process as well as the experiment result. Finally, Section 5.5 concludes the chapter.

5.1 Introduction

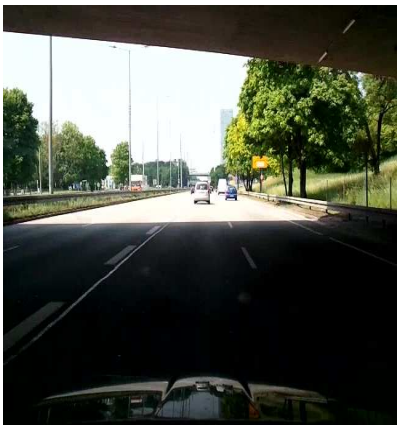
In recent years, Driving Assistance Systems (DAS) have been significantly developed to address the safety issues on road. As reported from the U.S. Transportation department in 2009: 59% of on-road traffic accidents were caused during the lane departure period. To avoid traffic accidents, the Lane Departure Warning system (LDW) has been proposed to detect the lane boundaries and estimate the geometry of the lane [109]. The performances of the remaining DAS applications have improved significantly. Given a scenario with known geometric constraints, the vehicle detection system could eliminate false detections which are out of the lane boundaries [110]. Also, by using the vehicle information, the lane boundaries are estimated precisely.

In urban environments, many issues make it difficult to detect lane boundaries, (e.g. shadows, poorly obscured markers, snow, strong sunlight etc.). Fig. 5.1 exhibits the representation of lane markers in four different scenarios: structured roadway, strong lighting, snow weather and constructions. Although several commercial products have already emerged in the market, it is still a challenge to fully detect lanes in all scenarios [111].

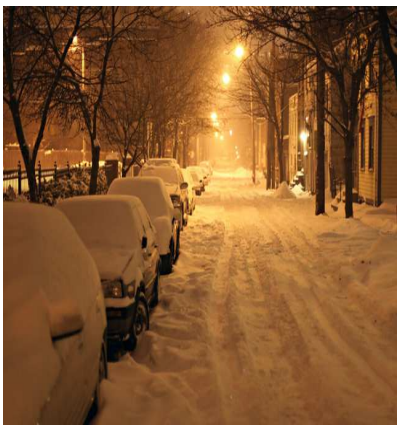
Most LDW systems consist of detecting the lane markings, fitting the lane models and tracking the lane poses. By using computer vision techniques, numbers of approaches have been developed, e.g. the adaptive thresholds method [112, 113], the steerable filters [114, 115] and the color based segmentation [116]. In general, the proposed methods could be mainly divided into: feature based approaches and model based approaches.

In feature based approaches, Hough transformation is the most commonly used method to extract particular shapes [117],[118],[119]. During the whole process, edges are extracted and

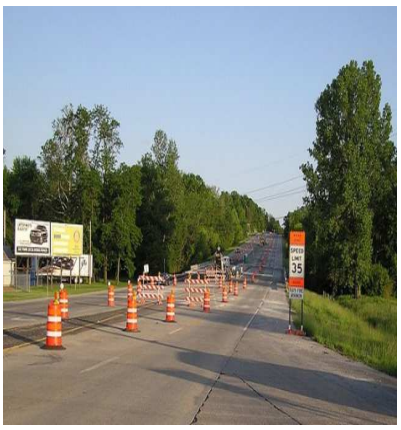
(a)



(b)



(c)



(d)

Figure 5.1: Lane detection in various situations. (a) Lane detection in ideal scenario. (b) Lane detection in shadow scenario. (c) Lane detection in snow scenario. (d) Lane detection in construction scenario.

(a) (b)

Figure 5.2: Image processing on region of interest. (a) Selection of the region of interest. (b) Image after ROI extraction and color transformation.

then concatenated as ploy-lines as well as lane markings [110], [120]. Since Hough transformation only detects straight lanes, it becomes difficult to fully detect irregular lanes in complex traffic environments [110, 121, 122, 123].

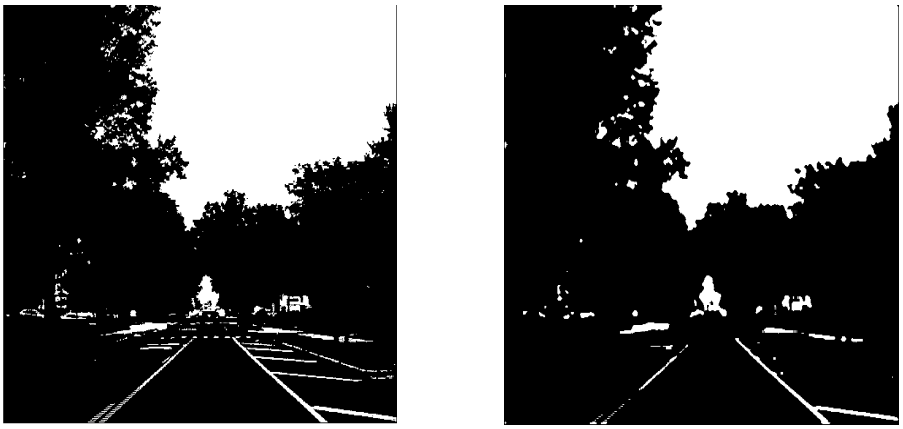
Model based solution is based on the geometric characteristics of lane markings, which assumes lanes are consisted by either straight models [124, 125] or parabolic curves [126, 127]. By doing this, the detected outliers are effectively eliminated as compared with feature based approaches. In [128], a set of curvature models are combined to calculate the best fitting model for each lanes. Meanwhile, Fardi et al. also proposed several models to extract lanes, which maintains high robustness in complex scenarios [129]. The challenge of model based approaches is to optimally fuse all models in a universal representation.

In this chapter, the Probability Hypothesis Density (PHD) filter is proposed to track lane markers. In contrast to the state-of-the-art approaches, the proposed method presents a recursive filtering solution under Bayesian network and achieves high performance in complex urban environments.

The proposed approach consists of the detection phase and the tracking phase. In detection phase, interesting points are extracted and then transformed to the vehicle coordinate system. In tracking phase, the PHD filter is used to estimate true lane markings.

5.2 Lane Marking Extraction

This section explains the extraction of features-of-interest (lane markings) for the further filtering process. In the first stage, an image is cropped to extract the Region Of Interest (ROI) where a median filter is utilized to remove the unnecessary noise. Meanwhile, the Otsu algorithm [130] is utilized to identify regions that potentially represent lanes where the image erosion is used to remove outliers. Finally, pixels stemming from the potential lane markings are collected as the set-valued measurement.



(a) (b)

Figure 5.3: Performance of image processing techniques. (a) Result of Otsu's segmentation. (b) Result of median filter processing.

- Region of Interest

In most scenarios, the road regions and non-road regions have obvious boundaries. The road regions are typically located under the horizon line whereas the non-road regions are above the horizon line [131]. Meanwhile, it is also expected that the road regions have a quasi-uniform color due to the fact that road regions are often represented as grey surfaces. Consequently, by segmenting the image as road and non-road regions, the outliers are much easier to detect and eliminate. To control the variations of different images, the Red Green Blue (RGB) representation is converted to a grey-scale representation . As exhibited in Fig. 5.2, the image is divided vertically into two parts and the ROI is transformed into the grey-scale afterwards.

- Median Filtering

In Fig. 5.2(b), the extracted ROI has already been transformed into the gray-scale representation, where the median filter is utilized to retain details and remove outliers. Meanwhile, the Otsu algorithm is also used to calculate an adaptive threshold to segment the image into binarization representation. Fig. 5.3 demonstrates the performances of the aforementioned techniques in the same scenario.

- Points-of-Interest Extraction

On one hand, an edge is a set of connected pixels which lie on the lane boundary between two regions. On the other hand, the outliers also exist. Therefore, the image erosion process not only extracts interesting points, but also eliminates outliers.

It is assumed that a pixel is identified as a potential lane marking only if it is a part of at least three horizontally connected pixels. The middlemost pixel of each set is extracted as the potential lane pixel. The implementation process is as follows: Start by searching for potential

Figure 5.4: Application of image erosion.

lane pixels at the bottom row, left to right. After the current row is processed, the erosion process moves on to the next row from bottom to top, line by line. Fig. 5.4 illustrates details of the proposed method.

- Coordinate Transformation

In Fig. 5.4 it is observed that the extracted pixels are considered as complete trajectories of lanes, whereas non-markings are treated as clutters. Thus the lane detection problem is converted to only track the interesting pixels.

In this scenario, the tracking process is implemented in the vehicle coordinate system with the same transformation process in Sec. 4.2.

- Time registration

To fully track lane pixels, the time sequences are also required at each frame. However, there is no information regarding the temporal representation of each feature. In this scenario, the proposed system records the row index of each pixel to represent the time sequences. Similarly, the row index is calculated from the bottom to the top row on the image and starts from step one.

5.3 PHD Filter Implementation

As mentioned earlier, the extracted markings are collected as set-valued measurements to recursively estimate the states. By doing this, outliers from the extracted pixels are eliminated during the filtering phase.

- State Model

The state of each pixel describes the corresponding pose related to the vehicle coordinate system. In this scenario, the state consists of the position and the velocity as follows:

$$\mathbf{x}_k = [x_k, y_k, \dot{x}_k, \dot{y}_k]^T \tag{5.1}$$

- Process Model

To predict future poses of lane pixels, the process model is proposed to represent the state as

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_k \tag{5.2}$$

with

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $\mathbf{w}$ denotes the process noise.

- Measurement Model

To map the projection between the estimated state and the acquired measurement, the measurement model is represented as

$$\mathbf{z}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k \tag{5.3}$$

with

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

where $\mathbf{v}$ denotes the process noise.

- PHD Recursion

As introduced in Sec. 2.2.1, the PHD filter is represented as follows

$$v_{k|k-1}(x_k) = \int P_{S,k}(x_{k-1}) f_{k|k-1}(x_k|x_{k-1}) v_{k-1}(x_{k-1}) \mathrm{d}x_{k-1} + \gamma_k(x_k)$$

$$v_k(x_k) = [1 - P_{D,k}(x_k)] v_{k|k-1}(x_k) + \sum\nolimits_{z \in Z_k} \frac{P_{D,k}(x_k) g_k(z|x_k) v_{k|k-1}(x_k)}{\kappa_k(z) + \int P_{D,k} g_k(z|\zeta) v_{k|k-1}(\zeta) \mathrm{d}\zeta}$$

where x_k equals to the state $\mathbf{x}_k$ in this scenario.

Notice that the PHD recursions have no closed form solution. Therefore, as in previous sections of this Thesis, the Gaussian Mixture PHD (GM-PHD) approximations is used for this scenario. More details about the implementation process could be found in Appendix. B.

- Implementation Issues

During the implementation, the birth model is utilized to represent the appearances of lane markings. To establish the birth model, the vehicle is assumed to drive on middle roads and the distance between two nearby lanes (left lane and right lane) is constant (3 meters in [132]). It should be noted that the proposed birth model is applied in most countries, with the same standard of the road construction.

Assuming the distance between the ego-vehicle and the right lane is σ_n, then the potential lane markings appear in both $[\sigma_n - 0.1, \sigma_n + 0.1]$ and $[\sigma_n - 0.1 - 3, \sigma_n + 0.1 - 3]$, respectively. For instance, if the vehicle is on the middle road, the lane markings are detected in regions of $[1.4, 1.6]$ and $[-1.6, -1.4]$, respectively. The extracted pixels which fall into the birth regions are utilized to initialize the states. When the ego-vehicle drives in complex environments exhibited in Fig. 5.1(c) and 5.1(d), the σ_n needs to be re-calibrated in the interval of five seconds.

Although interesting features are acquired during the nonlinear transformation, the standard Gaussian noises are considered as the noises.

5.4 Simulation

In this section, the performance of the proposed solution is demonstrated through a large variety of images in Sec. 4.2. All scenarios are captured with typical traffic participates, e.g. pedestrians, vehicles, lanes and shadows. The proposed approach is executed in Matlab on Duo CPU@3.0 Ghz, and the processing time is averaged as 85ms/frame (the image processing phase takes 70ms per frame and the PHD estimation phase takes 15ms per frame).

Fig. 5.5 illustrates the performance of the proposed solution where lane markings are represented as both straights and curves. The original image is exhibited on Fig. 5.5(a), in which it is observed that the left lane is modeled as curve and the middle lane still keeps straight. Instead of processing raw image, interesting points are extracted to ensure both the performance and the real-time requirement, respectively. Fig. 5.5(b) demonstrates the performance of the extraction phase, where most outliers are eliminated after median filtering. Fig. 5.5(c) demonstrates the estimated lane markings. Finally, the states are exhibited in the original image. It is observed that markings from the right lanes are insufficiently extracted, which is caused by the high threshold in Otsu's algorithms.

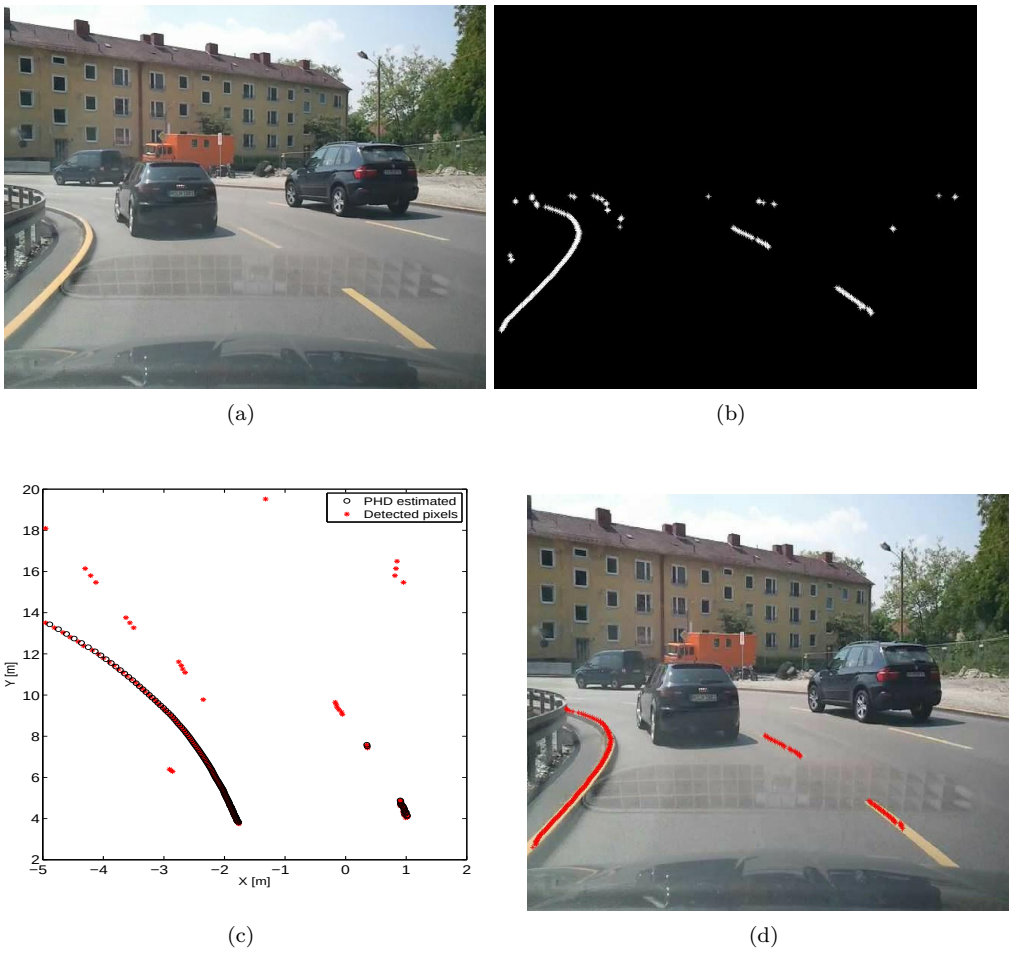
Fig. 5.6 exhibits the proposed approach for straight lanes. The additional lanes are tracked due to the fact that the extracted markings also fall into the birth regions. Since their geometry distributions are similar to each other, both of them are tracked during the estimation phase. It is noticed that the proposed solution only detects lanes which are parallel to the driving direction.

Tab. 5.1 also analyzes the proposed solution with respect to both the successful Detection Rate (DR) and the False Rate (FR). For each marking, if the distance between the estimated state and the ground-truth is within a given threshold, it is considered as a successful detection. Meanwhile, false detection also indicates that the estimated state may associate with outliers. For these tests, the ground-truth is acquired by using GPS and the street information from open street map. The final result is calculated by accumulating all estimations:

$$\mathrm{DR} = \frac{\mathrm{NTP}}{\mathrm{NGL}}, \ \mathrm{FR} = 1 - \frac{\mathrm{NTP}}{\mathrm{NDR}} \tag{5.4}$$

where NGL denotes the amount of the extracted lane markings, NTP denotes the amount of markings which are successfully detected. And, NDR denotes the amount of all states. Notice that the PHD filter performs reliable performance for lane detection in complex urban environments.

The benefits of the proposed solution are summarized as follows:



(a) (b)

(c) (d)

Figure 5.5: Lane detection on curvy lanes. (a) The original image. (b) The performance of the image processing phase. (c) The estimated lane pixels. (d) The final result.

Table 5.1: Performance of the proposed solution in urban environments

Scenario	DR	FR
Urban environment	93.6%	6.2%

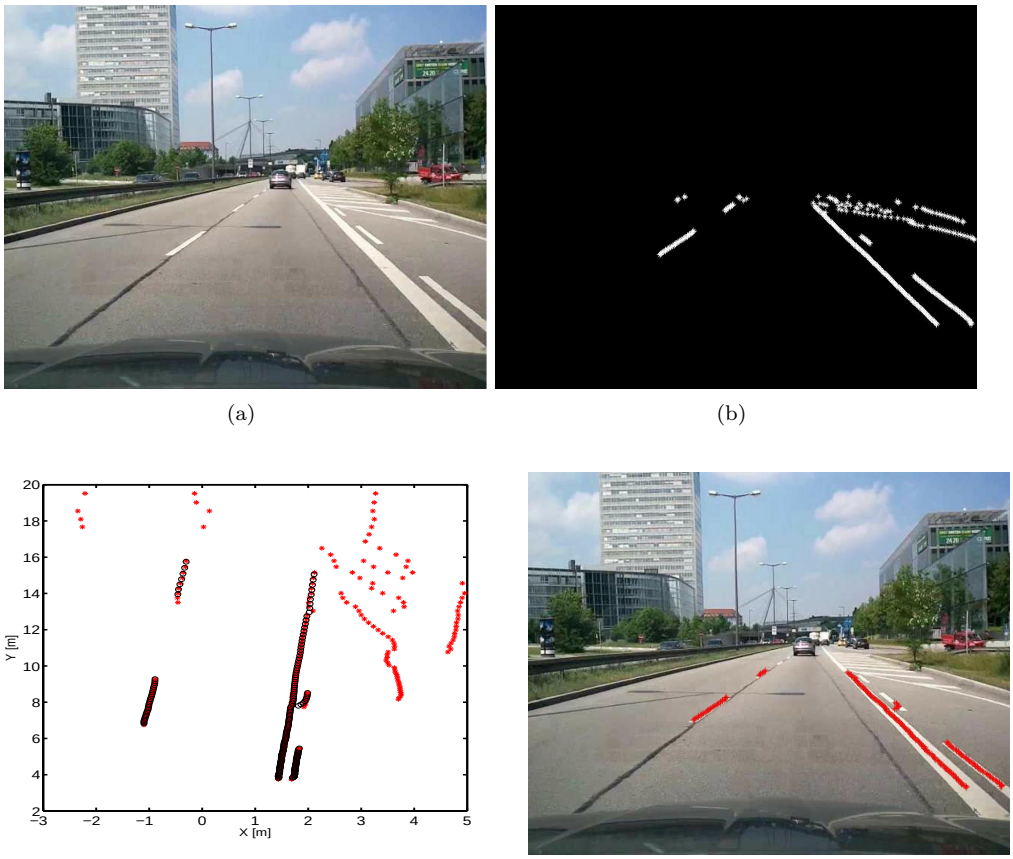


Figure 5.6: Lane detection on straight lanes. (a) The original image. (b) The performance of the image processing phase. (c) The estimated lane pixels. (d) The final result.

Again, following the main theme for this thesis, the PHD filter solves the computationally burdensome data association requirement. In traditional approaches, it is quite difficult to eliminate non-lane markings. However, in the PHD filter, all markings are collected as a set-valued measurement to estimate the set-valued state, recursively. However, it should be noted that the PHD filter relies the land-markings to detect the road-lanes, which might not be practical in complex environments like Fig. 5.1(c) and 5.1(d).

Furthermore, there is no requirement regarding to the prior knowledge of environments. Compared to the traditional approaches, the proposed solution detects lanes independent with their distributions.

5.5 Conclusion

This chapter presented the Probability Hypothesis Density (PHD) filter for lane detection, with the performance evaluated in real dataset containing both structure roads (straight lanes, little outliers) and unstructured roads (curves, lanes with partial observations, shadows and large number of outliers).

As the proposed solution directly operates on the set level, the data association challenge is addressed. In future, we will consider extending the proposed solution to jointly detect both lanes and vehicles, respectively.

Chapter 6

Conclusions

This chapter presents closing arguments for this thesis. Specially, a summary of contributions as well as directions for future work.

> Data fusion is the process of directing the right data sources on the right targets at the right time, with the goal of detecting, localizing, identifying and determining the threat potential of as many targets of interest as possible. –Ronald P.S.Mahler

6.1 Summary

Advanced Driver Assistance Systems (ADAS) are developed to reduce the cognitive burden of a vehicle driver and ultimately to improve safety and improve the driving experience. Surrounding objects are monitored by heterogeneous sensors with different fields of views. Thus the situation awareness system of ADAS systems consists of different sensors with different field of views, which acquires information of randomly varying numbers of targets. A rigorous mathematical foundation for dealing such scenarios has been developed in this thesis, which is based on the Random Finite Set (RFS) statistic. Under the RFS framework, observations are mathematically collected into a single 'meta-observation', called 'set-valued measurement'. Meanwhile, targets are mathematically collected into a single 'meta-target', called 'set-valued state'. In contrast to the state-of-the-art approaches, the RFS results in a systematic Bayesian unification of the detection, association and tracking in multi-sensor-multi-target scenarios.

In this thesis, we have employed the Probability Hypothesis Density (PHD) filter from the RFS framework, for dealing with an unknown and possibly time varying number of targets acquired from heterogeneous sensors in complex urban environments, where both the data association and sensor uncertainty challenges are addressed respectively. The proposed solution essentially concerns the 'Improve the performance of heterogeneous sensors integration for highly assisted driving functionality using the random finite set statistic', with respect to different applications of ADAS including localization, visual odometry and lane detection. Demonstrations using the proposed solution are also exhibited, compared to the state-of-the-art approaches.

All demonstrations and findings in this thesis were published in peer reviewed conferences and journals, thus proving its credibility and novelty.

6.2 Contributions

Traditional Bayesian techniques are mostly applied in multi-target-multi-sensor scenarios which results in significant challenges:

- **Data Association**-in which measurements are correctly associated to interesting targets. In case of an uncertain (or even missing) measurement-to-target association exists, it is a significant challenge to correctly estimate true states.
- **Spatial Registration**-in which measurements are transformed from individual platforms to a uniform reference system. Meanwhile, errors also contain both the random noises and the biases, which should be eliminated during the filtering phase.

A generic framework has been investigated for integrating data from heterogeneous sources to improve performance of highly assisted driving. In contrast to the traditional solutions which mainly rely on the single-object filtering, the proposed solution deals with random number of targets observed by random number of sensors with different field of views, where the corresponding recursions originate from a rigorous derivation of the novel likelihood for the finite set-valued observation. Experiments and simulations have illustrated that, random finite set statistic offers principled, reliable and efficient solution in complex urban environments.

First and foremost, the proposed solution integrates information from heterogeneous sources in presence of unknown data association and sensor uncertainty, to demonstrate a generic framework for further applications of ADAS. In practical, most traditional Bayesian fusion algorithms are developed with the condition of single-input-single-output, where the data association uncertainty becomes a significant challenge to fuse information from heterogeneous sensors. Also, sensors often suffer from biases that further dilute the strength of data association and increase the estimation uncertainty. In this thesis, a generic framework is proposed where data consists of finite sets of values, rather than each individual point or vector, which originates from measurement and state, respectively. The set-valued observation is utilized to detect, identify and estimate multiple objects and biases in the set-valued state, with no further need of data association and sensor calibration.

Second, a number of ADAS applications have been developed. The PHD filter has been proposed to track an unknown number of targets using the developed Lane Detection (LD), Vehicle Localization (VL) and Visual Odometry (VO) Systems. To analyze the performances of the developed systems, both quantitatively and qualitatively, the KITTI benchmark dataset and our own datasets have been evaluated and compared with the state-of-the-art approaches. In our implementation, we achieved high reliability and flexibility in complex urban environments.

Last but not least, the proposed solution exhibits the potential of modular architecture for ADAS. Under the RFS framework, information from heterogeneous sources is collected as a finite set of observations, whereas a generic likelihood function is generated according to the statistical models of each individual sensor. Meanwhile, the motion models and the likelihood functions are also generated in the set-valued state. Adding new sensors only requires specifying a specific configuration in the proposed solution, e.g. sensor range, field of view and statistical parameter.

Notice that the PHD filter is applicable to multi-sensor-multi-target, the following two limitations have been identified: essentially, the Bayesian filtering (Kalman filter, or Particle filter) provides a deterministic solution, the optimal solution however requires that the association and registration are known or directly estimated outside the filter. The PHD filter overcomes these issues but at the expense of computational complexity and sub-optimal solution. It is for the verification engineers to decide the most appropriate algorithm to use.

6.3 Future Work

Applying RFS in automotive domain is relatively novel compared to the traditional approaches. This thesis is among the early attempts to extend the RFS framework to applications of highly assisted driving, and there is still scope for further improvement in both theoretical and practical fields.

The RFS framework exhibits a potential direction for integrating data from heterogeneous sources with different fields of views, which has already been validated in previous chapters. However, integration of expert systems is still quite challenging. A unified, probabilistic foundation for different expert systems (fuzzy logic, Dempster-Schafer, Bayesian inference and rule-based evidence) integration could significantly enhance the performance of situation awareness. On one hand, this might lead to a better performance in terms of quality. On the other hand, this might lead to a modular fusion architecture in format of both hardware (sensor) and software (multi-expert system).

Another improvement can be the integration of social media. Recently, the ubiquity of mobile communication and the prosperity of location based services significantly help people experiencing traffic events. Compared to information acquired from single vehicle, social media emphasizes real time computing and applications with big data techniques. Information interactions are represented in forms of vehicle-to-vehicle, vehicle-to-website, vehicle-to-people and vehicle-to-infrastructures. An intelligent analysis of such data could optimize the utilization of both the transport infrastructure and the traffic management. In future work, the RFS framework might also process data from all social medias.

In conclusion, the proposed solution might be extended in a variety of ways, making it a fascinating framework for ADAS.

Appendix A

Bayes filter

In Chapter 2, $\pi_k(x_k|z_{1:k})$ (Eq. 2.7) contains whole information with respect to both the state and measurement up to time k. The posterior density is propagated by

$$\pi_{k|k-1}(x_k|z_{1:k-1}) = \int f_{k|k-1}(x_k|x_{k-1})\pi_{k-1}(x_{k-1}|z_{1:k-1})\mathrm{d}x_{k-1} \tag{A.1}$$

$$\pi_k(x_k|z_{1:k}) = \frac{g_k(z_k|x_k)\pi_{k|k-1}(x_k|z_{1:k-1})}{\int g_k(z_k|x)\pi_{k|k-1}(x|z_{1:k-1})\mathrm{d}x} \tag{A.2}$$

Notice that the Bayes recursion does not admit closed form solution, whereas the Kalman filter, the Extended Kalman filter, the Unscented Kalman filter and the Particle filter provide close form solutions in linear and non-linear scenarios.

A.1 Kalman filter

The Kalman filter is a closed form solution to the Bayes recursion in Gaussian cases, which assumes that the dynamic and measurement models are linear transformations [16].

$$x_k = F_{k-1}x_{k-1} + v_{k-1} \tag{A.3}$$

$$z_k = H_k x_k + w_k \tag{A.4}$$

where F_{k-1} and H_k denote the transition matrix and the observation matrix respectively. v_{k-1} and w_k are represented as independent zero-mean Gaussian variables with covariance matrices Q_{k-1} and R_k. The transition density and measurement likelihood are thus represented as

$$f_{k|k-1}(x_k|x_{k-1}) = \mathcal{N}(x; F_{k-1}x_{k-1}, Q_{k-1}) \tag{A.5}$$

$$g_k(z_k|x_k) = \mathcal{N}(z_k; H_k x_k, R_k) \tag{A.6}$$

where $\mathcal{N}(\,\cdot\,; m, P)$ denotes a Gaussian distribution with mean m and covariance P.

Assuming the posterior density is a Gaussian distribution at step $k-1$,

$$\pi_{k-1}(x_{k-1}|z_{1:k-1}) = \mathcal{N}(x_{k-1}; m_{k-1}, P_{k-1}) \tag{A.7}$$

the predicted density is also a Gaussian distribution in the form of

$$\pi_{k|k-1}(x_k|z_{1:k-1}) = \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}) \tag{A.8}$$

where

$$m_{k|k-1} = F_{k-1}x_{k-1} \tag{A.9}$$

$$P_{k|k-1} = Q_{k-1} + F_{k-1}P_{k-1}F_{k-1}^T \tag{A.10}$$

and $[\cdot]^T$ denotes transpose of a matrix $[\cdot]$.

The update density is Gaussian distributed and represented as

$$\pi_k(x_k|z_{1:k}) = \mathcal{N}(x_k; m_k, P_k) \tag{A.11}$$

where

$$m_k = m_{k|k-1} + K_k(z_k - H_k x_{k|k-1}) \tag{A.12}$$

$$P_k = [I - K_k H_k]P_{k|k-1} \tag{A.13}$$

$$K_k = P_{k|k-1}H_k^T S_k^{-1} \tag{A.14}$$

$$S_k = R_k + H_k P_{k|k-1}H_k^T \tag{A.15}$$

Note that K_k is considered as the Kalman gain, $z_k - H_k x_{k|k-1}$ is referred to the innovation and S_k is the innovation covariance.

A.2 Extended Kalman filter

The Kalman filter is utilized to estimate the state, in which the process function and the measurement function are both linearly represented. In conditions of nonlinear environment, the Extended Kalman Filter (EKF) is proposed to handle the nonlinearity based on the Taylor series expansion. Assuming f_k and h_{k+1} are sufficiently differentiable, the Taylor expansion can be calculated as follows:

$$f_k(x_k) = f_k(x_{k|k}) + \frac{\mathrm{d}f_k(x)}{\mathrm{d}x}|_{x=x_{k|k}}(x_k - x_{k|k}) + \triangle \tag{A.16}$$

$$h_{k+1}(x_{k+1}) = h_{k+1}(x_{k+1|k}) + \frac{\mathrm{d}h_{k+1}(x)}{\mathrm{d}x}|_{x=x_{k|k}}(x_k - x_{k|k}) + \triangle \tag{A.17}$$

where $\triangle$ denotes the higher oder terms. The derivatives of the corresponding Jacobian matrices are defined as F_k and H_{k+1}. By neglecting the higher order Taylor series, we have:

$$\begin{aligned} x_{k+1} &= F_k x_k + v_k + u_k \\ z_{k+1} &= H_{k+1}x_{k+1|k} + w_{k+1} + y_{k+1} \end{aligned} \tag{A.18}$$

where

$$u_k = f_k(x_{k|k}) - F_k x_{k|k} \tag{A.19}$$

$$y_{k+1} = h_{k+1}(x_{k+1|k}) - H_{k+1}x_{k+1|k} \tag{A.20}$$

Thus the EKF is derived via the standard Kalman filter as follows:

- Prediction

$$x_{k+1|k} = f_k(x_{k|k}) \tag{A.21}$$

$$P_{k+1|k} = Q_{k+1} + F_k P_{k|k}F_k^T \tag{A.22}$$

- Update

$$x_{k+1|k+1} = x_{k+1|k} + K_k(z_{k+1} - H_{k+1}x_{k+1|k}) \tag{A.23}$$

$$P_{k+1|k+1} = [I - K_{k+1}H_{k+1}]P_{k+1|k} \tag{A.24}$$

$$K_k = P_{k+1|k}H_{k+1}^T S_k^{-1} \tag{A.25}$$

$$S_k = R_{k+1} + H_{k+1}P_{k+1|k}H_{k+1}^T \tag{A.26}$$

The linearization process is often insufficient if the higher order terms of the Taylor series contribute significantly to introduce errors. However, the EKF is still considered as the most widely used filtering algorithm for nonlinear systems.

A.3 Unscented Kalman Filter

Compared to the EKF, the Unscented Kalman filter extracts $2n+1$ (n is the dimension of the state) sigma points $\{\omega_i, \chi_i\}$ and passes these through the nonlinear functions instead of the approximating state and measurement transition by Taylor series expansion.

1. Predict: a set of $2n+1$ sigma points are derived from the estimated state:

$$\chi_{k-1|k-1}^0 = x_{k-1|k-1}$$

$$\chi_{k-1|k-1}^i = x_{k-1|k-1} + (\sqrt{(n+\lambda)P_{k-1|k-1}})_i, \quad i = 1, ..., n$$

$$\chi_{k-1|k-1}^i = x_{k-1|k-1} - (\sqrt{(n+\lambda)P_{k-1|k-1}})_i, \quad i = 1+n, ..., 2n \tag{A.27}$$

where $(\sqrt{(n+\lambda)P_{k-1|k-1}})_i$ is the i_{th} column of the matrix square root of $(n+\lambda)P_{k-1|k-1}$. The predicted mean and covariance are computed by propagating the sigma points through:

$$\chi_{k|k-1}^i = f(\chi_{k-1|k-1}^i), \quad i = 0, ..., 2n$$

$$x_{k|k-1} = \sum_{i=0}^{2n} \omega_m^i \chi_{k|k-1}^i$$

$$P_{k-1|k-1} = \sum_{i=0}^{2n} \omega_c^i [\chi_{k|k-1}^i - x_{k|k-1}][\chi_{k|k-1}^i - x_{k|k-1}]^T + \mathbf{Q}_{k-1} \tag{A.28}$$

where the weights $\omega^i (i = 0, 1, ..., 2n)$ is given by

$$\omega_m^0 = \frac{\lambda}{n+\lambda}, \quad \omega_c^0 = \frac{\lambda}{n+\lambda} + (1 - \alpha^2 + \beta), \quad \omega_m^i = \omega_c^i = \frac{1}{2(n+\lambda)}$$

where λ is considered as the scaled factor.

2. Update: a set of $2n+1$ sigma points are derived to update the mean and covariance:

$$\chi_{k|k-1}^0 = x_{k|k-1}$$

$$\chi_{k|k-1}^i = x_{k|k-1} + (\sqrt{(n+\lambda)P_{k|k-1}})_i, \quad i = 1, ..., n$$

$$\chi^i_{k|k-1} = x_{k|k-1} - (\sqrt{(n+\lambda)P_{k|k-1}})_i, \quad i = n+1, ..., 2n \tag{A.29}$$

The sigma points are projected through the observation function as follows:

$$\mathcal{Y}^i_k = h(\chi^i_{k|k-1}), \quad i = 0, ..., 2n \tag{A.30}$$

where the predicted measurement mean and covariance are

$$\mathbf{y}_k = \sum_{i=0}^{2n} \omega^i_m \mathcal{Y}^i_k$$

$$P_{zz} = \sum_{i=0}^{2n} \omega^i_c [\mathcal{Y}^i_k - \mathbf{y}_k][\mathcal{Y}^i_k - \mathbf{y}_k]^T + \mathbf{R}_k \tag{A.31}$$

The state measurement covariance is given by

$$P_{xz} = \sum_{i=0}^{2n} \omega^i_c [\chi^i_{k|k-1} - x_{k|k-1}][\chi^i_{k|k-1} - x_{k|k-1}]^T \tag{A.32}$$

The UKF gain is thus computed as

$$K_k = P_{xz} P^{-1}_{zz} \tag{A.33}$$

The updated state and covariance are represented as

$$x_{k|k} = x_{k|k-1} + K_k(y_k - \mathbf{y}_k)$$

$$P_{k|k} = P_{k|k-1} + K_k P_{zz} K^T_k \tag{A.34}$$

A.4 Particle filter

The particle filter, also called Sequential Monte Carlo (SMC) recursion, is proposed to approximate the densities in the Bayes filter based on the random sample or point mass approximations [31, 32, 26, 33].

By using the Monte Carlo (MC) sampling and the Importance Sampling (IS), the particle filter achieves high precision in both non-linear and non-Gaussian scenarios.

Consider the goal of Monte Carlo sampling for an arbitrary probability density $\pi(\cdot)$ by assuming N independently and identically distributed (i.i.d.) samples from $\{x^i\}^N_{i=1}$, the density $\pi(\cdot)$ is calculated as

$$\pi(x) \approx \frac{1}{N} \sum_{i=1}^{N} \delta_{x^i}(x) \tag{A.35}$$

This illustrates that for any arbitrary function f that is π integrable, the asymptotic convergence is confirmed as

$$\frac{1}{N} \sum_{i=1}^{N} f(x^i) \underset{N\to\infty}{\overset{a.s.}{\to}} \int f(x)\pi(x)\mathrm{d}x \tag{A.36}$$

The importance sampling is often utilized in case of the density π is known up to a normalizing constant $\pi(x) \propto p(x)$, which is difficult to calculate. The concept of the importance sampling is to generate samples from a known density $q(\cdot)$ which considered as the importance

function that is close to $\pi(\cdot)$, on condition that weighting these samples accordingly in order to acquire a point mass approximation to $\pi(\cdot)$. The mathematical representation is as follows:

$$\pi(x) \approx \sum_{i=1}^{N} \widetilde{w}^i \delta_{x^i}(x) \tag{A.37}$$

where

$$\widetilde{w}^i = \frac{w(x^i)}{\sum_{i=1}^{N} w(x^j)}, w(x^i) = \frac{p(x^i)}{q(x^i)} \tag{A.38}$$

are defined as the normalized importance weights and the importance weights respectively. The asymptotic convergence is approximated as

$$\frac{1}{N} \sum_{i=1}^{N} \widetilde{w}^i f(x^i) \overset{a.s.}{\underset{N\to\infty}{\to}} \int f(x)\pi(x)\mathrm{d}x \tag{A.39}$$

The Sequential Importance Sampling (SIS) filter is thus proposed with the goal of utilizing importance sampling to recursively construct point mass approximations to the posterior density.

Assume at time $k-1$, the posterior density $\pi(\cdot)$ consists of a set of weighted particles $\{w^i_{k-1}, x^i_{k-1}\}_{i=1}^{N}$ as

$$\pi_{k-1}(x_{k-1|z_{1:k-1}}) \approx \sum_{i=1}^{N} w^i_{k-1} \delta_{x^i_{k-1}}(x_{k-1}) \tag{A.40}$$

If the given proposal density $q_k(\cdot|x^i_{k-1}, z_k)$ with

$$support(\pi_k) \subseteq support(q_k) \tag{A.41}$$

the posterior density $\pi_k(\cdot)$ is approximated as a new set of weighted particles as

$$\pi_k(x_{k|z_{1:k}}) \approx \sum_{i=1}^{N} w^i_k \delta_{x^i_k}(x_k) \tag{A.42}$$

where

$$w^i = \frac{\widetilde{w}^i_k}{\sum_{i=1}^{N} \widetilde{w}^i_k} \tag{A.43}$$

$$\widetilde{w}^i_k = w^i_{k-1} \frac{g_k(z_k|x^i_k)\pi_{k|k-1}(x^i_k|x^i_{k-1})}{q_k(x^i_k|x^i_{k-1}, z_k)} \tag{A.44}$$

$$x^i_k \sim q_k(\cdot|x^i_{k-1}, z_k) \tag{A.45}$$

The simple SIS algorithm is affected by the phenomenon of particle degeneracy where the variances of the importance weights increase over time. This phenomenon has affected the performance of the filter and been addressed by a re-sampling step. In [32], numerous of re-sampling schemes are available. All of them affect the computational load as well as the Monte Carlo approximation error. The Markov Chain Monte Carlo (MCMC) step is thus utilized to rejuvenate particle diversity if necessary [133, 134]. A pseudo-code of the whole procedure is shown in Algorithm 1, more details could be found in [33, 135, 136].

The basic idea of the MCMC step is to apply a Markov chain transition kernel $K(\cdot|x_k)$ with invariant distribution $f_k(\cdot|z_{1:k})$ on condition that $\in K(x^*_k|x_k)f_k(x_k|z_{1:k}) = f_k(x^*_k|z_{1:k})$.

A. BAYES FILTER

In particle filter, the target distributions are approximated by a number of particles, which are carried forward by utilizing the Sampling Importance Sampling (SIS) and re-sampling steps. Therefore, the performance of the particle filter depends on the design of the distribution. For instance, Rao-Blackwellization techniques are combined with the particle filter to improve performance for particular classes of models [137, 135]. The idea is to divide the state into a non-linear non-Gaussian component and a linear Gaussian component. The Kalman filter is utilized in linear condition, whereas the particle filter is applied to the non-linear condition.

In Monte Carlo techniques, the sampling is performed deterministically with the promise of a faster rate of convergence as well as better understood error propagation characteristics [138]. With smoothing techniques, the continuous approximations to the posterior density are obtained.

Algorithm 1: SIR Particle Filter outline

Input : $\{x_{k-1}^i, w_{k-1}^i\}_{i=1}^N$ and a new measurement z_k;
Output : $\{x_k^i, w_k^i\}_{i=1}^N$;
Sample initial particles $\{x_0^i\}_{i=1}^N$ from $p(x_0)$;
Set weights w_0^i to $\frac{1}{N}$;
Prediction :
while $i \leftarrow 1$ ***to*** N **do**
    Draw a sample: $\widetilde{x}_k^i \sim q_k(\cdot|x_{k-1}^i, z_k)$;
Update :
while $i \leftarrow 1$ ***to*** N_p **do**
    Calculate weights: $\widetilde{w}_k^i = w_{k-1}^i \frac{g_k(z_k|x_k^i)\pi_{k|k-1}(x_k^i|x_{k-1}^i)}{q_k(x_k^i|x_{k-1}^i, z_k)}$;
Normalize the weights :$\hat{w}_k^i = \frac{\widetilde{w}_k^i}{\sum_{i=1}^N \widetilde{w}_k^i}$;
Compute the weight degeneracy :
Effective Sample Size :$N_{\text{eff}} = [\sum_{i=1}^N (\hat{w}_k^i)^2]^{-1}$
Re-sample :
Draw a sample: $\widetilde{x}_k^i \sim q_k(\cdot|x_{k-1}^i, z_k)$;
if $N_{\text{eff}} < N_{\text{thr}}$ **then**
    Generate a new set of particles $\{x_k^i\}_{i=1}^N$, with $P(x_k^i = \widetilde{x}_k^i) = \hat{w}_k^i$ for any i;
    Set the weights w_k^i to $\frac{1}{N}$;
else
    Copy $(\widetilde{x}_k^i, \hat{w}_k^i)$ to (x_k^i, w_k^i), for $i = 1, \cdots, N$
end

Appendix B

Probability Hypothesis Density (PHD) filter

In Chapter 2, the PHD recursion is defined as:

$$v_{k|k-1}(x_k) = \int P_{S,k}(x_{k-1}) f_{k|k-1}(x_k|x_{k-1}) v_{k-1}(x_{k-1}) \mathrm{d}x_{k-1} + \gamma_k(x_k) \tag{B.1}$$

$$v_k(x_k) = [1 - P_{D,k}(x_k)] v_{k|k-1}(x_k) + \sum\nolimits_{z \in Z_k} \frac{P_{D,k}(x_k) g_k(z|x_k) v_{k|k-1}(x_k)}{\kappa_k(z) + \int P_{D,k} g_k(z|\zeta) v_{k|k-1}(\zeta) \mathrm{d}\zeta} \tag{B.2}$$

Notice that the PHD recursion does not admit closed form solution in general. On one hand, the PHD filter utilizes the Sequential Monte Carlo (SMC) implementation. On the other hand, the Gaussian Mixture (GM) implementation has also been proposed based on the Gaussian sum filters.

B.1 SMC-PHD filter

A generic SMC-PHD implementation for a non-linear-non-Gaussian assumption is described as follows: The posterior intensity function is approximated with a set of particles and their weights at each step. The expected number of the objects is given by the sum of the particle weights. The proposed SMC implementation is a simple application of the standard particle filter [23, 35].

Given an initial intensity function v_0 and the sequence of measurement sets $Z_{1:k}$, the posterior intensity function is calculated.

- Initialization

The intensity function v_0 is initialized with the particle and its weight as

$$\hat{v}_0(x) = \sum_{i=1}^{L_0} w_0^i \delta(x - x_0^i) \tag{B.3}$$

where L_0 and N_0 denotes the amounts of particles and objects at $k = 0$ respectively. $\delta(\cdot)$ is the Dirac delta function.

- Prediction

The intensity function v_{k-1} is approximated by the particles and their weights as

$$\hat{v}_{k-1}(x) = \sum_{i=1}^{L_{k-1}} w_{k-1}^{i} \delta(x - x_{k-1}^{i}) \tag{B.4}$$

where the predicted intensity function $\hat{v}_{k|k-1}(x)$ is given by

$$\hat{v}_{k|k-1}(x) = \sum_{i=1}^{L_{k-1}+J_k} \widetilde{w}_{k|k-1}^{i} \delta(x - \widetilde{x}_{k}^{i}) \tag{B.5}$$

with

$$\widetilde{x}_{k}^{i} \sim \begin{cases} q_k(\cdot|x_{k-1}^{i}, Z_k), & i = 1, ..., L_{k-1} \\ p_k(\cdot|Z_k), & i = L_{k-1}+1, ..., L_{k-1}+J_k \end{cases} \tag{B.6}$$

$$\widetilde{w}_{k|k-1}^{i} = \begin{cases} \frac{\phi_{k|k-1}(\widetilde{x}_{k}^{i}, x_{k-1}^{i})}{q_k(\widetilde{x}_{k}^{i}|x_{k-1}^{i}, Z_k)} w_{k-1}^{i}, & i = 1, ..., L_{k-1} \\ \frac{1}{J_k} \frac{\gamma_k(\widetilde{x}_{k}^{i})}{p_k(\widetilde{x}_{k}^{i}|Z_k)}, & i = L_{k-1}+1, ..., L_{k-1}+J_k \end{cases} \tag{B.7}$$

In $\hat{v}_{k|k-1}$, L_{k-1} particles are predicted with the kernel $\phi_{k|k-1}$. The additional J_k particles are also utilized to detect the new born targets.

- Update

The updated intensity function $\hat{v}_k$ is given by

$$\hat{v}_k(x) = \sum_{i=1}^{L_{k-1}+J_k} \widetilde{w}_{k}^{i} \delta(x - \widetilde{x}_{k}^{i}) \tag{B.8}$$

where

$$\widetilde{w}_{k}^{i} = \left[1 - P_{D,k}(\widetilde{x}_{k}^{i}) + \sum\nolimits_{z \in Z_k} \frac{P_{D,k}(\widetilde{x}_{k}^{i}) g_k(z|\widetilde{x}_{k}^{i})}{\kappa_k(z) + C_k(z)}\right] \widetilde{w}_{k|k-1}^{i} \tag{B.9}$$

$$C_k(z) = \sum_{j=1}^{L_{k-1}+J_k} P_{D,k}(\widetilde{x}_{k}^{j}) g_k(z|\widetilde{x}_{k}^{j}) \widetilde{w}_{k|k-1}^{j} \tag{B.10}$$

- Re-sampling

The new particles and their weights are obtained by re-sampling from the given particles as

$$\left\{\widetilde{x}_{k}^{i}, \widetilde{w}_{k}^{i} / \widetilde{N}_k\right\}_{i=1}^{L_{k-1}+J_k} \tag{B.11}$$

where $\widetilde{N}_k = \sum_{i=1}^{L_{k-1}+J_k} \widetilde{w}_{k}^{i}$ denotes the estimated number of the targets at step k. The sum of the particle weights in the re-sampled set is the same as before since they are scaled by $\widetilde{N}_k$.

The re-sampling step is utilized to avoid the degeneracy since the particle weights might become negligible after a few iterations. After re-sampling, an optional Markov Chain Monte Carlo (MCMC) step is utilized to increase the particle diversity. More details can be found in [23].

Notice that the number of particles does not increase with time since the re-sampling step maintains the number of particles.

B.2 GM-PHD filter

The GM-PHD filter assumes the target motion model, the observation model, and the posterior intensity function are approximated by the sum of Gaussian distributions. An overview of the Gaussian multi-target model for the PHD filter is introduced as follows:

The linear Gaussian models include a number of assumptions on target birth, death as well as detection with its dynamical model.

- Target is described as a linear Gaussian model.

$$f_{k|k-1}(x|\zeta) = \mathcal{N}(x; F_{k-1}\zeta, Q_{k-1}) \tag{B.12}$$

$$g_k(z|x) = \mathcal{N}(z; H_k x, R_k) \tag{B.13}$$

  where x refers to the current state, z to the current measurement, ζ to the previous state, $\mathcal{N}(\cdot\,; m, P)$ denotes the Gaussian distribution with mean m and covariance P, F_{k-1} is the state transition matrix, Q_{k-1} is the process noise covariance, H_k is the observation matrix and R_k is the observation noise covariance.

- Both survival and detection probabilities are independent.

$$P_{S,k}(x) = P_(S, k), \tag{B.14}$$

$$P_{D,k}(x) = P_(D, k), \tag{B.15}$$

  where the target survival and detection probabilities have been derived in [24].

- The birth and spawn RFSs are both Gaussian mixtures.

$$\gamma_k(x) = \sum_{i=1}^{J_{\gamma,k}} \omega_{\gamma,k}^i \mathcal{N}(x; m_{\gamma,k}^i, P_{\gamma,k}^{(i)}) \tag{B.16}$$

$$\beta_{k|k-1}(x|\zeta) = \sum_{j=1}^{J_{\beta,k}} \omega_{\beta,k}^j \mathcal{N}(x; F_{\beta,k-1}^j \zeta + d_{\beta,k-1}^j, Q_{\beta,k-1}^j) \tag{B.17}$$

  where $J_{\gamma,k}, \omega_{\gamma,k}^i, m_{\gamma,k}^i$ and $P_{\gamma,k}^{(i)}$ are given parameters which determine the property of the birth intensity; $J_{\beta,k}, \omega_{\beta,k}^j, F_{\beta,k-1}^j$ and $Q_{\beta,k-1}^j$ determine the spawning intensity of an object with previous state ζ.

According to the dynamical models above, the GM-PHD filter recursion consists of the prediction and update steps as follows

- **Prediction**: Assume the posterior intensity v_{k-1} is a Gaussian mixture form

$$v_{k-1}(x) = \sum_{i=1}^{J_{k-1}} \omega_{k-1}^i \mathcal{N}(x; m_{k-1}^i, P_{k-1}^i) \tag{B.18}$$

  the predicted intensity $v_{k|k-1}$ is also a Gaussian mixture given by

$$v_{k|k-1}(x) = v_{S,k|k-1}(x) + v_{\beta,k|k-1}(x) + \gamma_k(x) \tag{B.19}$$

with

$$v_{S,k|k-1} = P_{S,k} \sum_{j=1}^{J_{k-1}} \omega_{k-1}^{j} \mathcal{N}(x; m_{S,k|k-1}^{j}, P_{S,k|k-1}^{j}) \tag{B.20}$$

$$m_{S,k|k-1}^{j} = F_{k-1} m_{k-1}^{j} \tag{B.21}$$

$$P_{S,k|k-1}^{j} = Q_{k-1} + F_{k-1} P_{k-1}^{j} F_{k-1}^{T} \tag{B.22}$$

$$v_{\beta,k|k-1} = \sum_{j=1}^{J_{k-1}} \sum_{l=1}^{J_{\beta,k}} \omega_{k-1}^{j} \omega_{\beta,k}^{l} \mathcal{N}(x; m_{\beta,k|k-1}^{j,l}, P_{\beta,k|k-1}^{j,l}) \tag{B.23}$$

$$m_{\beta,k|k-1}^{j,l} = F_{k-1}^{l} m_{k-1}^{j} + d_{\beta,k-1}^{l} \tag{B.24}$$

$$P_{\beta,k|k-1}^{j,l} = Q_{k-1}^{l} + F_{\beta,k-1}^{l} P_{\beta,k-1}^{j} (F_{\beta,k-1}^{l})^{T} \tag{B.25}$$

- **Update**: Assume the predicted intensity $v_{k|k-1}$ is a Gaussian mixture form

$$v_{k|k-1} = \sum_{i=1}^{J_{k|k-1}} \omega_{k|k-1}^{i} \mathcal{N}(x; m_{k|k-1}^{i}, P_{k|k-1}^{i}) \tag{B.26}$$

the posterior intensity v_k is therefore given by

$$v_k = (1 - P_{D,k}) v_{k|k-1}(x) + \sum_{z \in Z_k} v_{D,k}(x; z) \tag{B.27}$$

where

$$v_{D,k}(x; z) = \sum_{j=1}^{J_{k|k-1}} \omega_{k}^{j}(z) \mathcal{N}(x; m_{k|k}^{j}(z), P_{k|k}^{j}) \tag{B.28}$$

$$\omega_{k}^{j}(z) = \frac{P_{D,k} \omega_{k|k-1}^{j} q_{k}^{j}(z)}{\kappa_k(z) + P_{D,k} \sum_{l=1}^{J_{k|k-1}} \omega_{k|k-1}^{l} q_{k}^{l}(z)} \tag{B.29}$$

$$q_{k}^{j} = \mathcal{N}(z; H_k m_{k|k-1}^{j}, H_k P_{k|k-1}^{j} (H_k)^{T} + R_k) \tag{B.30}$$

$$m_{k|k}^{j}(z) = m_{k|k-1}^{j} + K_{k}^{j}(z - H_k m_{k|k-1}^{j}) \tag{B.31}$$

$$P_{k|k}^{j} = [I - K_{k}^{j} H_k] P_{k|k-1}^{j} \tag{B.32}$$

$$K_{k}^{j} = P_{k|k-1}^{j} (H_k)^{T} [H_k P_{k|k-1}^{j} (H_k)^{T} + R_k]^{-1} \tag{B.33}$$

The expected number of targets $\hat{N}_k$ and $\hat{N}_{k|k-1}$ are associated with v_k and $v_{k|k-1}$ by following assumptions:

$$\hat{N}_{k|k-1} = \hat{N}_{k-1} \left(p_{S,k} + \sum_{j=1}^{J_{\beta,k}} \omega_{\beta,k}^{j} \right) + \sum_{j=1}^{J_{\gamma,k}} \omega_{\gamma,k}^{j} \tag{B.34}$$

$$\hat{N}_k = \hat{N}_{k|k-1}(1 - p_{D,k}) + \sum_{z \in Z_k} \sum_{j=1}^{J_{k|k-1}} \omega_{k}^{j}(z) \tag{B.35}$$

A simple pseudo-code for GMPHD filter is summarized in Algorithm 2. Although the GM-PHD filter is utilized to deal with linear Gaussian models, the formulation also accommodates non-linear models by linearization or unscented transforms [24].

- Multi-object State Estimation

Although the means of Gaussian terms give the local maxima of the intensity function, it is suggested to chose the means of Gaussian weights bigger than an appropriately chosen threshold as the states.

$$\hat{X}_k = \{m_k^i : \omega_k^i > \omega_t\} \tag{B.36}$$

The pseudo-codes for state estimation is summarized in Algorithm 3. The GM-PHD filer thus avoids the clustering process, compared with the SMC-PHD filter.

- Pruning and Merging Step

The GM-PHD filter also suffers computation problems with respect to the increasing number of Gaussian components. The PHD filter requires a number of $(J_{k-1}(1+J_{\beta,k})+J_{\gamma,k})(1+|Z_k|)$ Gaussian components to represent the intensity function at time k, which implies that the number of components infinitely increasing. The pruning and merging techniques are therefore proposed to reduce the Gaussian components [24]. The proposed solutions can be implemented by either discarding the lower weights components, or keeping those components with strongest weights. When Gaussian components are within the given thresholds, they are automatically merged into a single Gaussian term. The intensity function is thus calculated as

$$\tilde{v}_k(x) = \sum_{i=1}^{\tilde{J}_k} \tilde{\omega}_k^i \mathcal{N}(x; \tilde{m}_k^i, \tilde{P}_k^i) \tag{B.37}$$

The pseudo-codes for pruning and merging step is summarized in Algorithm 4.

B. PROBABILITY HYPOTHESIS DENSITY (PHD) FILTER

Algorithm 2: Pseudo-code For Gaussian Mixture PHD Filter

Given $\{\omega_{k-1}^i, m_{k-1}^i, P_{k-1}^i\}_{i=1}^{J_{k-1}}$ and the measurement set Z_k
Step 1. Birth targets prediction;
i=0 ;
for $j = 1, \cdots, J_{\gamma,k}$ **do**
  $i := i + 1$;
  $\omega_{k|k-1}^i = \omega_{\gamma,k}^j, m_{k|k-1}^i = m_{\gamma,k}^j, P_{k|k-1}^i = P_{\gamma,k}^j$;
for $j = 1, \cdots, J_{\beta,k}$ **do**
  for $l = 1, \cdots, J_{k-1}$ **do**
    $i := i + 1$;
    $\omega_{k|k-1}^i = \omega_{k-1}^l \omega_{\beta,k}^j$;
    $m_{k|k-1}^i = d_{\beta,k-1}^j + F_{\beta,k-1}^j m_{k-1}^l$;
    $P_{k|k-1}^i = Q_{\beta,k-1}^j + F_{\beta,k-1}^j P_{k-1}^l (F_{\beta,k-1}^j)^T$;
Step 2. Existing targets prediction;
for $j = 1, \cdots, J_{k-1}$ **do**
  $\omega_{k|k-1}^i = p_{S,k}\omega_{k-1}^j, m_{k|k-1}^i = F_{k-1} m_{k-1}^j, P_{k|k-1}^i = Q_{k-1} + F_{k-1} P_{k-1}^j (F_{k-1})^T$;
$J_{k|k-1} = i$;
Step 3. Construction of update components
for $j = 1, \cdots, J_{k|k-1}$ **do**
  $\eta_{k|k-1}^j = H_k m_{k|k-1}^j$, $S_k^j = R_k + H_k P_{k|k-1}^j H_k^T$;
  $K_k^j = P_{k|k-1}^j H_k^T [S_k^j]^{-1}, P_{k|k}^j = [I - K_k^j H_k] P_{k|k-1}^j$;
Step 4. Update;
for $j = 1, \cdots, J_{k|k-1}$ **do**
  $\omega_k^j = (1 - p_{D,k})\omega_{k|k-1}^j \quad m_k^j = m_{k|k-1}^j, P_k^j = P_{k|k-1}^j$;
$l := 0$;
for $z \in Z_k$ **do**
  $l := l + 1$;
  for $j = 1, \cdots, J_{k|k-1}$ **do**
    $\omega_k^{(lJ_{k|k-1}+j)} = p_{D,k}\omega_{k|k-1}^j \mathcal{N}(z; \eta_{k|k-1}^j, S_k^j)$;
    $m_k^{(lJ_{k|k-1}+j)} = m_{k|k-1}^j + K_k^j (z - \eta_{k|k-1}^j)$;
    $P_k^{(lJ_{k|k-1}+j)} = P_{k|k}^j$;
  $\omega_k^{(lJ_{k|k-1}+j)} := \frac{\omega_k^{(lJ_{k|k-1}+j)}}{\kappa_k(z) + \sum_{l=1}^{J_{k|k-1}} \omega_k^{(lJ_{k|k-1}+i)}}$, for $j = 1, \cdots, J_{k|k-1}$;
$J_k = lJ_{k|k-1} + J_{k|k-1}$;
Output $\{\omega_k^i, m_k^i, P_k^i\}_{i=1}^{J_k}$

Algorithm 3: Multi-object State Extraction

Given $\{\omega_k^i, m_k^i, P_k^i\}_{i=1}^{J_k}$;
Set $\hat{X}_k = \emptyset$;
for $i = 1, \cdots, J_k$ **do**
    if $\omega_k^i > 0.5$ **then**
        for $j = 1, \cdots, w_k^i$ **do**
            Update $\hat{X}_k = [\hat{X}_k, m_k^i]$
Output $\hat{X}_k$ as the multi-object state estimate.

Algorithm 4: Pruning For Gaussian Mixture PHD Filter

Given $\{\omega_k^i, m_k^i, P_k^i\}_{i=1}^{J_k}$ and the truncation threshold T,the merging threshold U, and the maximum allowable number of Gaussian terms J_{max}.
Step 1. Set $l = 0$, and $I = \{i = 1, \cdots, J_k | \omega_k^i > T\}$;
Step 2. Loop
while $I = \emptyset$ **do**
    $l := l + 1$;
    $j := \arg\max \omega_k^i$;
    $L := \{i \in I | (m_k^i - m_k^j)^T (P_k^i)^{-1} (m_k^i - m_k^j) \leq U\}$;
    $\widetilde{\omega}_k^l = \sum_{i \in L} \omega_k^i$;
    $\widetilde{m}_k^l = \frac{1}{\widetilde{\omega}_k^l} \sum_{i \in L} \omega_k^i x_k^i$;
    $\widetilde{P}_k^l = \frac{1}{\widetilde{\omega}_k^l} \sum_{i \in L} \omega_k^i \{P_k^i + (\widetilde{m}_k^l - m_k^i)(\widetilde{m}_k^l - m_k^i)^T\}$;
    $I := \frac{I}{L}$;
if $l > J_{max}$ **then**
    replace $\{\widetilde{\omega}_k^i, \widetilde{m}_k^i, \widetilde{P}_k^i\}_{i=1}^{l}$ by those of the J_{max} Gaussians with largest weights ;
Output $\{\omega_k^i, m_k^i, P_k^i\}_{i=1}^{l}$ as pruned Gaussian components.

Appendix C

Euler Rotation Equation

In Cartesian coordinate system, an elemental rotation is considered as a rotation matrix that rotates vectors by the angle α, β and γ in three axes using the right hand rule:

$$R_x(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

$$R_z(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

where

$$R = R_z(\alpha)R_y(\beta)R_x(\gamma) \tag{C.1}$$

represents a rotation in which yaw, pitch, row angles are α, β and γ respectively. The adopted Euler angles move the reference frame to the referred frame, following the Z-Y-X rotation sequence.

Appendix D

The RANSAC Approach

The RANdom SAmple Consensus (RANSAC) approach is a parameter estimation algorithm to remove outliers in the given dataset.

The RANSAC approach is a learning technique to model the estimations by random sampling the observed data. Given a dataset which includes both inliers and outliers, RANSAC approach uses a voting scheme to search the optimal fitting models. The implementation of such voting scheme is based on the following assumptions:

- For any single model, the noisy features would not vote consistently.
- There should be enough features to estimate a good model.

Thus the RANSAC algorithm is summarized as follows:

Algorithm 5: RANSAC Algorithm

1. Randomly select the minimum number of points to determine the model parameters ;
2. Calculate the parameters for the model;
3. Calculate the number of the points to fit with a predefined tolerance ϵ ;
4. If the fraction of the number of inliers over the number of the whole dataset exceeds the given threshold τ, re-estimate the corresponding parameters by all the identified inliers and terminate ;
5. Otherwise, repeats steps 1 through 4 (Maximum of N times).

In RANSAC, the maximum number of iterations has to be chosen high enough to ensure that at least one of the sets of samples does not contain an outlier. Let u denotes the probability that any selected pint is an inlier and $v = 1 - u$ thus denotes the probability of the outlier. N iterations of the minimum number of points denoted as m where

$$1 - p = (1 - u^m)^N$$

thus N is calculated as

$$N = \frac{\log(1 - p)}{\log(1 - (1 - v)^m)}$$

Appendix E

Singular Value Decomposition (SVD)

Given $P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_n\}$ and $Q = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_n\}$ as two sets of points, a rigid transformation which optimally aligns the given sets is represented as follows

$$(R, T) = \underbrace{\arg\min}_{R,T} \sum_{i=1}^{n} w_i ||(R\mathbf{p}_i + T) - \mathbf{q}_i||^2$$

where $w_i > 0$ denotes the corresponding weight for each point pair.

Based on the least square method, the SVD for motion computation is calculated as follows:

Algorithm 6: Least Square SVD

1 Compute the weighted centroids of both sets:

$$\overline{\mathbf{p}} = \frac{\sum_{i=1}^{n} w_i \mathbf{p}_i}{\sum_{i=1}^{n} w_i}, \; \overline{\mathbf{q}} = \frac{\sum_{i=1}^{n} w_i \mathbf{q}_i}{\sum_{i=1}^{n} w_i}$$

2 Compute the centered vectors

$$\mathbf{x}_i := \mathbf{p}_i - \overline{\mathbf{p}}, \; \mathbf{y}_i := \mathbf{q}_i - \overline{\mathbf{q}}, \; i = 1, 2, ...n.$$

3 Compute the covariance matrix

$$S = XWY^T$$

where X and Y denote the $d \times n$ matrices that have $\mathbf{x}_i$ and $\mathbf{y}_i$ as their columns and $W = \text{diag}(w_1, w_2, ...w_n)$.

4 Compute the singular value decomposition $S = U \sum V^T$. The rotation is thus represented as

$$R = V \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & & \text{def}(VU^T) \end{pmatrix} U^T$$

5 Compute the optimal translation as

$$T = \overline{\mathbf{q}} - R\overline{\mathbf{p}}.$$

Appendix F

Trajectory reconstruction based on dead reckoning

In Sec. 3.1, the Visual Odometry (VO) system was proposed to estimate the ego-motion vector. By accumulating the estimations, the vehicle's trajectory is acquired.

Assuming

$$\mathbf{P}_k = [X_k^0, Y_k^0, Z_k^0]^T \tag{F.1}$$

refers the current location of the vehicle denoted in coordinates of the global coordinate system in 3D representation. It is deduced that the vehicle position at a given time is calculated by integrating the translational vector:

$$\mathbf{P}_k = \mathbf{T}_{0\to k}^0 = \mathbf{T}_{0\to k-1}^0 + \mathbf{T}_{k-1\to k}^0 = \mathbf{P}_{k-1} + \mathbf{T}_{k-1\to k}^0 \tag{F.2}$$

where it is emphasized that the translational vector is denoted in coordinates of the reference system. In Sec. 3.1, the ego-motion vector consists of the rotation matrix $\mathbf{R}_{k-1}^k$ and the translational vector $\mathbf{T}_{k\to k-1}^k$, which describe the relative displacement between two consecutive steps:

$$\mathbf{P}_k = \mathbf{R}_{k-1}^k \mathbf{P}_{k-1} + \mathbf{T}_{k\to k-1}^k \tag{F.3}$$

Thus the location $\mathbf{P}_k$ is represented as a function of the ego-motion vector ($\mathbf{R}_{k-1}^k$ and $\mathbf{T}_{k\to k-1}^k$) that corresponding to the previous step. Furthermore, the global translational vector is expressed as a function of the estimated parameters:

$$\mathbf{T}_{0\to 1}^0 = \mathbf{R}_1^0 \mathbf{T}_{0\to 1}^1 = -\left[\mathbf{R}_0^1\right]^t \mathbf{T}_{1\to 0}^1$$

$$\mathbf{T}_{1\to 2}^0 = \mathbf{R}_2^0 \mathbf{T}_{1\to 2}^2 = -\mathbf{R}_1^0 \mathbf{R}_2^1 \mathbf{T}_{2\to 1}^2 = -\left[\mathbf{R}_0^1\right]^t \left[\mathbf{R}_1^2\right]^t \mathbf{T}_{2\to 1}^2 = -\left[\mathbf{R}_0^2\right]^t \mathbf{T}_{2\to 1}^2$$

$$\mathbf{T}_{2\to 3}^0 = \mathbf{R}_3^0 \mathbf{T}_{2\to 3}^3 = -\left[\mathbf{R}_0^1\right]^t \left[\mathbf{R}_1^2\right]^t \left[\mathbf{R}_2^3\right]^t \mathbf{T}_{3\to 2}^3 = -\left[\mathbf{R}_0^2\right]^t \left[\mathbf{R}_2^3\right]^t \mathbf{T}_{3\to 2}^3 = -\left[\mathbf{R}_0^3\right]^t \mathbf{T}_{3\to 2}^3$$

$$\vdots$$

$$\begin{aligned}\mathbf{T}_{k-1\to k}^0 = \mathbf{R}_k^0 \mathbf{T}_{k-1\to k}^k &= -\left(\prod_{j=1}^{k-1}\left[\mathbf{R}_j^{j+1}\right]^t\right)\mathbf{T}_{1\to k-1}^k = \cdots \\ &= -\left[\mathbf{R}_0^{k-1}\right]^t \left[\mathbf{R}_{k-1}^k\right]^t \mathbf{T}_{k\to k-1}^k = -\left[\mathbf{R}_0^k\right]^t \mathbf{T}_{k\to k-1}^k\end{aligned} \tag{F.4}$$

Similarly, the vehicle's position at step k yields as:

$$\mathbf{P}_k = \mathbf{P}_{k-1} + \mathbf{T}^0_{k-1\to k} = \mathbf{P}_{k-1} - \left[\mathbf{R}^k_0\right]^t \mathbf{T}^k_{k\to k-1} = \mathbf{P}_{k-1} - \left[\mathbf{R}^{k-1}_0\right]^t \left[\mathbf{R}^k_{k-1}\right]^t \mathbf{T}^k_{k\to k-1} \quad \text{(F.5)}$$

Thus the position $\mathbf{P}_k$ is computed based on the coordinates of previous point $\mathbf{P}_{k-1}$ by subtracting a term which includes the cumulative value form both orientation and translation between consecutive steps. Notes that the term $\left[\mathbf{R}^k_0\right]^t = \left[\mathbf{R}^{k-1}_0\right]^t \left[\mathbf{R}^k_{k-1}\right]^t$ represents the orientation and $\mathbf{T}^k_{k\to k-1}$ represents the translation. Therefore, the global position is computed by the recursive equations as:

$$\begin{cases} \left[\mathbf{R}^k_0\right]^t = \left[\mathbf{R}^{k-1}_0\right]^t \left[\mathbf{R}^k_{k-1}\right]^t \\ \mathbf{P}_k = \mathbf{P}_{k-1} - \left[\mathbf{R}^k_0\right]^t \mathbf{T}^k_{k\to k-1}; \quad k = 1, 2, \cdots, N \end{cases} \quad \text{(F.6)}$$

where

$$\mathbf{P}_0 = \mathbf{0}, \qquad \mathbf{R}_0 = \mathbf{I}$$

References

[1] Wikipedia. **Geodetic system — Wikipedia, The Free Encyclopedia**, 2012. [accessed 06-March-2012]. v, 20

[2] **http://www.who.int/gho/road safety/mortality/en/index.html**. 1

[3] Karl Iagnemma and Martin Buehler. **Editorial for Journal of Field Robotics, Special Issue on the DARPA grand challenge**. *Journal of Field Robotics*, **23**(8):461–462, 2006. 1

[4] **http://www.google.com/selfdrivingcar/**. 1

[5] Erico Guizzo. **How google self driving car works**. *IEEE Spectrum Online, October*, **18**, 2011. 1

[6] A. Vahidi and A. Eskandarian. **Research advances in intelligent collision avoidance and adaptive cruise control**. *Intelligent Transportation Systems, IEEE Transactions on*, **4**(3):143–153, Sept 2003. 1

[7] Parag H. Batavia. *Driver-Adaptive Lane Departure Warning Systems*. PhD thesis, The Robotics Institute Carnegie Mellon University, September 1999. 1

[8] R. Okuda, Y. Kajiwara, and K. Terashima. **A survey of technical trend of ADAS and autonomous driving**. In *VLSI Technology, Systems and Application (VLSI-TSA), Proceedings of Technical Program - 2014 International Symposium on*, pages 1–4, April 2014. 1

[9] Lu M, Wevers K, and Van der Heuden R. **Technical feasibility of advanced driver assistance systems (ADAS) for road traffic safety**. *TRANSPORTATION PLANNING AND TECHNOLOGY*, **3**(28):167–187, 2005. 2

[10] TI. **Advanced Driver Assistance (ADAS) Solutions Guide**, 2015. 3

[11] Ba Tuong Vo. *Random Finite Sets in Multi-Object Filtering*. PhD thesis, School of Electrical, Electronic and Computer Engineering, THE UNIVERSITY OF WESTERN AUSTRALIA, October 2008. 3

[12] Y. Bar-Shalom and X.R.Li. *Tracking and Data Association*. Academic Press, San Diego, 1988. 3, 15

[13] S.S. Blackman. **Multiple hypothesis tracking for multiple target tracking**. *Aerospace and Electronic Systems Magazine, IEEE*, **19**(1):5–18, Jan 2004. 3

[14] Feihu Zhang, Christian Buckl, and Alois Knoll. **Multiple Vehicle Cooperative Localization with Spatial Registration Based on a Probability Hypothesis Density Filter**. *Sensors*, **14**(1):995–1009, 2014. 4

[15] Melanie Anne Edith Bocquel. *Random Finite Sets in Multi-target Tracking*. PhD thesis, University of Twente, 2013. 4

[16] R. E. Kalman. **A New Approach to Linear Filtering and Prediction Problems**. *Journal of Fluids Engineering*, pages 35–45, 1960. 4, 10, 79

[17] I. Goodman, R. Mahler, and H. Nguyen. *Mathematics of Data Fusion*. Kluwer Academic Publishers, Dordrecht/Boston/London, 1995. 4, 5, 10, 11, 13

[18] D. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Process*. Springer Verlag, Berlin, Germany, 1988. 5, 11

[19] R. Mahler. *An introduction to multisource-multitarget statistics and its applications*. Lockheed Martin Technical Monograph, March 2000. 5, 11, 13

[20] R.P.S. Mahler. **"Statistics 101" for multisensor, multitarget data fusion**. *Aerospace and Electronic Systems Magazine, IEEE*, **19**(1):53–64, Jan 2004. 5, 11, 13

[21] R.P.S. Mahler. **Multitarget Bayes filtering via first-order multitarget moments**. *Aerospace and Electronic Systems, IEEE Transactions on*, **39**(4):1152–1178, Oct 2003. 5, 11

[22] Hedvig Sidenbladh. **Multi-target particle filtering for the probability hypothesis density**. In *Information Fusion, 2003. Proceedings of the Sixth International Conference of*, **2**, pages 800–806, July 2003. 5, 14

[23] Ba-Ngu Vo, Sumeetpal Singh, and A. Doucet. **Sequential monte carlo implementation of the phd filter for multi-target tracking**. In *Information Fusion, 2003. Proceedings of the Sixth International Conference of*, **2**, pages 792–799, July 2003. 5, 14, 85, 86

[24] Ba-Ngu Vo and Wing-Kin Ma. **The Gaussian Mixture Probability Hypothesis Density Filter**. *Signal Processing, IEEE Transactions on*, **54**(11):4091–4104, Nov 2006. 5, 14, 87, 88, 89

[25] Ba-Ngu Vo and Wing-Kin Ma. **A closed-form solution for the probability hypothesis density filter**. In *Information Fusion, 2005 8th International Conference on*, **2**, pages 8 pp.–, July 2005. 5, 14

[26] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. **A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking**. *Signal Processing, IEEE Transactions on*, **50**(2):174–188, Feb 2002. 10, 11, 82

[27] Simon Maskell. *Sequentially Structured Bayesian Solutions*. PhD thesis, Engineering Department, Cambridge University, 2004. 10

[28] B. Anderson and J. Moore. *Optimal Filtering*. Prentice Hall, Englewood, NJ, 1979. 10

[29] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. Academic Press, New York, 1970. 10

[30] E. A. Wan and R. Van der Merwe. **The unscented kalman filter for nonlinear estimation.** In *Symposium of Adaptive Systems, Signal Processing, Communications and Control*, Lake Louise, AB, Canada, 2000. 10

[31] S. J. Godsill A. Doucet and C. Andrieu. **On sequential Monte Carlo methods for Bayesian filtering**. *Stat. Comp.*, **10**:197–208, 2000. 11, 82

[32] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. **Novel approach to nonlinear/non-Gaussian Bayesian state estimation**. *Radar and Signal Processing, IEE Proceedings F*, **140**(2):107–113, Apr 1993. 11, 82, 83

[33] S. Arulampalam B. Ristic and N. Gordon. *Beyond Kalman Filter: Particle Filters for Tracking Applications*. Artech House, Boston/London, 2004. 11, 82, 83

[34] Ronald P. S. Mahler. *Advances in Statistical Multisource-Multitarget Information Fusion*. Artech House, 2014. 13

[35] Ba-Ngu Vo, Sumeetpal Singh, and Arnaud Doucet. **Sequential Monte Carlo methods for multitarget filtering with random finite sets**. *Aerospace and Electronic Systems, IEEE Transactions on*, **41**(4):1224–1245, Oct 2005. 13, 14, 85

[36] R.P.S. Mahler, Ba-Ngu Vo, and Ba-Tuong Vo. **The forward-backward Probability Hypothesis Density smoother**. In *Information Fusion (FUSION), 2010 13th Conference on*, pages 1–8, July 2010. 13

[37] R. Mahler. **Approximate multisensor CPHD and PHD filters**. In *Information Fusion (FUSION), 2010 13th Conference on*, pages 1–8, July 2010. 13

[38] R. Mahler. **Engineering statistics for multi-object tracking**. In *Multi-Object Tracking, 2001. Proceedings. 2001 IEEE Workshop on*, pages 53–60, 2001. 13

[39] R. Mahler. **PHD filters for nonstandard targets, I: Extended targets**. In *Information Fusion, 2009. FUSION '09. 12th International Conference on*, pages 915–921, July 2009. 13

[40] D.E. Clark and J. Bell. **Data Association for the PHD Filter**. In *Intelligent Sensors, Sensor Networks and Information Processing Conference, 2005. Proceedings of the 2005 International Conference on*, pages 217–222, Dec 2005. 14

[41] L. Lin, Y. Bar-Shalom, and T. Kirubarajan. **Track labeling and PHD filter for multitarget tracking**. *Aerospace and Electronic Systems, IEEE Transactions on*, **42**(3):778–795, July 2006. 14

[42] K. Panta, Ba-Ngu Vo, and Sumeetpal Singh. **Novel data association schemes for the probability hypothesis density filter**. *Aerospace and Electronic Systems, IEEE Transactions on*, **43**(2):556–570, April 2007. 14

[43] K. Panta, B. Vo, and S. Singh. **Improved Probability Hypothesis Density (PHD) Filter for Multitarget Tracking**. In *Intelligent Sensing and Information Processing, 2005. ICISIP 2005. Third International Conference on*, pages 213–218, Dec 2005. 14

[44] K. Panta, Ba-Ngu-Vo, and D.E. Clark. **An Efficient Track Management Scheme for the Gaussian-Mixture Probability Hypothesis Density Tracker**. In *Intelligent Sensing and Information Processing, 2006. ICISIP 2006. Fourth International Conference on*, pages 230–235, Oct 2006. 14

[45] D.B. Reid. **An algorithm for tracking multiple targets**. *Automatic Control, IEEE Transactions on*, **24**(6):843–854, Dec 1979. 15

[46] Kiyoshi Hirose, Hitoshi Doki, and Akiko Kondo. **A calibration method of magnetic field sensor for body motion measurement using Extended Kalman filer**. In *SICE Annual Conference (SICE), 2013 Proceedings of*, pages 72–77, Sept 2013. 15

[47] Chi Ming Cheuk, Tak Kit Lau, Kai Wun Lin, and Yunhui Liu. **Automatic calibration for inertial measurement unit**. In *Control Automation Robotics Vision (ICARCV), 2012 12th International Conference on*, pages 1341–1346, Dec 2012. 15

[48] Rui Zhang, F. Hoflinger, and L.M. Reind. **Calibration of an IMU Using 3-D Rotation Platform**. *Sensors Journal, IEEE*, **14**(6):1778–1787, June 2014. 15

[49] Rashmi Bajaj, Samantha Lalinda Ranaweera, and Dharma P Agrawal. **GPS: location-tracking technology**. *Computer*, **35**(4):92–94, 2002. 19

[50] Neil Boasman, Daniel Clarke, Steve Davison, and Roger Stokes. **Advanced Test Methods for Integrated Navigation Systems**. *Royal Institute of Navigation*, 2005. 19

[51] R. Lerner, E. Rivlin, and H.P. Rotstein. **Pose and motion recovery from feature correspondences and a digital terrain map**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **28**(9):1404 –1417, sept. 2006. 19

[52] S. Laible and A. Zell. **Building local terrain maps using spatio-temporal classification for semantic robot localization**. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 4591–4597, Sept 2014. 19

[53] Yang Yang, Miao Jin, and Hongyi Wu. **3D surface localization with terrain model**. In *INFOCOM, 2014 Proceedings IEEE*, pages 46–54, April 2014. 19

[54] Tianyi Li, Ming Yang, Liuyuan Deng, Yong He, and Chunxiang Wang. **An extended probabilistic self-localization algorithm using hybrid maps**. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 81–86, Oct 2014. 19

[55] M. Kais, S. Dauvillier, A. Foretllea, I. Masakid, and C. Laugier. **Towards outdoor localization using GIS, vision system and stochastic error propagation**. In *International Conference on Autonomous Robots and Agents*, 2004. 19

[56] Seung-Beom Han, Jong-Hwan Kim, and Hyun Myung. **Landmark-Based Particle Localization Algorithm for Mobile Robots With a Fish-Eye Vision System**. *Mechatronics, IEEE/ASME Transactions on*, **18**(6):1745–1756, Dec 2013. 19

[57] L. Jayatilleke and Nian Zhang. **Landmark-based localization for Unmanned Aerial Vehicles**. In *Systems Conference (SysCon), 2013 IEEE International*, pages 448–451, April 2013. 19

[58] Jae-Yeong Lee and Wonpil Yu. **Robust self-localization of ground vehicles using artificial landmark**. In *Ubiquitous Robots and Ambient Intelligence (URAI), 2014 11th International Conference on*, pages 303–307, Nov 2014. 19

[59] P. Vorst and A. Zell. **Fully autonomous trajectory estimation with long-range passive RFID**. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 1867 –1872, may 2010. 19

[60] C.F. Olson, L.H. Matthies, M. Schoppers, and M.W. Maimone. **Stereo ego-motion improvements for robust rover navigation**. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, **2**, pages 1099 – 1104 vol.2, 2001. 19

[61] G. Antonelli, S. Chiaverini, and G. Fusco. **An odometry calibration method for mobile robots based on the least-squares technique**. In *American Control Conference, 2003. Proceedings of the 2003*, **4**, pages 3429–3434, June 2003. 19

[62] M. Agrawal and K. Konolige. **Real-time Localization in Outdoor Environments using Stereo Vision and Inexpensive GPS**. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, **3**, pages 1063 –1068, 0-0 2006. 19

[63] S. Umeyama. **Least-squares estimation of transformation parameters between two point patterns**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **13**(4):376 –380, apr 1991. 23

[64] Feihu Zhang, Stahle Hauke, Guang Chen, Chao Chen, Simon Carsten, Christian Buckl, and A. Knoll. **A sensor fusion approach for localization with cumulative error elimination**. In *Multisensor Fusion and Integration for Intelligent Systems (MFI), 2012 IEEE Conference on*, pages 1–6, 2012. 25, 26

[65] B. Kitt, A. Geiger, and H. Lategahn. **Visual odometry based on stereo image sequences with RANSAC-based outlier rejection scheme**. In *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 486 –492, june 2010. 27, 60

[66] H. Wymeersch, J. Lien, and M.Z. Win. **Cooperative Localization in Wireless Networks**. *Proceedings of the IEEE*, **97**(2):427–450, Feb 2009. 31

[67] S.I. Roumeliotis and I.M. Rekleitis. **Analysis of multirobot localization uncertainty propagation**. In *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, **2**, pages 1763 – 1770 vol.2, oct. 2003. 32

[68] A. Howard, M.J. Mataric, and G.S. Sukhatme. **Putting the 'I' in 'team': an ego-centric approach to cooperative localization**. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, **1**, pages 868 – 874 vol.1, sept. 2003. 32

[69] D. Fox, W. Burgard, and S. Thrun. **Markov localization for mobile robots in dynamics environments**. *Journal of Artificial Intelligence Research*, **11**:391–427, 1999. 32

[70] E.D. Nerurkar, S.I. Roumeliotis, and A. Martinelli. **Distributed maximum a posteriori estimation for multi-robot cooperative localization**. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 1402 –1409, may 2009. 32

[71] M.D.P. Moratuwage, W.S. Wijesoma, B. Kalyan, N.M. Patrikalakis, and P. Moghadam. **Collaborative multi-vehicle localization and mapping in high clutter environments**. In *Control Automation Robotics Vision (ICARCV), 2010 11th International Conference on*, pages 1422 –1427, dec. 2010. 32

[72] Esha D. Nerurkar, Ke X. Zhou, and Stergios I. Roumeliotis. **A hybrid estimation framework for Cooperative Localization under communication constraints**. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 502 –509, sept. 2011. 32

[73] Nadir Karam, Frederic Chausse, Romuald Aufrere, and Roland Chapuis. **Localization of a Group of Communicating Vehicles by State Exchange**. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 519 –524, oct. 2006. 32, 35

[74] Hao Li and F. Nashashibi. **Cooperative multi-vehicle localization using split covariance intersection filter**. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 211 –216, june 2012. 32

[75] N. Karam, F. Chausse, R. Aufrere, and R. Chapuis. **Cooperative Multi-Vehicle Localization**. In *Intelligent Vehicles Symposium, 2006 IEEE*, pages 564 –570, 0-0 2006. 32

[76] S.I. Roumeliotis and G.A. Bekey. **Distributed multirobot localization**. *Robotics and Automation, IEEE Transactions on*, **18**(5):781 – 795, oct 2002. 32

[77] W. Li and H. Leung. **Simultaneous registration and fusion of multiple dissimilar sensors for cooperative driving**. *Intelligent Transportation Systems, IEEE Transactions on*, **5**(2):84–98, June 2004. 38

[78] W. Li, H. Leung, and Yifeng Zhou. **Space-time registration of radar and ESM using unscented Kalman filter**. *Aerospace and Electronic Systems, IEEE Transactions on*, **40**(3):824–836, July 2004. 38

[79] Huang Jianjun, Zhong Jiali, and Jiang Feng. **A CKF based spatial alignment of radar and infrared sensors**. In *Signal Processing (ICSP), 2010 IEEE 10th International Conference on*, pages 2386–2390, Oct 2010. 38

[80] Zhang Xin-hua, Guo Hui-dong, and Xia Zhi-jun. **An Asynchronous Multisensor Spatial Registration Algorithm**. In *Fuzzy Systems and Knowledge Discovery, 2007. FSKD 2007. Fourth International Conference on*, **4**, pages 16–20, Aug 2007. 38

[81] P. Furgale, J. Rehder, and R. Siegwart. **Unified temporal and spatial calibration for multi-sensor systems**. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 1280–1286, Nov 2013. 38

[82] F. Lian, C. Han, W. Liu, and H. Chen. **Joint spatial registration and multi-target tracking using an extended probability hypothesis density filter**. *Radar, Sonar Navigation, IET*, **5**(4):441–448, 2011. 38

[83] Anthony Kim and MF Golnaraghi. **Initial calibration of an inertial measurement unit using an optical position tracking system**. In *Position Location and Navigation Symposium, 2004. PLANS 2004*, pages 96–101. IEEE, 2004. 45

[84] C. McCarthy and N. Barnes. **Performance of optical flow techniques for indoor navigation with a mobile robot**. In *IEEE International Conference on Robotics and Automation, ICRA '04*, pages 5093 – 5098, Vol. 5, May 2004. 45

[85] J. Campbell, R. Sukthankar, and I. Nourbakhsh. **Techniques for evaluating optical flow for visual odometry in extreme terrain**. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2004*, pages 3704 – 3711, Vol. 4, Sept. 2004. 45

[86] P. Corke, D. Strelow, and S. Singh. **Omnidirectional visual odometry for a planetary rover**. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2004*, pages 4007 – 4012, Vol. 4, Oct. 2004. 45

[87] S. Benoit and F.P. Ferrie. **Towards direct recovery of shape and motion parameters from image sequences**. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1395–1402 vol.2, Oct 2003. 45

[88] R.G. Garcia, M.A. Sotelo, I. Parra, D. Fernandez, and M. Gavilan. **2D Visual Odometry method for Global Positioning Measurement**. In *IEEE International Symposium on Intelligent Signal Processing, WISP 2007*, pages 1 –6, Oct. 2007. 46

[89] Moravec H. *Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover*. PhD thesis, Stanford University, 1980. 46

[90] E. Rosten and T. Drummond. **Machine learning for high-speed corner detection**. In *European Conference on Computer Vision*, **1**, pages 430–443. 46

[91] D.G. Lowe. **Object recognition from local scale-invariant features**. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 1150 –1157, Vol. 2, 1999. 46, 47

[92] Herbert Bay., Tinne Tuytelaars., and Luc Van Gool. **SURF: Speeded Up Robust Features**. In *9th European Conference on Computer Vision*, pages 404–417, May 7-13 2006. 46, 57

[93] C. Harris and M.J. Stephens. **A combined corner and edge detector**. In *Alvey Vision Conference*, pages 147–152, 1988. 46

[94] C. Tomasi and T. Kanade. **Detection and tracking of point features**. Technical report, Carnegie Mellon, April 1991. 46

[95] O. Chum and J. Matas. **Optimal Randomized RANSAC**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **30**(8):1472 –1482, aug. 2008. 46

[96] Motilal Agrawal Kurt Konolige and Joan Sola. **Large Scale Visual Odometry for Rough Terrain**. In *International Symposium on Research in Robotics (ISRR)*. 46

[97] Chi Hay Tong and T.D. Barfoot. **Gaussian Process Gauss-Newton for 3D laser-based Visual Odometry**. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 5204–5211, May 2013. 46

[98] Hyukdoo Choi, Kwang Woong Yang, and Euntai Kim. **Simultaneous Global Localization and Mapping**. *Mechatronics, IEEE/ASME Transactions on*, **19**(4):1160–1170, Aug 2014. 46

[99] D. Nister, O. Naroditsky, and J. Bergen. **Visual odometry**. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004*, pages I–652 – I–659, Vol. 1, July 2004. 47

[100] D. Burschka and G.D. Hager. **V-GPS(SLAM): vision-based inertial system for mobile robots**. In *2004 IEEE International Conference on Robotics and Automation*, pages 409 – 415, Vol.1, May 2004. 47

[101] A. Howard. **Real-time stereo visual odometry for autonomous ground vehicles**. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 3946–3952, Sept 2008. 47

[102] T. Koletschka, L. Puig, and K. Daniilidis. **MEVO: Multi-environment stereo visual odometry**. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 4981–4988, Sept 2014. 47

[103] M. Persson, T. Piccini, M. Felsberg, and R. Mester. **Robust stereo visual odometry from monocular techniques**. In *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pages 686–691, June 2015. 47

[104] D. Scaramuzza and R. Siegwart. **Appearance-Guided Monocular Omnidirectional Visual Odometry for Outdoor Ground Vehicles**. *IEEE Transactions on Robotics*, (5):1015 –1026, Vol. 24, Oct. 2008. 47

[105] Zhengyou Zhang. **Flexible camera calibration by viewing a plane from unknown orientations**. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, **1**, pages 666 –673 vol.1, 1999. 48

[106] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 48

[107] **OpenStreetMap database**. [web page] http://www.openstreetmap.org/. 51

[108] Daniel Scharstein and Richard Szeliski. **A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms**. *International Journal of Computer Vision*, pages 7–42, 2002. 59

[109] Xun Dai, A. Kummert, Su Birm Park, and D. Neisius. **A warning algorithm for Lane Departure Warning system**. In *Intelligent Vehicles Symposium, 2009 IEEE*, pages 431 –435, 2009. 65

[110] M. Bertozzi and A. Broggi. **GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection**. *Image Processing, IEEE Transactions on*, **7**(1):62 –81, January 1998. 65, 67

[111] Albert S. Huang. *Lane Estimation for Autonomous Vehicles using Vision and LIDAR*. PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, February 2010. 65

[112] M. Meuter, S. Muller-Schneiders, A. Mika, S. Hold, C. Nunny, and A. Kummert. **A novel approach to lane detection and tracking**. In *Intelligent Transportation Systems, 2009. ITSC '09. 12th International IEEE Conference on*, pages 1–6, Oct 2009. 65

[113] T. Veit, J.-P. Tarel, P. Nicolle, and P. Charbonnier. **Evaluation of Road Marking Feature Extraction**. In *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*, pages 174 –181, oct. 2008. 65

[114] J.C. McCall and M.M. Trivedi. **Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation**. *Intelligent Transportation Systems, IEEE Transactions on*, **7**(1):20–37, March 2006. 65

[115] J.C. McCall, D.P. Wipf, M.M. Trivedi, and B.D. Rao. **Lane Change Intent Analysis Using Robust Operators and Sparse Bayesian Learning**. *Intelligent Transportation Systems, IEEE Transactions on*, **8**(3):431–440, Sept 2007. 65

[116] Kuo-Yu Chiu and Sheng-Fuu Lin. **Lane detection using color-based segmentation**. In *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, pages 706–711, June 2005. 65

[117] Azali Saudi, Jason Teo, Mohd Hanafi Ahmad Hijazi, and Jumat Sulaiman. **Fast lane detection with Randomized Hough Transform**. In *Information Technology, 2008. ITSim 2008. International Symposium on*, **4**, pages 1 –5, 2008. 65

[118] J.C. McCall and M.M. Trivedi. **An integrated, robust approach to lane marking detection and lane tracking**. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 533 – 537, june 2004. 65

[119] A.A.M. Assidiq, O.O. Khalifa, R. Islam, and S. Khan. **Real time lane detection for autonomous vehicles**. In *Computer and Communication Engineering, 2008. ICCCE 2008. International Conference on*, pages 82 –88, may 2008. 65

[120] A. Sarafraz F. Samadzadegan and M. Tabib. **Automatic Lane Detection in Image Sequences for Vision-Based Navigation Purposes**. In *Proceedings of the ISPRS Commission V Symposium 'Image Engineering and Vision Metrology'*, 2006. 67

[121] A. Takahashi and Y. Ninomiya. **Model-based lane recognition**. In *Intelligent Vehicles Symposium, 1996., Proceedings of the 1996 IEEE*, pages 201 –206, September 1996. 67

[122] P. Coulombeau and C. Laurgeau. **Vehicle yaw, pitch, roll and 3D lane shape recovery by vision**. In *Intelligent Vehicle Symposium, 2002. IEEE*, **2**, pages 619 – 625 vol.2, june 2002. 67

[123] ZuWhan Kim. **Robust Lane Detection and Tracking in Challenging Scenarios**. *Intelligent Transportation Systems, IEEE Transactions on*, **9**(1):16 –26, 2008. 67

[124] S. Lakshmanan and D. Grimmer. **A deformable template approach to detecting straight edges in radar images**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **18**(4):438–443, Apr 1996. 67

[125] Dong Jung Kang, Jang Won Choi, and In So Kweon. **Finding and tracking road lanes using ldquo;line-snakes rdquo;**. In *Intelligent Vehicles Symposium, 1996., Proceedings of the 1996 IEEE*, pages 189–194, Sep 1996. 67

[126] Bing Ma, S. Lakshmanan, and A.O. Hero. **Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion**. *Intelligent Transportation Systems, IEEE Transactions on*, **1**(3):135–147, Sep 2000. 67

[127] Jianfeng Wang, Fangde Gu, Chao Zhang, and Guanzhe Zhang. **Lane boundary detection based on parabola model**. In *Information and Automation (ICIA), 2010 IEEE International Conference on*, pages 1729–1734, June 2010. 67

[128] Shengyan Zhou, Yanhua Jiang, Junqiang Xi, Jianwei Gong, Guangming Xiong, and Huiyan Chen. **A novel lane detection based on geometrical model and Gabor filter**. In *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 59 –64, 2010. 67

[129] B. Fardi, U. Scheunert, H. Cramer, and G. Wanielik. **A new approach for lane departure identification**. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pages 100–105, June 2003. 67

[130] NOBUYUKI Otsu. **A Threshold Selection Method from Gray-Level Histograms**. *Systems, Man and Cybernetics, IEEE Transactions on*, **9**(1):62 –66, jan. 1979. 67

[131] W.N.Lu and Z.K.Shi. **Synchronous Detection of the Lane Marking and Road Boundary on Monocular Vision**. *Chinese Journal of Sensors and Actuators*, **20**:1171 – 1175, May 2007. 68

[132] Wikipedia. **Lane — Wikipedia, The Free Encyclopedia**, 2015. [Online; accessed 11-June-2015]. 71

[133] Arnaud Doucet, N.J. Gordon, and V. Krishnamurthy. **Particle filters for state estimation of jump Markov linear systems**. *Signal Processing, IEEE Transactions on*, **49**(3):613–624, Mar 2001. 83

[134] N. Chopin. **A sequential particle filter method for static models**. *Biometrika*, **89**(3):539–51, Aug 2002. 83

[135] N. de Freitas A. Doucet and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001. 83, 84

[136] J.P. Villa and V. Rossi. **Nonlinear filtering in discrete time: A particle convolution approach**. Technical report, Biostatistic Group of Montpellier, Apr 2004. 83

[137] J. Akashi and H. Kumamoto. **Random sampling approach to state estimation in switching environments**. *Automatica*, **13**:429–434, 1977. 84

[138] Dong Guo and Xiaodong Wang. **Quasi-Monte Carlo filtering in nonlinear dynamic systems**. *Signal Processing, IEEE Transactions on*, **54**(6):2087–2098, June 2006. 84