ulm university universität
uulm

Intersection-Based Road User Tracking Using a Classifying-Multiple-Model PHD Filter

Daniel Meißner

Band 15

Schriftenreihe des Instituts für Mess-, Regel- und Mikrotechnik

Daniel Alexander Meißner

**Intersection-Based Road User Tracking Using a Classifying
Multiple-Model PHD Filter**

Schriftenreihe des
Instituts für Mess-, Regel- und Mikrotechnik
Universität Ulm

Herausgeber:
Prof. Dr.-Ing. Klaus Dietmayer

Band 15

Daniel Alexander Meißner

Intersection-Based Road User Tracking Using a Classifying Multiple-Model PHD Filter

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.dnb.de> abrufbar.

Dissertation, Universität Ulm,
Fakultät für Ingenieurwissenschaften und Informatik, 2015

Impressum

Universität Ulm
Institut für Mess-, Regel- und Mikrotechnik
Prof. Dr.-Ing. Klaus Dietmayer
Albert-Einstein-Allee 41
89081 Ulm
<http://www.uni-ulm.de/mrm>

Eine Übersicht über alle Bände der Schriftenreihe finden Sie unter
<http://www.uni-ulm.de/mrmschriften>.

Diese Veröffentlichung ist im Internet auf dem Ulmer Volltextserver (<http://vts.uni-ulm.de>) verfügbar und dort unter der Lizenz "B - Standard (ohne Print-On-Demand)" publiziert.
Details zur Lizenz sind unter <http://www.uni-ulm.de/index.php?id=50392> zu finden.

Institut für Mess-, Regel- und Mikrotechnik der Universität Ulm 2016

e-ISBN 978-3-941543-19-5

Intersection-Based Road User Tracking Using a Classifying Multiple-Model PHD Filter

DISSERTATION

zur Erlangung des akademischen Grades eines

DOKTOR-INGENIEURS

(Dr.-Ing.)

der Fakultät für Ingenieurwissenschaften
und Informatik der Universität Ulm

von

Daniel Alexander Meißner
aus Ansbach

Gutachter:	Prof. Dr.-Ing. Klaus Dietmayer Prof. Dr.-Ing. Hans-Joachim Wünsche
Amtierende Dekanin:	Prof. Dr. Tina Seufert

Ulm, 27.10.2015

Acknowledgements

This dissertation results from my time at the Institute of Measurement, Control, and Microtechnology at the Ulm University. My special thanks goes to Professor Dr.-Ing. Klaus Dietmayer for his exceptional support throughout the last years, the conceded academic freedom, and the excellent working environment.

My gratitude also goes to Professor Dr.-Ing. Hans-Joachim Wünsche for being the second examiner of my dissertation.

The great working atmosphere at the “MRM” , fruitful discussions with colleagues, and the frequent social events tremendously influenced my work. Especially, the “Kaffeecke” and the Friday morning “Nusszopf” offered the opportunity to escape from apparently unsolvable academic or coding issues and enriched the daily work. In particular, I should mention Stephan Reuter, Elias Strigel, Benjamin Wilking, Florian Seeliger, Dominik Nuß, Hendrik Deusch, and Jürgen Wiest for crucial discussions and advises. Additionally, my acknowledge goes to Franz Degenhard, Oliver Betz, Martin Nieß, and Thomas Löffler for constructing accurate measurement setups and Claudia Fricke for her assistance in administrative concerns.

This dissertation results from the joint project Ko-PER - which is part of the project initiative Ko-FAS - and has been funded by the German Bundesministerium für Wirtschaft und Technologie (Federal Ministry of Economics and Technology) under grant number 19S9022G. In particular, I would like to thank the Ko-PER project leader Reiner Wertheimer and all involved project partners for the exceptional collaboration.

Last but not least, I would like to thank my family and especially my wife Stefanie for her constant support, encouragement, appreciation, and love. You were a refreshing source of life outside the PhD.

Abstract

Due to an increasing number of traffic accident fatalities at intersections, the prevention of traffic accidents at intersection attracts more and more attention in public as well as in research. One example therefore, is the joint-project Ko-PER of the research initiative Ko-FAS which aims to increase the preventive traffic safety. This is realized by an preferably exhaustive recognition of the vehicles' environments based on locally gathered and communicated perception data. Therefore, especially at urban intersections, an intersection-based perception system which communicates the current traffic situation to vehicles in the intersection's surrounding is essential. Thus, a busy public intersection in Aschaffenburg, Germany was equipped with an intersection perception system, which contains multiple laserscanners and video cameras as well as data processing units. Beside the planning of the perception system, the core of this thesis is the laserscanner-based detection and classification of road users and the tracking of them by fusing detections of the laserscanners and video cameras.

Since the laserscanners at the intersection are mounted at posts of lamps, traffic-lights as well as surrounding buildings, segmentation algorithms are introduced which utilize the knowledge about the consistent environment to segment the laserscanner measurements in background and foreground. Subsequently, the object detection in the foreground data relies on position as well as motion information. The 3D measurement points are clustered with a real-time capable implementation of the density based spatial clustering for applications with noise (DBSCAN) algorithm. Nevertheless, particularly large objects are sometimes not entirely detected due to occlusions, poor reflection characteristics and measurement uncertainties. This may lead to the separation of one object in multiple clusters. To avoid the over-clustering, a combination of 2D tracking and 3D clustering methods is proposed. Beside the benefit of the considerably more robust object detection, additionally the determination of the objects' orientations is improved. Finally, the detected objects are classified by means of features of their laserscanner point clusters.

The focus of this thesis lays on the tracking of road users at traffic intersections. Thus, a tracking approach is required which is able to handle multiple objects detected by multiple classifying sensors with different field of views. Due to the top-down modeling of the multi-object tracking problem and the availability of efficient

implementations, a Gaussian mixture probability hypothesis density (GM-PHD) filter is used. Based on this the classifying multiple-model probability hypothesis density (CMMPHD) filter is developed which facilitates to track road users of different type with its appropriate motion model and classify them. The classification is able to distinguish pedestrians, bikes, cars, and trucks. Therefore, features of the measurements and of the tracks are used. To model the object class of each track a basic belief assignment (BBA) of the Dempster-Shafer theory of evidence (DST) is exploited. Based on the estimated objects' classes the objects' motion model transition probabilities of the multiple model filter are adapted. To reliably recognise the road users in their varying appearances, laserscanners and video cameras are used. The fusion of the sensor measurements with the multi-sensor CMMPHD filter is realized by the introduction of the sensor individual modeling of the recognition abilities and the detection probability. In the end, the developed methods to recognize and track road users at traffic intersections are evaluated with real-world sensor and reference data.

Kurzfassung

Aufgrund der steigenden Anzahl von Verkehrstoten an Straßenkreuzungen rückt die Vermeidung von Kreuzungsunfällen immer mehr in den Fokus der Öffentlichkeit und der Forschung. Ein Beispiel hierfür ist das Teilprojekt Ko-PER der Forschungsinitiative Ko-FAS, dessen Ziel es ist die präventive Sicherheit im Straßenverkehr zu erhöhen. Dies wird durch eine möglichst umfassende Erkennung des Fahrzeugumfeldes basierend auf lokal erzeugten und kommunizierten Perzeptionsdaten realisiert. Speziell an innerstädtischen Kreuzungen ist hierzu ein kreuzungslokales Perzeptionssystem, das die aktuelle Verkehrssituation an die Fahrzeuge im Kreuzungsbereich kommuniziert unerlässlich. Dazu wurde eine stark befahrene öffentliche Kreuzung in Aschaffenburg mit einem Kreuzungsperzeptionssystem, das mehrere Laserscanner und Videokameras sowie Datenverarbeitungseinheiten umfasst, ausgestattet. Neben der Planung des Kreuzungsperzeptionssystems liegt der Schwerpunkt dieser Arbeit auf der laserscannerbasierten Detektion und Klassifikation von Verkehrsteilnehmern sowie dem Verfolgen dieser durch die Fusion der Laserscanner- und Videokameradetektionen.

Die Laserscanner sind an der Kreuzung an Masten von Laternen und Ampeln sowie angrenzenden Gebäuden befestigt. Daher werden Algorithmen vorgestellt, die das Wissen über die gleichbleibende Umgebung zur Segmentierung der Laserscannermessungen in Hinter- und Vordergrund nutzen. Die anschließende Objektdetektion in den Vordergrunddaten stützt sich sowohl auf Positions- als auch Bewegungsinformation. Die 3D Messpunkte werden mit einer echtzeitfähigen Implementierung des *density based spatial clustering for applications with noise* (DBSCAN) Algorithmus zu Clustern gruppiert. Jedoch werden speziell große Objekte aufgrund von Verdeckungen, schlechten Reflektionseigenschaften und Messunsicherheiten zum Teil nicht komplett erfasst. Dies kann zum Aufteilen eines Objekts in mehrere Cluster führen. Um das Überclustern zu vermeiden, wird eine Kombination von 2D und 3D Clustertechniken vorgestellt. Neben dem Vorteil einer deutlich robusteren Objekterkennung wird auch eine Verbesserung in der Orientierungsbestimmung erreicht. Zuletzt werden die detektierten Objekte anhand der Merkmale ihres Punkteclusters klassifiziert.

Der Schwerpunkt dieser Arbeit liegt auf dem Verfolgen von Verkehrsteilnehmern an Straßenkreuzungen. Dies erfordert ein Verfolgungsverfahren, das mit mehreren Objekten, die von mehreren klassifizierenden Sensoren mit unterschiedlichen Er-

fassungsbereichen detektiert werden, umgehen kann. Aufgrund der ganzheitlichen Modellierung des Multi-Objekt-Verfolgung Problems und der Verfügbarkeit von effizienten Implementierungen wird ein Gauß-Mixturen *probability hypothesis density* (PHD) Filter verwendet. Basierend darauf wurde der *classifying multiple model PHD* (CMMPHD) Filter entwickelt, mit dem es möglich ist Verkehrsteilnehmer unterschiedlicher Klassen mit zutreffendem Bewegungsmodell zu verfolgen und zu klassifizieren. Die Objektklassifikation kann Fußgänger, Zweiräder, PKWs und LKWs unterscheiden. Hierzu werden Merkmale der Messungen sowie der Tracks genutzt, die in einem *basic belief assignment* (BBA) der Dempster-Shafer Evidenztheorie für jeden Track fusioniert werden und schließlich die Objektklasse der einzelnen Tracks modellieren. Neben der Schätzung der Objektklasse dienen die BBAs der Tracks zur Bestimmung der Transitionswahrscheinlichkeiten zwischen den Bewegungsmodellen des Multi-Modell Filters. Die Fusion von Laserscanner- und Videokameramessungen ermöglicht die zuverlässige Erkennung von Verkehrsteilnehmer in ihren unterschiedlichen Erscheinungsformen. Durch die sensorindividuelle Modellierung des Erkennungsvermögens und des Erkennungsbereichs ermöglicht der *multi-sensor* CMMPHD Filter die Fusion von heterogenen Sensormessungen. Abschließend werden die entwickelten Methoden zum Erkennen und Verfolgen von Verkehrsteilnehmern an Straßenkreuzungen mit realen Sensordaten und Referenzdaten evaluiert.

Contents

1	Introduction	1
2	Infrastructure-Based Environment Perception for Driver Assistance Systems	5
2.1	The Joint Project Ko-PER	5
2.2	The Ko-PER Cooperative Multi-Sensor Intersection Perception System	6
2.2.1	Sensor Setup	6
2.2.2	Temporal and Spatial Calibration of the Sensors	11
2.3	Related Work	12
3	Fundamentals in Object Tracking	15
3.1	The Single-Object Bayes Filter	16
3.2	The Kalman Filter	18
3.2.1	The Constant Gain Kalman Filter	20
3.2.2	The Extended Kalman Filter	20
3.2.3	The Unscented Kalman Filter	21
3.2.4	The Interacting Multiple-Model Kalman Filter	23
3.3	Multi-Object Tracking	25
3.3.1	Gating	26
3.3.2	Data Association Approaches	27
3.4	Motion Models	29
3.4.1	Combining Different Motion Models	31
3.4.2	Extended Object Tracking using Reference Points	32
4	Fundamentals in Probability Hypothesis Density Filtering	33
4.1	Random Finite Sets	34
4.1.1	Multi-Object Probability Distributions	35
4.2	The Multi-Object Bayes Filter	38
4.2.1	Multi-Object Likelihood Functions	39
4.2.2	Multi-Object Markov Densities	42
4.3	The Probability Hypothesis Density Filter	43
4.3.1	The Probability Hypothesis Density	44
4.3.2	The PHD Filter	47
4.3.3	The Gaussian-Mixture PHD Filter	48

4.3.4	GM-PHD Filter Pruning and Merging	54
4.3.5	State Dependent Detection and Survival Probability	55
4.3.6	Birth models	56
4.4	The Gaussian Mixture Multiple-Model PHD Filter	57
4.4.1	The Jump Markov System	57
4.4.2	The GM-MMPHD Filter	58
4.4.3	The GM-MMPHD Filter Incorporating Different State Spaces	61
5	Multi-Sensor Classifying MMPHD Filter	63
5.1	Introduction to the Dempster-Shafer Theory of Evidence	65
5.2	The Classifying MMPHD Filter	69
5.2.1	Using BBAs for Track Classification	72
5.2.2	Using BBAs to Adapt the Transition Matrix	74
5.3	The Multi-Sensor CMMPHD Filter	75
5.3.1	Multi-Sensor PHD filters	76
5.3.2	Fusing Sensors with Different FOVs	78
5.3.3	Fusing Sensors with Different Recognition Abilities	79
6	Laserscanner-Based Detection and Classification of Road Users at Intersections	81
6.1	Information Maps	81
6.2	Object Detection	82
6.2.1	Background Subtraction	84
6.2.2	3D Clustering	88
6.2.3	2D Pathway Tracking	90
6.2.4	Correction of Over-Clustered Objects and Object Generation	93
6.3	Road User Classification	95
7	Application of the CMMPHD Filter to Road User Tracking at Intersections	101
7.1	Sensor Models	101
7.1.1	Laserscanner	102
7.1.2	Low Resolution Camera	104
7.1.3	High Definition Camera	106
7.2	Parameterization of the CMMPHD Filter	107
7.2.1	Survival Probability	107
7.2.2	Road User Classification	108
7.2.3	Motion Models	109
8	Evaluation	113
8.1	Performance Measures	114
8.1.1	Absolute Estimation Errors	114
8.1.2	Optimal Subpattern Assignment Metric	114

8.1.3	Filter Consistency	116
8.2	Improvement of Object Detection Using Pathway Tracks	117
8.3	Road User Tracking	119
8.3.1	Comparison of a Single-Model and a Multiple-Model Filter	119
8.3.2	Tracking Results for Representative Maneuvers	121
8.3.3	Accuracy of Cars' State Estimation	126
8.3.4	Laserscanner-Based Multi-Object Tracking Performance	130
8.3.5	Benefit of Multi-Sensor Fusion	133
8.4	Road User Classification Performance	135
8.5	Conclusion	137
9	Conclusion and Future Work	139
9.1	Conclusion	139
9.2	Future Work	141
A	Sensor Planning for Intersection Sensors	143
B	Sensor Calibration	147
	Acronyms	151
	List of Symbols	155
	Bibliography	159
	Publications	173
	Supervised Theses	175

Chapter 1

Introduction

The number of registered vehicles and consequently the traffic volume in European countries raises year by year, e.g. [Sta14]. However, in contrast to the obvious conclusion of an increasing number of traffic accident fatalities, the number has nearly halved since 2000 in the 24 European countries considered by [BBP⁺12]. This continuous reduction may be ascribed to the spreading of systems like the anti-lock breaking system (ABS) and the electronic stability control (ESC) system which improve driving stability. Further, the introduction of advanced driver assistance systems (ADAS) perceiving the vehicles' environment with sensors contributes to increasing traffic safety. In-market systems like adaptive cruise control (ACC), parking assists, lane departure warning, and pre-safe systems are based on on-board ultrasonic, radar, video, and lidar sensors. As these sensors are on-board, they are able to assist the driver in observing the vehicle's surrounding. Consequently, it is not surprising that in dense traffic situations where the field of views (FOVs) of the driver and of on-board perception systems are highly restricted, currently available ADAS are not able to achieve a significant reduction of fatalities. Thus, the portion of fatalities that occur at urban intersections, where various road users converge on a confined space, has been slightly over 20 % throughout the last decade [BPT⁺12; PTS⁺12]. Hence, research to realize intersection collision avoidance systems has attracted attention in recent years.

In this thesis, innovative and general-purpose algorithms to recognize and track road users at urban traffic intersections are developed. Applying these algorithms to the cooperative multi-sensor intersection perception system which was developed and installed during the joint project cooperative perception (Ko-PER) of the research initiative cooperative driver assistance systems (Ko-FAS), a preventive traffic safety system is realized. The aim of the intersection perception system is to detect all road users in the central widespread and the approaches of the intersection, track them continuously on their way through the intersection, and classify their type. Therefore, the intersection was equipped with several spatially

distributed laserscanners and video cameras which provide temporal and spatial consistent information about the current situation at the intersection. The basis for an accurate dynamic representation of the traffic situation at the intersection are the sensor dependent object detection algorithms. For the laserscanners new object detection algorithms are proposed which exploit the static sensor setup to segment measurements originated from static obstacles and dynamic objects. Subsequently, the highly accurate 3D information about the dynamic objects in combination with information about the temporal evolution of those objects is used to generate reliable and precise object hypotheses. These object hypotheses are finally validated by a point cloud feature based object classification. For the cameras well-known detection by classification methods which facilitate the recognition of vehicles or pedestrians in the camera images are used. Thus, the class independent object recognition of laserscanners and the class dependent vehicle and pedestrian recognition of cameras complement each other which is superior to systems using only one sensor type.

The complementary sensor characteristics require a multi-sensor tracking algorithm which is capable to model the recognition abilities of the sensors. For this purpose the concept of Dempster-Shafer theory of evidence (DST) based multi-sensor fusion [MMD10] is enhanced. Furthermore, at an intersection in general multiple relevant objects of different type and with different type dependent motion characteristics are present, e.g. pedestrians and cars. To realize a persistent and reliable classification and tracking of multiple objects with different motion characteristics gathered by complementary sensors, the multi-sensor classifying multiple-model probability hypothesis density (CMMPHD) filter is proposed. The core of the CMMPHD filter is the probability hypothesis density (PHD) of a random finite set (RFS) which facilitates an approximate and computationally tractable implementation of the multi-object Bayes filter introduced by Mahler [GMN97; Mah07a]. This approximation, known as the PHD filter [Mah03], is the multi-object counterpart of the constant gain Kalman filter in single-object tracking. Since the PHD filter propagates a representation of the entire multi-object state over time and features a theoretically funded consideration of missed detections, clutter measurements, track termination, and track birth, it outperforms classical multi-instance Kalman filters in tracking multiple maneuvering objects. Additionally, the PHD filter does not require a computationally costly measurement-to-track association, since both the multi-object state and the multi-object measurement are modeled as RFSs.

Starting from the multiple-model probability hypothesis density (MMPHD) filter using a Gaussian mixture (GM) implementation [Mah12; PVTM09], several extensions are required to solve the described road user tracking and classification problem which finally leads to the CMMPHD filter. At first, each mixture component (MC) of the GM representing the PHD is enhanced by a basic belief assignment (BBA) according to Dempster-Shafer theory of evidence (DST) which represents the object's class. Adding class information as an attribute to the MCs, enables

the usage of measurement-based classification results by maintaining the freedom of association of the PHD filter and its low computational complexity. Additionally, the dynamics like the velocities of the MCs can be exploited as features for object classification. Furthermore, the BBA representation of the object's class facilitates the fusion of classifying sensors differing in their FOV, and their recognition abilities, which means that not every sensor has to be able to recognize every object type. Even contradicting information about the existence or the class of an object is handled in a natural way by the Dempster-Shafer-based modeling of the objects' classes and the sensors' recognition abilities. Obviously, the relevant objects at an intersection have different and type dependent motion characteristics. Therefore, the class-BBAs are used to deduce MC individual motion model probabilities to predict the MCs sufficiently within the MMPHD filter. All these features are summarized in the proposed multi-sensor CMMPHD filter which is developed to jointly track and classify objects of different type and with different motion characteristics which are observed by sensors differing in their FOVs and recognition abilities.

This thesis is organized as follows: First, the cooperative multi-sensor intersection perception system developed and installed at a public urban intersection during the joint project Ko-PER is described in Chapter 2.2. Further, some other prominent research projects focusing on intersection safety are outlined in Section 2.3. Subsequently, fundamentals in single and multi-object tracking are given in Chapters 3 and 4. While in Chapter 3 traditional linear and non-linear variants of the Kalman filter, multi-instance approaches for multi-object tracking, and motion models are outlined, Chapter 4 introduces the RFS theory. Therein, PHD filters involving multiple motion models and multiple sensors are introduced. After the fundamentals, the main contribution of this thesis, the CMMPHD filter, is presented in Chapter 5. Starting with a brief review of the DST, the CMMPHD filter is introduced in general in Section 5.2 and in its multi-sensor version in Section 5.3. After the theoretical consideration of the employed tracking algorithms, Chapter 6 concerns the laserscanner-based detection and classification of road users in general as well as the application of the methods to the Ko-PER intersection. Within Chapter 7 the application of the CMMPHD filter to the Ko-PER intersection including the modeling of the sensors is presented. Performance measures for the evaluation of the single and multi-object tracking results as well as the class confusion matrix (CCM) to quantify the classification accuracy are introduced in Chapter 8. Subsequently, the achieved tracking and classification quality of the CMMPHD filter is evaluated with real-world data.

Chapter 2

Infrastructure-Based Environment Perception for Driver Assistance Systems

2.1 The Joint Project Ko-PER

The joint project cooperative perception (Ko-PER) [WK11] of the research initiative cooperative driver assistance systems (Ko-FAS) [KoF14] was launched in 2010 and aimed during its term of four years a virtually exhaustive dynamic representation of the local traffic situation to promote preventive traffic safety. Therefore, the superposition of perception results of spatially distributed sensor networks to complement locally missing pieces of information about the current traffic situation was investigated. Involved sensor networks are on the one hand vehicles equipped with video cameras, laserscanners, or radar sensors to gather environment information and on the other hand hazardous intersections perceiving their environment with networks of spatially distributed video cameras and laserscanners.

Each perception system gathers information about the road users in its individual FOV and provides this information via vehicle-to-vehicle (V2V) or infrastructure-to-vehicle (I2V) communication. In the receiver vehicle, the information about relevant road users, which may comprises the position, orientation, extent, dynamics, and type of the objects, is fused with its own on-board perception results. Consequently, the environment model of the vehicle enlarges to lanes, ranges, or even roads of the intersection which are not visible for the driver or on-board sensors at the current vehicle location. Further, in platooning situations on roads or in dense traffic at intersections, occlusions might be resolved by the exchange of perception results of systems with different FOVs and viewpoints. In addition to the environment perception, a precise self-localization of the vehicles and a common time shared

by all communication partners is required to utilize the communicated perception results. The reason for this is that the vehicle needs to be able to associate the received objects with their own findings with respect to space and time. Finally, the exhaustive dynamic environment representation is used to run algorithms to interpret the current traffic situation and derive advisory warnings for the driver [SWP⁺14].

To realize the advisory warnings at urban intersections where the FOV of drivers and on-board sensors is highly restricted due to dense traffic and urban development, the intersection perception system has an important part in Ko-PER. In this thesis the Ko-PER intersection perception system with focus on the laserscanner-based object recognition, road user tracking and classification, as well as multi-sensor fusion is addressed.

2.2 The Ko-PER Cooperative Multi-Sensor Intersection Perception System

In the framework of the joint project cooperative perception (Ko-PER) a public traffic intersection located in the city center of Aschaffenburg, Germany (latitude: 49.970447° N, longitude: 9.161500° E) was equipped with sensors. This installation facilitates the development and testing of algorithms for road user recognition and tracking under realistic conditions.

The main road of the intersection features two straight ahead lanes and a separate left turn lane, as evident from Figures 2.1 and 2.2. It is crossed by a smaller road with one straight ahead lane and a separate left turn lane on one side. Since the intersection is crossed by over 22,000 vehicles per day along the main road, located along the way of several bus routes, and adjacent to the University of Applied Sciences, it is one of the busiest intersections in Aschaffenburg with a huge variety of road users. Consequently, this intersection is suitable to develop an intersection perception system which might be a key component to realize driver assistance systems for urban intersections.

2.2.1 Sensor Setup

The intersection is equipped with a perception system to detect, classify, and track road users. In this section only the components of the intersection perception system used in this thesis are highlighted. For a detailed description of the entire system

Figure 2.1: Realistic 3D model of the Ko-PER intersection. The aligned range measurements of 14 laserscanners are depicted by orange points [MRSD14].

Figure 2.2: Field of view of the low resolution (LR) cameras. The images 1-3 are gathered by the cameras whose field of view (FOV) is depicted in Figures A.2a-A.2c respectively [MRSD14].

Figure 2.3: Field of view of the high resolution (HR) cameras. The images 1-2 are gathered by the cameras whose FOV is depicted in Figure A.2d [GSM⁺12].

Figure 2.4: Pictures of mounted intersection sensors [GSM⁺12].

Table 2.1: Complementary Sensor Characteristics

	laserscanner	camera
pros	range information	texture information
	independent of illumination	huge horizontal and vertical FOV
	huge horizontal FOV	high resolution
cons	no texture information	no range information
	minor vertical FOV	dependent on illumination

refer to [GSM⁺12]. Information about the environment is gathered by a network of 14 SICK LD-MRS 8-layer research laserscanners, three monochrome low resolution (LR) cameras, and two monochrome high resolution (HR) cameras. The reason for the choice of the sensors results from the complementarity of the sensors' characteristics summarized in Table 2.1. In Figure 2.4 three pictures of the sensors mounted at the intersection are provided to give an impression of the setup. In the following the sensors, the spatial sensor planning, and the temporal and spatial sensor calibration are introduced.

Laserscanner

The laserscanners are active sensors basing on the light detection and ranging (LIDAR) technology. The sensor emits several laser beams, whose azimuthal angle is controlled by a rotating mirror and the elevation angle is controlled by the tilt of this mirror. Receiving reflected laser beams with a photo diode, the radial distance between the laserscanner's origin and the objects is determined by means of time-of-flight calculations. The major characteristics of the laserscanner and the used

Table 2.2: Laserscanner Characteristics and Used Settings

characteristic	value
measurement range	0.5m – 200m
horizontal FOV	85°
vertical FOV	3.2° – 6.4°
measurement error (1σ)	0.1m
layers	8
used angular resolution	0.25°
used scan frequency	12.5Hz

Figure 2.5: Illustration of the FOV and characteristics of the LD-MRS laserscanner.

settings are summarized in Table 2.2.

The laserscanner features eight scan layers which enable a vertical FOV from 3.2° to 6.4° dependent on the azimuthal angle. Figure 2.5a illustrates the arrangement of the layers for the azimuthal angle with the largest vertical FOV. Here, the elevation angles of neighboring layers differ by 0.8°. In contrast to the limited vertical FOV, the horizontal FOV is with 85° very huge (see Figure 2.5b). This range is sampled with a step size of 0.5° by each layer. Since two neighboring layers have an offset of 0.25°, the over-all horizontal angular resolution is 0.25°.

Providing an external synchronization signal, the laserscanner scans its environment with 12.5 Hz. Thus, the sensor gathers the reflections of the laser beams for each azimuthal angle and each layer every 80 ms and calculates the distance, the echo pulse width which is the duration of the puls on the time line, as well as the 3D position of the reflection point in Cartesian sensor coordinates. Since the laserscanner

is an active sensor, the detection performance is not influenced by illumination. Even during heavy rain or light snowfall, the degradation was acceptable for the perception system. Additionally, due to the vertical and horizontal FOV, the sensor enables to build a 3D point cloud representation of its surrounding. In Figure 2.1 the 3D point cloud gathered by the laserscanner network at the intersection is depicted.

To plan and realize a laserscanner system which facilitates a complete observation of the intersection as visualized in Figure 2.1, a 3D sensor planning framework was developed [MD10]. Based on an accurate 3D model of the intersection composed of buildings, intersection infrastructure, lane markings, and vegetation, feasible mounting positions and sensor poses were planned. Furthermore, the sensor planning framework supports the simulation of the laserscanners' FOV as well as their measurements considering the environment conditions. Thus, optimal sensor positions with respect to sensor coverage, spatial resolution, detection range, and occlusion avoidance were determined. Occlusions by stationary objects were eliminated and robustness against occlusions by moving objects like trucks was achieved by using high mounting positions between 4 m and 12 m above road level and redundant FOVs. According to the sensor planning shortly addressed in Appendix A, the laserscanner network was installed and the sensor coverage illustrated by Figure 2.1 was achieved. The comparison of the simulated coverage and the real coverage emphasizes the accuracy of the environment and sensor modeling in the simulation framework [MD10]. As visualized in Figure 2.1, the laserscanner system facilitates the observation of the intersection widespread, two sidewalks, and three approaches up to high ranges.

Video Camera

The focus of this thesis is not on video-based object detection but rather on fusing object detections of camera and laserscanner systems. Consequently, the used cameras are only described briefly. For detailed information refer to [GSM⁺12] and [SMS⁺14]. The type of the three monochrome low resolution (LR) cameras is TXG04 from Baumer which has a resolution of 656×494 pixels. They are synchronized in time and operate at 25 Hz in phase with the laserscanners. Also the LR cameras were simulated with the sensor planning framework which yields the FOVs in Figures A.2a-A.2c. Each camera observes one approach to the intersection. Example images of the cameras are given by Figure 2.2.

Finally, the two high resolution (HR) cameras cover a neuralgic area of the intersection (see Figure 2.3). As shown in the simulation results in Figure A.2d, their optical axis form an angle of approximately 90° to cope with occlusions. Furthermore, those AVT Prosilica GX1910 cameras with a resolution of 1920×1080 pixels provide a wide

angle stereo to detect the 3D position and the intention of pedestrians [KGB⁺13]. The cameras operate at 50 Hz and are also in phase with the laserscanners.

Generally, it is beneficial that the FOV and the resolution of the cameras are design parameters which can be influenced by the choice of the lens and the photosensor. However, the quality of the gathered texture information heavily depends on the lighting conditions at the intersections. Due to the installed hardware, both camera systems are not able to detect road users during night-time and show a degraded performance if they are dazzled.

2.2.2 Temporal and Spatial Calibration of the Sensors

The temporal calibration of the intersection system has to meet two requirements: First, to facilitate a consistent fusion of the sensors, highly accurate timestamps corresponding to the data acquisition time have to be associated with the measurements. The timestamps are required to be derived from a time base which is common for the entire intersection system. Second, as introduced in Section 2.1, the result of the intersection perception is communicated to vehicles enlarging their knowledge about the current traffic situation at the intersection. This extends the timestamps requirements from an intersection wide validity to a global validity. Therefore, the trigger pulses for the intersection sensors are derived from the global positioning system (GPS) time and dedicated universal time, coordinated (UTC) timestamps are provided for the sensor measurements. The accuracy of the temporal calibration was evaluated in [GSM⁺12].

The spatial calibration of the sensors is required to utilize the spatially distributed sensors for the realization of a highly accurate road user detection and tracking system. Therefore, an intersection coordinate system C_I is defined which has east-north-up (ENU) orientation and its (x, y) -plane is parallel to the road surface. The spatial calibration of the sensors provides the transformation of the sensor data gathered in the sensor individual coordinate system C_S to the common C_I system. Independent of the sensor type, a point in homogeneous sensor coordinates $p^{C_S} = [p_x, p_y, p_z, 1]^T$ can be transformed to C_I with the extrinsic homogeneous transformation matrix $\underline{T}_{C_S}^{C_I}$:

$$p^{C_I} = \underline{T}_{C_S}^{C_I} \cdot p^{C_S} = (\underline{R} \ t)_{C_S}^{C_I} \cdot p^{C_S}. \quad (2.1)$$

The rotation matrix $\underline{R}$ and the translation vector t specify the pose of the sensors. For the cameras, even the lenses have to be considered. Here the intrinsic parameters of the pinhole camera model in combination with the extrinsic parameters facilitate

the transformation between intersection coordinates and 2D pixel coordinates:

$$\begin{pmatrix} x_p \\ y_p \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \end{pmatrix} \cdot \underline{T}_{C_I}^{C_S} \cdot \begin{pmatrix} p^{C_I} \\ 1 \end{pmatrix}. \quad (2.2)$$

Here, the focal length and the principal point are denoted by f_x , f_y , c_x , and c_y in the camera intrinsic matrix. For the cross calibration of sensors with different measurement principles, a calibration object was designed which is suitable for both laserscanner and camera calibration. The approaches to determine the intrinsic and extrinsic calibration parameters are summarized in Appendix B.

2.3 Related Work

In the United States of America the cooperative intersection collision avoidance systems-stop sign assist (CICAS-SSA) project [ACG⁺06; GMC⁺08] equipped rural intersections with radar sensors and laserscanners to reduce crashes at thru-stop controlled intersections. The sensors were mounted at street level next to the main road and provide sensing information to calculate the vehicles' dynamic states. Based on the speeds of the vehicles on the main road, algorithms trigger dynamic signs to warn minor road drivers against unsafe traffic gaps. The CICAS-SSA system differs from the Ko-PER intersection system in several points: First, the perceived dynamic object information is not communicated to surrounding vehicles but processed directly at the intersection to warn the driver. Second, the CICAS-SSA system was only designed to observe vehicles. Consequently, the classification of objects is not required. Finally, the sensors are mounted on road level which makes the system vulnerable to occlusions during dense traffic.

In Europe the SafeSpot subproject INFRASENSE [Ehr10] utilized laserscanners, cameras, and RFID-systems to recognize and track road users at urban intersections. While the Ko-PER approach fuses the object detections of multiple sensor types, in INFRASENSE the sensors are investigated separately and the laserscanners are still vulnerable to occluded views due to low mounting positions. This drawback was solved by the intersection perception system of the Inter Safe 2 project [RFW⁺11] which was solely based on spatially distributed laserscanners mounted at infrastructure installations like lamp posts high above the road level. However, the system abandons to use cameras to gather texture information and uses application specific local features like the foot-way regions to classify pedestrians [SR11].

Finally, in [ZCZ⁺09] a laserscanner and camera-based intersection perception system to recognize and track road users at urban intersections was presented. Using a

network of laserscanners and cameras sensing from different viewpoints to reduce occlusions and to fuse the detected objects of both complementary sensor types, this intersection system has most in common with the Ko-PER system. Beside the similarities, the laserscanners in [ZCZ⁺09] have no bird's-eye perspective, the perceived dynamic representation of the intersection scene is not communicated, and the object detection and tracking algorithms developed in this thesis are of general purpose.

Chapter 3

Fundamentals in Object Tracking

The term object tracking means the continuous estimation of time dependent object properties based on uncertain measurements. Generally, the time dependent object properties are denoted as the objects' states and their change in time is predicted using an uncertain motion model. Since in many applications not all state components are measurable, the transformation between the measurement space and the state space is done using a characteristic sensor model.

An illustration of the object tracking procedure for the single-object case is given by the upper part of Figure 3.1. Here, it is assumed that each object originates one measurement in the measurement space. In the state space the object is represented by a track which is predicted using the object's motion model and corrected according to the received measurement. In the lower part of Figure 3.1, the propagation procedures of popular single-object tracking methods are shown which are introduced and discussed in this chapter.

In Section 3.1 the recursive Bayes filter which provides the foundation for all single-object tracking methods addressed in this thesis is introduced. Afterwards, with the Kalman filter [Kal60] a closed-form implementation of the recursive Bayes filter using Gaussian distributions is summarized. Additionally, non-linear and multiple-model versions of the Kalman filter are outlined in Section 3.2. While Section 3.3 presents common multi-object tracking algorithms, the chapter is closed with Section 3.4 introducing motion models for object tracking.

single-object Bayes filter:

$$p_k(x_k|Z^k) \xrightarrow{\text{prediction}} p_{k+1|k}(x_{k+1}|Z^k) \xrightarrow{\text{update}} p_{k+1}(x_{k+1}|Z^{k+1})$$

Kalman filter:

$$(\hat{x}_k, \underline{P}_k) \xrightarrow{\text{prediction}} (\hat{x}_{k+1|k}, \underline{P}_{k+1|k}) \xrightarrow{\text{update}} (\hat{x}_{k+1}, \underline{P}_{k+1})$$

constant gain Kalman filter:

$$\hat{x}_k \xrightarrow{\text{prediction}} \hat{x}_{k+1|k} \xrightarrow{\text{update}} \hat{x}_{k+1}$$

Figure 3.1: Visualization and overview of common single-object tracking algorithms and their recursion steps. In the measurement space objects are represented by circles and measurements by stars. In the state space tracks are illustrated by crossed circles.

3.1 The Single-Object Bayes Filter

In the recursive Bayes filter the states of an object are represented as a state vector x and the recursion is composed of a prediction step

$$p_{k+1|k}(x_{k+1}|Z^k) = \int f_{k+1|k}(x_{k+1}|x_k) p_k(x_k|Z^k) dx_k \quad (3.1)$$

and innovation step

$$p_{k+1}(x_{k+1}|Z^{k+1}) = \frac{g(z_{k+1}|x_{k+1}) p_{k+1|k}(x_{k+1}|Z^k)}{\int g(z_{k+1}|x_{k+1}) p_{k+1|k}(x_{k+1}|Z^k) dx_{k+1}}. \quad (3.2)$$

The aim of the recursion illustrated by Figure 3.1 is to estimate the posterior probability density function (PDF) $p_{k+1}(x_{k+1}|Z^{k+1})$ of an object with state x_{k+1} at time $k+1$ given the set of measurements $Z^{k+1} = \{z_1, \dots, z_{k+1}\}$. Therefore, the estimated posterior PDF of the previous time step k is multiplied with a Markovian

transition density

$$f_{k+1|k}(x_{k+1}|x_k) \quad (3.3)$$

in the prediction step (3.1). The Markovian transition density represents the change of the object's state between data collection times k and $k + 1$ and is defined by the motion model of the object:

$$x_{k+1|k} = f(x_k, v_k). \quad (3.4)$$

Here, v_k represents the uncertainty of the motion model. One must consider that due to the Markov assumption the whole history $(1, \dots, k)$ is subsumed in the state x_k which results in (3.3) being just dependent on x_k . Thus, the prior PDF $p_{k+1|k}(x_{k+1}|Z^k)$ is predicted from time step k to $k + 1$ by integrating $f_{k+1|k}(x_{k+1}|x_k)p_k(x_k|Z^k)$ over x_k in (3.1).

In the innovation step (3.2) the posterior PDF of time step $k + 1$ is determined. Here, the result of the prediction step (3.1) is multiplied by the measurement likelihood $g(z_{k+1}|x_{k+1|k})$ which models the likelihood for receiving a measurement z_{k+1} given the predicted state $x_{k+1|k}$. The measurement likelihood is based on the measurement model

$$z_{k+1|k} = h_{k+1}(x_{k+1|k}, w_{k+1}), \quad (3.5)$$

which transforms the predicted state $x_{k+1|k}$ from the state space into the measurement space considering the measurement noise w_{k+1} . Consequently, the value of $g(z_{k+1}|x_{k+1|k})$ is high if z_{k+1} favors $x_{k+1|k}$ and small if otherwise. Since it is not guaranteed that $g(z_{k+1}|x_{k+1|k}) \cdot p_{k+1|k}(x_{k+1|k}|Z^k)$ integrates to one, it has to be divided by a normalizing factor to receive a valid posterior PDF from (3.2). The posterior PDF $p_{k+1}(x_{k+1}|Z^{k+1})$ now encapsulates the whole knowledge about the object's state. Finally, choosing the most probable state leads to a Bayes-optimal state estimate.

For readability reasons the subscript for predicted variables $k + 1|k$ is substituted by $+$ and the subscripts k and $k + 1$ are dropped in the following. Since the prior PDF (3.1) just depends on posterior PDFs and the right side of the innovation (3.2) contains only prior PDFs, the notation is still distinct.

3.2 The Kalman Filter

The Kalman filter is the by far most prevalent implementation of the recursive Bayes filter and was originally derived in [Kal60] using least-squares. In [HL64] identical equations have been derived using Gaussian distributions for the PDFs, Gaussian distributed white noise with zero mean, and a linear measurement and motion model in the recursive Bayes filter. The popularity of the Gaussian distribution

$$\mathcal{N}(x; \bar{x}, \underline{P}) = \frac{1}{\sqrt{\det(2\pi\underline{P})}} \exp\left(-\frac{1}{2}(x - \bar{x})^T \underline{P}^{-1}(x - \bar{x})\right) \quad (3.6)$$

in statistics is due to the fact that it is completely characterized by its first two moments, the mean value $\bar{x}$ and the covariance matrix $\underline{P}$. Thus, instead of propagating the entire PDFs over time, it is sufficient to confine it to the first two statistical moments. This simplification becomes apparent comparing the single-object Bayes filter and the Kalman filter sections in Figure 3.1. If the requirements of the derivation are fulfilled the Kalman filter is a closed-form Bayes-optimal single-object state estimator.

Applying linear motion and measurement models, equations (3.4) and (3.5) simplify to

$$x_+ = \underline{F}x + v \quad (3.7)$$

$$z_+ = \underline{H}x_+ + w, \quad (3.8)$$

with the process matrix $\underline{F}$ and the measurement matrix $\underline{H}$. All variables are assumed to be Gaussian distributed random variables, further the process noise v and the measurement noise w have to be uncorrelated, white and zero-mean. Meeting these requirements, leads to the Kalman filter's equations for the prediction of the state and its covariance:

$$\hat{x}_+ = \underline{F}\hat{x} \quad (3.9)$$

$$\underline{P}_+ = \underline{F}\underline{P}\underline{F}^T + \underline{Q}. \quad (3.10)$$

The matrix $\underline{Q}$ denotes the covariance of the process noise.

Having predicted the state to the time of the next measurement, an expectation for this measurement based on the predicted state is required in the innovation step of the Kalman filter. Analogous to the state prediction, the measurement prediction is

given by:

$$\hat{z}_+ = \underline{H}\hat{x}_+ \quad (3.11)$$

$$\underline{S} = \underline{H}\underline{P}_+\underline{H}^T + \underline{R}. \quad (3.12)$$

The sum of the predicted state covariance $\underline{P}_+$ in the measurement space with the covariance of the measurement noise $\underline{R}$ results in the innovation covariance $\underline{S}$ of the measurement residual

$$\gamma = z - \hat{z}_+. \quad (3.13)$$

The predicted state covariance $\underline{P}_+$ expresses the uncertainty of the state prediction and the innovation covariance $\underline{S}$ expresses the uncertainty of the innovation. Thus, the dependency of the posterior state $\hat{x}$ on the uncertainties of the motion and measurement model are considered using the Kalman gain

$$\underline{K} = \underline{P}_+\underline{H}^T\underline{S}^{-1}. \quad (3.14)$$

Consequently, the sum of the prior state and the measurement residuum transformed and weighted by the Kalman gain leads to the measurement updated estimate of the object's state:

$$\hat{x} = \hat{x}_+ + \underline{K}\gamma. \quad (3.15)$$

Accordingly, the covariance of the state is innovated:

$$\underline{P} = \underline{P}_+ - \underline{K}\underline{S}\underline{K}^T \quad (3.16)$$

$$= (\underline{I} - \underline{K}\underline{H})\underline{P}_+(\underline{I} - \underline{K}\underline{H})^T + \underline{K}\underline{R}\underline{K}^T. \quad (3.17)$$

Exploiting a limit value consideration, the role of the Kalman gain $\underline{K}$ in (3.15) and (3.16) becomes apparent. Assuming the measurement error covariance $\underline{R}$ approaches zero, the Kalman gain raises the influence of the current measurement on the posterior state and covariance. On the contrary, if the prior state error covariance $\underline{P}_+$ approaches zero the low $\underline{K}$ suppresses the correction of the prior state and covariance in (3.15) and (3.16). The two Equations (3.16) and (3.17) for the covariance innovation are theoretically identical but differ in terms of numerical stability. The quadratic form of both terms in (3.17) increases the numerical stability which makes it the better choice for implementation. Nevertheless, for notational convenience the short form (3.16) is used in the following.

Below, common variants of the Kalman filter are covered. At first, the efficient constant gain Kalman filter for time invariant systems is introduced. Subsequently, since many applications require non-linear models, the extended Kalman filter (EKF)

and unscented Kalman filter (UKF), which are filters for non-linear systems, are summarized in Sections 3.2.2 and 3.2.3. The Kalman filter introduction is concluded with the interacting multiple-model (IMM) Kalman filter to regard the motion states of maneuvering objects.

3.2.1 The Constant Gain Kalman Filter

Assuming time invariance for the process and measurement model as well as for its covariance matrices in (3.10) and (3.12), leads to a constant Kalman gain $\underline{K}$ (3.14) and covariance matrix $\underline{P}$ which can be calculated in advance. Thus, the update of $\underline{P}$ (3.16), which is the computationally most expensive part of the Kalman filter due to the matrix inversion, can be skipped. Consequently, the constant gain Kalman filter is an approximation of the Bayes filter which propagates only the first moment $\hat{x}$ of the posterior PDF over time (see Figure 3.1). Popular representatives are the α - β filter for second-order and the α - β - γ filter for third order kinematic models [HM04a; Kal84].

In contrast to the advantage of the constant gain in terms of computational costs, it is not optimal in the initialization phase of the filter. Compared to the Kalman filter, the constant gain Kalman filter features no stabilization phase to compensate initial mismatches of the state and covariance estimates. Furthermore, non-stationary noise processes lead to deviations in the estimation result.

3.2.2 The Extended Kalman Filter

The extended Kalman filter (EKF) [BL93; SSM62] is based on the first-order Taylor series approximation of the non-linear process and measurement models around the current estimate. Dependent on the required accuracy, even EKFs of higher order, which means that the Taylor series is not truncated after the linear term, can be applied. Due to the rising computational complexity, the first-order EKF is used in this thesis. Based on the already introduced general motion (3.4) and measurement models (3.5), which contain the eventually non-linear functions f and h , the process matrix $\underline{F}$ in the standard Kalman filter equations is replaced with the Jacobian

$$\underline{F}^J = \left. \frac{\partial f}{\partial x} \right|_{x=\hat{x}} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix} \bigg|_{x=\hat{x}} \quad (3.18)$$

and the measurement matrix $\underline{H}$ is substituted by the Jacobian

$$\underline{H}^J = \left. \frac{\partial h}{\partial x} \right|_{x=\hat{x}_+} = \left(\begin{array}{ccc} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_m}{\partial x_1} & \cdots & \frac{\partial h_m}{\partial x_n} \end{array} \right) \bigg|_{x=\hat{x}_+}. \quad (3.19)$$

The Jacobian matrices are evaluated for the current state estimate $\hat{x}$ and the predicted state $\hat{x}_+$ respectively. Replacing the process matrix $\underline{F}$ and the measurement matrix $\underline{H}$ with the Jacobian matrices $\underline{F}^J$ and $\underline{H}^J$, the Kalman filter equations facilitate the estimation of an object's state in case of a slightly non-linear system. In case of stronger non-linearities the linearization around the current estimate exploiting only the first derivative of f and h is not appropriate, so higher-order Kalman filters or the unscented Kalman filter (UKF) have to be used.

3.2.3 The Unscented Kalman Filter

In contrast to the EKF, the equations of the system are not linearized in the UKF [JU96; JUD95]. Instead, the prior distribution is approximated by a symmetric set of points which capture the first three moments of the Gaussian distribution. Transforming these so called sigma points χ with the original non-linear system equations, an approximation of the mean and covariance of the transformed prior distribution can be calculated. This general method for approximating non-linear transformations has been introduced by Julie and Uhlmann [JU96] and is denoted as the unscented transformation (UT).

The Unscented Transformation

In [JU96] the UT is characterized with the intuition that it should be easier to approximate a Gaussian distribution using a fixed number of directly transformable parameters than it is to approximate an arbitrary non-linear function. Consequently, the UT approximates the non-linear transformation of multivariate Gaussian distributions by representing the distributions using a set of weighted sigma points χ chosen on a level curve of the Gaussian distribution. Each of the sigma points is directly transformable, which facilitates the calculation of the transformed Gaussian distribution entirely described by the mean and the covariance of the transformed sigma points.

To choose a symmetric set of sigma points representing the mean and covariance of a multivariate Gaussian, the positive and negative square roots of the covariance

matrix $\underline{P}$ shifted by the mean $\bar{x}$ of the Gaussian are used. Adding the Gaussian's mean leads to $2N + 1$ sigma points in case of an N dimensional Gaussian distribution:

$$\chi_0 = \bar{x} \quad (3.20)$$

$$\chi_{1:2N} = \bar{x} \pm \sqrt{(N + \kappa)\underline{P}}. \quad (3.21)$$

With κ the extent of the level curve, on which the sigma points are chosen, can be parametrized. The value $\kappa = 3 - N$ is widely used, since it minimizes the mean square error of the approximation up to the fourth order moment of the distribution [JU96]. The square root of $\underline{P}$ is commonly determined using the Cholesky decomposition.

The next step of the UT is the non-linear transformation of the sigma points:

$$\chi_x^* = f(\chi_x), \quad (3.22)$$

$$\chi_z^* = f(\chi_z). \quad (3.23)$$

While χ_x^* denotes the transformed set of sigma points of the state, χ_z^* represents those of the measurement. Thus, the transformed Gaussian's mean and covariance are approximated by:

$$\bar{x}^* = \frac{1}{N + \kappa} \left(\kappa \chi_{x,0}^* + \frac{1}{2} \sum_{i=1}^{2N} \chi_{x,i}^* \right) \quad (3.24)$$

$$\underline{P}^* = \frac{1}{N + \kappa} \left(\kappa (\chi_{x,0}^* - \bar{x}^*)(\chi_{x,0}^* - \bar{x}^*)^T + \frac{1}{2} \sum_{i=1}^{2N} (\chi_{x,i}^* - \bar{x}^*)(\chi_{x,i}^* - \bar{x}^*)^T \right). \quad (3.25)$$

The Equations (3.20)-(3.25) facilitate the non-linear transformation of a Gaussian distributed random variable without the need of a linearization.

Applying the UT to predict the state (3.9) and the measurement (3.12) as well as approximating the cross correlation matrix

$$\underline{P}_{+H} = \frac{1}{N + \kappa} \left(\kappa (\chi_{x,0}^* - \bar{x}^*)(\chi_{z,0}^* - \bar{z}^*)^T + \frac{1}{2} \sum_{i=1}^{2N} (\chi_{x,i}^* - \bar{x}^*)(\chi_{z,i}^* - \bar{z}^*)^T \right) \quad (3.26)$$

in (3.14) of the Kalman filter, leads to the UKF which outperforms the first-order EKF [JU96]. Equivalently to (3.24), the transformed measurement $\bar{z}^*$ is determined.

Figure 3.2: Structure of an IMM Kalman filter (KF) with two models.

3.2.4 The Interacting Multiple-Model Kalman Filter

Regardless of whether the system is linear or non-linear, the used model has to be suitable to predict the object's motion in every situation. However, in tracking maneuvering objects, it is difficult to design a motion model which is able to accurately represent all varying kinds of object motion. Hence, in multiple-model (MM) filtering a finite set of models is defined which covers all motion possibilities. This parametrized finite set of state models is also known as jump Markov system (JMS) which is generally introduced in Section 4.4.1. One example for a JMS is to use a constant position (CP), a constant velocity (CV), and a constant acceleration (CA) model to represent different motion states of a vehicle. To integrate the mode of the system in the estimation process, the augmented state

$$\overset{o}{x} = (x, o), \quad (3.27)$$

where x is the state of the object and o represents the corresponding motion model, is introduced. The state (3.27) is also denoted as hybrid state, since it has both continuous uncertainties in case of the state and discrete uncertainties in case of the model. The most popular representative of MM filters in single-object tracking is the interacting multiple-model (IMM) Kalman filter [BB88; BL98]. Figure 3.2 visualizes the structure of an IMM filter with $N_o = 2$ models (o_1 and o_2). Based on each model an individual Kalman filter estimates the object's state. The popularity of the filter is due to the interaction step just before the state prediction. The interaction

provides an individual combination of the filters' posteriors for every single filter, which considers the validity of the models at the current time step and the history of the model transitions. This is done by modeling the model transition probabilities over time as a Markov process. Thus, assuming a linear Gaussian system, the model probabilities can be predicted by the weighted sum of the posterior model probabilities:

$$\mu_{o_i,+} = \sum_{j=1}^{N_o} t_{j,i} \mu_{o_j}. \quad (3.28)$$

The weights $t_{i,j}$ are the elements of the Markovian transition matrix

$$T = \begin{pmatrix} t_{1,1} & \dots & t_{1,N_o} \\ \vdots & \ddots & \vdots \\ t_{N_o,1} & \dots & t_{N_o,N_o} \end{pmatrix} \quad (3.29)$$

which can be interpreted as a process model for the transition between the discrete models o_i . Each element $t_{i,j}$ represents the probability of transition from model o_i to o_j . Similar to the motion model in the state prediction, the transition probability matrix is supposed to accurately model the change of the object's motion model at each time step. Modeling the motion of the object with a fixed set of models, all rows of T must sum to unity, since one of the models is assumed to be appropriate. Receiving a new measurement, the model probabilities are updated according to:

$$\mu_{o_i} = \frac{\Lambda_i \mu_{o_i,+}}{\sum_{j=1}^{N_o} \Lambda_j \mu_{o_j,+}}. \quad (3.30)$$

The likelihood function Λ_i for the i^{th} filter with model o_i weights the predicted model probabilities according to the normalized innovation squared (NIS) error of the individual filters:

$$\Lambda_i = \frac{1}{\sqrt{\det(2\pi \underline{S}_i)}} \exp \left(-\frac{1}{2} (\gamma_i^T \underline{S}_i^{-1} \gamma_i) \right). \quad (3.31)$$

Here the innovation covariance $\underline{S}_i$ and the measurement residuum γ_i of the i^{th} filter are determined according to (3.12) and (3.13), respectively.

After the estimation of the model probabilities which is illustrated on the right side of Figure 3.2, the mixing probabilities to perform the interaction between the filters and calculate individual posteriors for the filters are determined. The mixing

probabilities are defined as

$$\mu_{i,j} = \frac{t_{i,j}\mu_{o_i}}{\mu_{o_j,+}} \quad (3.32)$$

which expresses the influence of the estimate of the i^{th} Kalman filter on the new mixed state for the j^{th} filter. Consequently, the mixed state and covariance are:

$$\hat{x}_j^0 = \sum_{i=1}^{N_o} \hat{x}_i \mu_{i,j}, \quad (3.33)$$

$$\underline{P}_j^0 = \sum_{i=1}^{N_o} \mu_{i,j} (\underline{P}_i + (\hat{x}_i - \hat{x}_j^0)(\hat{x}_i - \hat{x}_j^0)^T) \quad (3.34)$$

which are calculated as input for each filter in the interaction step. After the iteration of the Kalman filters according to the already introduced equations, a distinct estimate of the IMM Kalman filter is received by combination of the model conditioned state and covariance estimates according to the Equations (3.33) and (3.34). Here the mixing probabilities $\mu_{i,j}$ are replaced by the updated model probabilities μ_{o_i} (3.30) and the current filter estimates are used. In case of a high variety of motion possibilities or improper models, the combined estimate suffers from an accuracy degradation which is mainly due to the influence of unnecessary or wrong models [LB96]. Therefore, in [LB96] the variable structure multiple-model approach has been introduced in which the set of models is variable to remove unnecessary models.

3.3 Multi-Object Tracking

The preceding section introduced the Kalman filter for single-object tracking assuming the presence of only one track and one measurement. Albeit, many applications require the simultaneous tracking of multiple objects confirmed by multiple measurements. In the last decades several approaches exploiting a bank of Kalman filters have been proposed. Each approach has to deal with the ambiguity of measurement-to-track association which is the core problem in multi-object tracking.

Independent of the objects' type, the sensors, and methods to detect the objects, multi-object tracking always suffers from inaccurate measurements, missed detections, and false alarms. Thus, the data association module has to be able to cope with these cases. In Figure 3.3 the processing chain of a classical multi-instance tracker is illustrated. The input is an object list which is generated by an arbitrary object

Figure 3.3: Multi-object tracking structure based on multiple instances of single-object trackers as in [Mah09b].

detection or segmentation module. The object list is composed of measurements representing detected objects which might be used to innovate the state of the tracks and clutter measurements. Thus, a measurement-to-track association algorithm is required which assigns the gathered measurements to the currently available object individual single-object trackers. Subsequently, each tracker innovates its state estimate using the assigned measurement and provides its estimate to the list of tracks. Finally, measures like the age, existence, or type of tracks are used to confirm tracks in the validation or classification stage.

3.3.1 Gating

The complexity of the data association approaches directly depends on the number of relevant measurement-to-track pairings. To reduce this number, the gating step eliminates unlikely measurement-to-track pairings. In literature several types of gates are present [BP99] but in Kalman filter applications the ellipsoidal gate defined by the quadratic Mahalanobis distance (MHD)

$$d_{MHD}^2(z_i, z_{j,+}) = (z_i - z_{j,+})^T \underline{S}_j^{-1} (z_i - z_{j,+}) \quad (3.35)$$

is common. It is calculated between the predicted measurement $z_{j,+}$ of j^{th} filter and the current measurement z_i with the innovation covariance $\underline{S}_j$ (3.12). The quadratic MHD spans an ellipsoidal region in the measurement space which contains likely measurement-to-track pairings [BL93, p. 95]. All measurements outside this region are neglected. The extent of the ellipsoid is controlled by the confidence level α . Thus, the measurements inside the gate of j^{th} track are:

$$Z_j = \{z : d_{MHD}^2(z, z_{j,+}) < \alpha\}. \quad (3.36)$$

The parameter α has to be chosen carefully, since too low values might gate relevant measurements. In case of a Gaussian system, the quadratic MHD is χ^2 distributed with $\dim(z)$ degrees of freedom. Hence, the confidence level α can be determined according to the required probability that the true measurement is located within the gating region.

3.3.2 Data Association Approaches

The nearest neighbor (NN) association algorithm is by far the simplest algorithm to associate a set of measurements Z with N_z elements to a set of tracks X including N_x tracks. The algorithm sequentially seeks for measurement-to-track associations, starting with an arbitrary track, by minimizing the MHD (3.35) between the track and the measurements within the gating region. The NN association only performs well in cases of well-separated objects compared to the sensor resolution, i.e. not overlapping gating regions which include only one measurement. Additionally, the result of the NN algorithm depends on the processing order. Considering $N_z = 1$ measurement in the gating region of $N_x = 2$ tracks, the NN algorithm will always assign the measurement to the first investigated track.

To overcome these drawbacks, the global nearest neighbor (GNN) approaches [BP99, Sec. 6.4] might be useful. This class of association algorithms facilitates to resolve ambiguities and processing order dependencies by minimizing a global association distance. The common principle of all GNN approaches is that they consider all measurement-to-track associations using an assignment matrix $\underline{A}$. The elements $a_{i,j}$ of the assignment matrix represent the likelihood or cost for the assignment of track i to measurement j . Finally, approaches like the Hungarian method [Kuh55], the Munkres algorithm [BL71; Mun57], or the auction algorithm [Ber88] yield the most likely measurement for each track.

The NN and GNN approaches provide hard association decisions for each track which yields good results in situations with well separated objects and no clutter. However, in the majority of cases real tracking applications have to be able to track multiple

objects in clutter even if they are close together. Thus, it is essentially not possible to make a hard association decision, since it might lead to track losses and false tracks. Consequently, other data association algorithms, like the multi-hypothesis tracking (MHT) [Rei79] and the probabilistic data association (PDA) [BT75], were developed which avoid hard association decisions [BB00]. The MHT and PDA differ in the way of resolving ambiguous associations.

The MHT sets up several possible track hypotheses and expects that the subsequent measurements resolve the conflict [BP99; Rei79]. Similar to NN approaches only one measurement is used to update a track but several hypotheses can be updated by the same measurement.

The PDA avoids ambiguous measurement-to-track assignments at an earlier stage. For each existing track a set of updated tracks based on the measurements within the gating region of the track is calculated using Equations (3.15) and (3.16). Finally, the updated estimate for each track is determined by a weighted sum over the elements of the respective sets. Commonly, the weight is based on a distance measure between the track and the measurement. Thus, several tracks might be influenced by the same measurement, which increases the uncertainty of the estimates. Since the PDA has originally been developed to track a single object in a cluttered environment, the joint probabilistic data association (JPDA) [BF88; FBS83] extension facilitates the tracking of multiple objects in clutter. Here again the tracks are updated including the weighted residuals of all measurements within the gating region, but the weight of the measurements which are present in multiple gating regions is reduced. In [BP99, p. 353] further details are given. The JPDA performs better in situations with dense objects, i.e. measurements are within the gating region of multiple tracks. To additionally cover the initialization and deletion of tracks, Musicki et al. [MES94] introduced the integrated probabilistic data association (IPDA) which uses a Markov chain for the object existence. Finally, the joint integrated probabilistic data association (JIPDA) by [ME04] combines the joint association probabilities in the filter update and the existence estimation.

This section introduced only a few classes of widely used assignment approaches. A more complete overview is given in [BB00]. Beside the use of generic association methods, application specific methods also exploit characteristic features of the measurement for data association. One example is given in Section 6.2.3.

3.4 Motion Models

As already discussed in Section 3.2, motion models are necessary to predict the change of the object's state between two time steps. Dependent on the type of the object and its current state of motion, different models may be appropriate. Since in this thesis only position measuring sensors are used and errors resulting from a double differentiation should be avoided, only models based on the constant velocity (CV) assumption are used.

In the following, two motion models assuming CV are introduced. For a more comprehensive review of motion models in tracking applications refer to [BB00; BF88]. The first model is denoted as the linear constant velocity (LCV) model which assumes a straight motion and does not consider the current orientation or yaw rate of the object. Due to its linearity, it allows an optimal propagation of the posterior distribution according to the linear state transition equation (3.9) with the Cartesian state space

$$x^{LCV} = \begin{pmatrix} x_p \\ y_p \\ \dot{x}_p \\ \dot{y}_p \end{pmatrix} \quad (3.37)$$

and the process matrix

$$\underline{F}^{LCV} = \begin{pmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (3.38)$$

Substituting (3.37) and (3.38) in (3.9), the predicted state composed of the position in x - and y -direction (x_p, y_p) and its velocities ($\dot{x}_p, \dot{y}_p$) after one measurement cycle with the duration T is determined. For systems with no constant cycle time, the prediction time T is variable and results from the time gap between measurements updating the state. The LCV model can be easily extended for further state dimensions.

The second model is the constant turn rate and velocity (CTRV) model which is a curvilinear model and non-linear in the states. In this model the motion direction corresponds to the orientation which prevents lateral movements. In case of the

CTRV model the state space is defined as:

$$x^{CTRV} = \begin{pmatrix} x_p \\ y_p \\ |v| \\ \phi \\ \dot{\phi} \end{pmatrix}. \quad (3.39)$$

Due to the non-linearity of the model, the predicted state is calculated by using the function

$$f(x^{CTRV}) = x^{CTRV} + \begin{pmatrix} \frac{|v|}{\dot{\phi}} (\sin(\dot{\phi}T + \phi) - \sin(\phi)) \\ \frac{|v|}{\dot{\phi}} (\cos(\phi) - \cos(\dot{\phi}T + \phi)) \\ 0 \\ \dot{\phi}T \\ 0 \end{pmatrix} \quad (3.40)$$

in (3.4). Here, the velocity is represented in spherical coordinates, since the movements with velocity $|v|$ are only permitted in direction of the object's orientation ϕ (yaw angle). Additionally, the influence of the yaw rate $\dot{\phi}$ is considered. Since (3.40) has a singularity for $\dot{\phi} = 0$, the second summand is replaced with

$$f(x^{CTRV}) = x^{CTRV} + \begin{pmatrix} |v|T \cos(\phi) \\ |v|T \sin(\phi) \\ 0 \\ \dot{\phi}T \\ 0 \end{pmatrix} \quad (3.41)$$

if $|\dot{\phi}| < \epsilon$.

For both models an additive, white, Gaussian noise is assumed. In case of the LCV model the noise models velocity changes in the direction of the considered motion directions. Beside the velocity changes, the CTRV model additionally covers the changes of the yaw rate. Depending on the model, the position estimates vary in their uncertainty as visualized by Figure 3.4.

Here the region, surrounding the object in red, represents the position uncertainty and by association the motion capabilities of the object according to the motion model. As already mentioned the orientation and motion direction are not coupled in the LCV model (Figure 3.4a). Thus, the uncertainty region is circular. This behavior is appropriate to model objects with high agility. On the other hand, the uncertainty region for the CTRV model (Figure 3.4b) is ellipsoidal and oriented along the motion direction of the object. This models a rigid motion where no lateral

Figure 3.4: Motion models with characteristic uncertainties.

movements are permitted. This kind of model is common to predict the motion of vehicles [SAO⁺11].

3.4.1 Combining Different Motion Models

As pointed out by [BP99, p. 227], the gain of multiple-model (MM) methods is most significant in cases where the state models represent different types of object motion like the introduced LCV and CTRV state models (see Figure 3.4). To realize the model interaction in the interacting multiple-model (IMM) Kalman filter which involves a mixing of the model specific states and covariances, a transformation between the Cartesian LCV and the polar CTRV representation of the velocity is required. Due to $(x_p, y_p)^{LCV} = (x_p, y_p)^{CTRV}$ no position transformation is needed. The remaining transformation from CTRV to LCV states are:

$$\dot{x}_p = |v| \cos(\phi) \quad (3.42)$$

$$\dot{y}_p = |v| \sin(\phi) \quad (3.43)$$

and from LCV to CTRV:

$$|v| = \sqrt{\dot{x}_p^2 + \dot{y}_p^2} \quad (3.44)$$

$$\phi = \text{atan2}(\dot{y}_p, \dot{x}_p) \quad (3.45)$$

$$\dot{\phi} = 0. \quad (3.46)$$

Both (3.42), (3.43) in ϕ and (3.44), (3.45) in $(\dot{x}_p, \dot{y}_p)$ show non-linearities. Therefore, the UT introduced in Section 3.2.3 is exploited to transform the states and covariances. Special attention has to be paid to the discontinuities in (3.44) and (3.45). In the vicinity of this discontinuities which are at $\dot{x}_p = 0$ and $\dot{y}_p = 0$ the UT yields inaccurate transformation results. Thus, the nonlinear transformation in this region is approximated by linearization of (3.44) and (3.45) as described in Section 3.2.2.

Chapter 4

Fundamentals in Probability Hypothesis Density Filtering

The multi-object tracking methods presented in the previous chapter are based on a bank of object individual single-object trackers which require measurement-to-track association. The parallel usage of independent single-object trackers results in difficulties when dealing with nearby objects and clutter. Further, the birth and disappearance of tracks is often based on heuristics.

To improve the multi-object tracking performance with a bank of single-object trackers, a multitude of extensions has been presented in literature. Hence, Mahler indicated in [Mah04] that algorithms can always be cobbled together to fix a problem. But this involves the risk of difficulties in behaviour diagnoses, generality, numerical stability, and convergence. To overcome these drawbacks, Mahler introduced the probability hypothesis density (PHD) filter in [Mah03; Mah07a] which is an approximation of the multi-object Bayes filter [Mah07a] and based on finite set statistics (FISST). The major attributes of the probability hypothesis density (PHD) filter and according to that the major differences to the multi-object tracking methods discussed in Section 3.3 are:

- unifies two goals of multi-object tracking: estimation of the number of objects and their individual states
- based on statistical multi-sensor multi-object models
- does not require explicit measurement-to-track association

The aim of this chapter is to introduce the fundamentals of PHD filtering, give implementation possibilities, and draw analogies to the single-object case in Chapter 3. With the derivation of the multi-object Bayes filter in [Mah07a] a basis was established to derive theoretically principled approximations which are computational tractable just as the Kalman filter in the single-object case. One key component in the

derivation of the multi-object Bayes filter are random finite sets (RFSs). Thus, they are the topic of the subsequent section.

4.1 Random Finite Sets

In single-object tracking, the object is completely described by a random state vector x . As introduced in Section 3.3, classical multi-object tracking approaches use object individual Kalman filters to estimate the state vectors of the objects. However, in general objects may appear or disappear from the perception area of the system. Consequently, the number of objects is also random which is not explicitly modeled in classical approaches. Therefore, in finite set statistics (FISST) the objects are not described with individual and independent random vectors in the state space but with a random finite set (RFS)

$$\mathbf{X} = \{x^{(1)}, \dots, x^{(n)}\} \quad (4.1)$$

composed of a random number $|\mathbf{X}| = n \geq 0$ of unordered random state vectors $x^{(1)}, \dots, x^{(n)}$ as depicted in Figure 4.1. With that, also the empty set $\mathbf{X} = \emptyset$ is possible.

Figure 4.1: Mapping of individual and independent random vectors modeling the tracks and measurements to an RFS representing the multi-object measurement or state.

Similar arguments apply to the measurement space. It is not guaranteed that the sensor detects every object in every measurement cycle. Additionally, false alarms or clutter measurements can influence the number of detected objects. Figure 4.2 depicts an example output of an image-based vehicle detector. Obviously, the van is not detected while the traffic sign originates a false alarm. Since the van might be detected or the traffic sign might not originate a false alarm in the next image, an

Figure 4.2: Output of a video camera-based vehicle detector.

RFS is perfectly suitable to model the measurements. Thus, all measurements are combined in the multi-object measurement

$$\mathbf{Z} = \{z^{(1)}, \dots, z^{(m)}\} \quad (4.2)$$

which is an RFS consisting of $|\mathbf{Z}| = m$ unordered random measurement vectors $z^{(1)}, \dots, z^{(m)}$, while $m \geq 0$ is a random variable.

4.1.1 Multi-Object Probability Distributions

The counterpart to a probability density of a random state vector is the multi-object probability density or multi-object probability distribution

$$\pi(\mathbf{X}) = \begin{cases} \pi(\emptyset) & \text{if } \mathbf{X} = \emptyset, \quad |\mathbf{X}| = 0 \\ \pi(\{x_1\}) & \text{if } \mathbf{X} = \{x_1\}, \quad |\mathbf{X}| = 1 \\ \pi(\{x_1, x_2\}) & \text{if } \mathbf{X} = \{x_1, x_2\}, \quad |\mathbf{X}| = 2 \\ \vdots & \vdots \end{cases} \quad (4.3)$$

Beside the form of the multi-object probability distribution, the units of measurement also vary with the number of objects. While $\pi(\emptyset)$ is unitless, the units of $\pi(\mathbf{X})$ are $u^{-|\mathbf{X}|}$ with u being the units of x and $|\mathbf{X}|$ being the number of elements in the set $\mathbf{X}$. Analogously to the single object probability distribution, the multi-object probability distribution represents the uncertainty about the multi-object state $\mathbf{X}$ which is composed of the uncertainty of the individual states as well as the number of objects. Further requirements on $\pi(\mathbf{X})$ to be a multi-object probability distribution

are that $\pi(\mathbf{X}) \geq 0$ for all $\mathbf{X}$ and:

$$\int \pi(\mathbf{X}) \delta \mathbf{X} = 1. \quad (4.4)$$

Since $\pi(\mathbf{X})$ in (4.4) is a multi-object probability density, a set integral is required to integrate over it. The set integral sums over all possible numbers of objects and is defined as:

$$\int \pi(\mathbf{X}) \delta \mathbf{X} = \sum_{i=0}^{\infty} \frac{1}{i!} \int \pi(\{x_1, \dots, x_i\}) dx_1, \dots, x_i \quad (4.5)$$

A very illustrative example for a multi-object probability distribution is presented in [Mah07a, pp. 350] which is reviewed here to clarify the meaning of a multi-object probability distribution. Considering a single twinkling star in night sky, whose position features the spatial distribution $p(x)$ due to atmospheric turbulence. With a probability of observability r the multi-object probability density yields

$$\pi(\mathbf{X}) = \begin{cases} 1 - r & \text{if } \mathbf{X} = \emptyset \\ r \cdot p(x) & \text{if } \mathbf{X} = \{x\} \\ 0 & \text{if otherwise} \end{cases} \quad (4.6)$$

which represents both the spatial and the cardinality uncertainty. Due to the expectation of only a single star, all values of $\pi(\mathbf{X})$ for $|\mathbf{X}| \geq 2$ are zero.

In the succeeding part of this section, multi-object probability distributions essential for the PHD filter are introduced. For a more comprehensive overview please refer to [Mah07a].

Multi-Object Dirac Delta Distribution

The multi-object Dirac delta probability distribution $\delta_{\mathbf{Y}}(\mathbf{X})$ centered at $\mathbf{Y}$ has the convenient property:

$$\int \delta_{\mathbf{Y}}(\mathbf{X}) \cdot \pi(\mathbf{X}) \delta \mathbf{X} = \pi(\mathbf{Y}) \quad (4.7)$$

for any function $\pi(\mathbf{X})$ of a finite set variable $\mathbf{X}$. It is defined by

$$\delta_{\mathbf{Y}}(\mathbf{X}) = \begin{cases} 0 & \text{if } |\mathbf{Y}| \neq |\mathbf{X}| \\ 1 & \text{if } \mathbf{Y} = \mathbf{X} = \emptyset \\ \sum_{\sigma} \delta_{y_{\sigma 1}}(x_1) \cdots \delta_{y_{\sigma n}}(x_n) & \text{if } \mathbf{Y} = \{y_1, \dots, y_n\}, \mathbf{X} = \{x_1, \dots, x_n\} \end{cases}, \quad (4.8)$$

where the summation is taken over all permutations on the numbers $1, \dots, n$. The Dirac delta function

$$\delta_y(x) = \begin{cases} 0 & \text{if } y \neq x \\ \infty & \text{if } y = x \end{cases} \quad (4.9)$$

is characterized by the fact that

$$\int f(x) \delta_y(x) dx = f(y), \quad (4.10)$$

for every ordinary function $f(x)$.

Multi-Object Poisson Distribution

The multi-object probability distribution of a Poisson distributed RFS $\mathbf{X} = \{x^{(1)}, \dots, x^{(n)}\}$ is defined as

$$\pi(\mathbf{X}) = e^{-\lambda} \cdot \lambda^n \cdot p(x^{(1)}) \cdots p(x^{(n)}). \quad (4.11)$$

The distribution of the i^{th} object is represented by the probability density $p(x^{(i)})$ and λ denotes the estimated number of objects.

The probability hypothesis density (PHD) of a Poisson distributed RFS is

$$\kappa(z) = \lambda_c \cdot p(z) \quad (4.12)$$

and central to the theory of PHD filters. The PHD and the PHD filter are introduced in Section 4.3. The Poisson RFS is also commonly used in multi-object tracking to model the PHD of the clutter process with an average number of λ_c clutter measurements per time step. In this special case, the probability distribution $p(z)$ is assumed to be an uniform distribution over the measurement space.

Multi-Object Bernoulli Distribution

A Bernoulli process is characterized by two items: a spatial probability density $p(x)$ and an existence probability r which expresses the probability that an object with $p(x)$ exists. Thus, a Bernoulli distributed RFS is given by

$$\pi(\mathbf{X}) = \begin{cases} 1 - r & \text{if } \mathbf{X} = \emptyset \\ r \cdot p(x) & \text{if } \mathbf{X} = \{x\} \\ 0 & \text{if } |\mathbf{X}| \geq 2 \end{cases} . \quad (4.13)$$

In (4.13) another characteristic of a Bernoulli process becomes apparent. Since $\pi(\mathbf{X}) = 0$ if $|\mathbf{X}| \geq 2$, the Bernoulli distribution only models the existence of a singleton.

To model the distribution of an RFS with more than one element using a Bernoulli process, the multi-Bernoulli process is used. Under the assumption that all M Bernoulli RFSs $\mathbf{X}^{(i)}$ are independent of each other, the multi-Bernoulli distributed RFS is $\mathbf{X} = \bigcup_{i=1}^M \mathbf{X}^{(i)}$ and the multi-object distribution of a multi-Bernoulli distributed RFS $\mathbf{X} = \{x^{(1)}, \dots, x^{(n)}\}$ is

$$\pi(\mathbf{X}) = \prod_{j=1}^M (1 - r^{(j)}) \sum_{1 \leq i_1 \neq \dots \neq i_n \leq M} \prod_{j=1}^n \frac{r^{(i_j)}}{1 - r^{(i_j)}} . \quad (4.14)$$

Beside of the probability that none of the M objects exists $\pi(\emptyset) = \prod_{j=1}^M (1 - r^{(j)})$ it is necessary to sum over all $n \leq M$ permutations of the state vectors $x^{(i)}$. According to [Mah07a, p. 584] the PHD of a multi-Bernoulli distributed RFS is:

$$v(x) = \sum_{i=1}^M r^{(i)} p(x^{(i)}) \quad (4.15)$$

As introduced in [Mah07a], the multi-Bernoulli process is central to the multi-object measurement model (Section 4.2.1) and the multi-object motion model (Section 4.2.2).

4.2 The Multi-Object Bayes Filter

Based on finite set statistics (FISST), Mahler [Mah07a] introduced the multi-object Bayes filter which is a top-down formulation of the multi-object tracking problem. It extends the single-object Bayes filter (Section 3.1) by substituting the single-object distributions with multi-object distributions $\pi(\mathbf{X})$ which comprise the uncertainty

in the objects' number and their individual states (see Figure 4.1). The prediction and update of the multi-object distribution according to [Mah07a] are:

$$\pi_+(\mathbf{X}|\mathbf{Z}^k) = \int f(\mathbf{X}_+|\mathbf{X})\pi(\mathbf{X}|\mathbf{Z}^k)\delta\mathbf{X}, \quad (4.16)$$

$$\pi(\mathbf{X}|\mathbf{Z}^{k+1}) = \frac{g(\mathbf{Z}|\mathbf{X})\pi_+(\mathbf{X}|\mathbf{Z}^k)}{\int g(\mathbf{Z}|\mathbf{X})\pi_+(\mathbf{X}|\mathbf{Z}^k)\delta\mathbf{X}}. \quad (4.17)$$

Both steps are very similar to the prediction (3.1) and update (3.2) in the single-object case but include the calculation of set integrals (4.5). The superscript at the multi-object measurement $\mathbf{Z}$ represents the dependency on all previous measurements, thus $\mathbf{Z}^k$ represents the history of measurement RFSs $(\mathbf{Z}_1, \dots, \mathbf{Z}_k)$. Analogous to the single-object Bayes filter, the multi-object Bayes filter requires two prior densities: the multi-object Markov transition density $f(\mathbf{X}_+|\mathbf{X})$ and the multi-object measurement likelihood function $g(\mathbf{Z}|\mathbf{X})$. The densities will be discussed in the next two Sections.

To implement the multi-object Bayes filter, sequential Monte-Carlo (SMC) methods similar to the single-object Bayes filter are used [Mah07a; VSD05]. In [RWW⁺13], a real-time capable implementation of the filter based on an approximation of the computationally complex multi-object likelihood function (4.19) is presented. Additionally, there exist a whole series of approximations of the multi-object Bayes filter. Some of these are: the PHD filter [Mah03], the cardinalized probability hypothesis density (CPHD) filter [Mah07b], and the multi-Bernoulli filter [Mah07a; VVC09]. Recently, Vo and Vo [VV13b] proposed the generalized labeled multi-Bernoulli (GLMB) filter which facilitates an analytical implementation of the multi-object Bayes filter. Whereas the complexity of the GLMB filter's prediction and update is cubic, the labeled multi-Bernoulli (LMB) filter published in [RVVD14b] approximates the posterior GLMB distribution by an LMB RFS, which enables a prediction with linear complexity but implicates a loss of cardinality information. However, the update of the LMB RFS is still cubic in complexity. Due to its low computational complexity, the PHD filter is still a very popular approximation and therefore used in this thesis. An introduction to the filter and its implementation is given in Section 4.3.

4.2.1 Multi-Object Likelihood Functions

The multi-object likelihood function $g(\mathbf{Z}|\mathbf{X})$ which is involved in the Bayesian update of the multi-object posterior (4.17) has to comprehensively describe the sensor. This is a huge difference to the single-object case, where the likelihood function mainly incorporates the transformation from the state space to the measurement space and

the measurement uncertainties. The multi-object likelihood function additionally considers the sensor's characteristics in terms of its probability of detection, FOV, false alarm rate, and data transmission drop-outs to name only a few.

To encapsulate the sensor's characteristics, the standard multi-object measurement model was introduced in [Mah07a] which is based on the following assumptions:

- a sensor detects a single object in the scene with the single-object measurement likelihood function $g(z|x)$ (Section 3.1) and no measurement is generated by more than one object.
- each object in the scene with state x generates either a single detection with the detection probability $p_D(x)$ or is not detected with the missed detection probability $q_D(x) = 1 - p_D(x)$. Consequently, each state $x \in \mathbf{X}$ generates an RFS $\Upsilon(x)$ that either consists of $\Upsilon(x) = \{z\}$ if the object is detected or is the empty set $\Upsilon(x) = \emptyset$. Thus, the measurement process follows a Bernoulli distribution (4.13).
- false alarms (clutter) are modeled by a Poisson distributed RFS $\mathbf{C}$ (4.11) with an estimated number of λ_C clutter measurements with spatial distribution $c(z)$. To accentuate the distributions and variables associated with clutter, they are denoted with $c(\cdot)$ or the subscript C .
- the measurements are independent of the objects' states.

Under the assumption that all object generated measurements and all clutter measurements are statistically independent, the RFS measurement model has the following form

$$\mathbf{Z} = \Upsilon(x_1) \cup \dots \cup \Upsilon(x_n) \cup \mathbf{C} = \Upsilon(\mathbf{X}) \cup \mathbf{C}, \quad (4.18)$$

i.e. it consists of a multi-Bernoulli distributed RFS $\Upsilon(\mathbf{X})$ with $|\Upsilon(\mathbf{X})| = |\mathbf{X}| = n$, representing the object generated detections or missed detections and a Poisson distributed RFS $\mathbf{C}$ for the clutter process. The three cases considered in $\mathbf{Z}$ are illustrated in Figure 4.3. Given the condition that both not every object generates a measurement and not every measurement is object originated, the measurement-to-track association which is common in classical multi-object tracking systems (see Section 3.3.2) might be ambiguous. Hence, the multi-object likelihood function

Figure 4.3: Illustration of the standard multi-object measurement model as in [Mah07a].

[Mah07a, p. 410]

$$g(\mathbf{Z}|\mathbf{X}) = \pi_C(\mathbf{Z})\pi(\emptyset|\mathbf{X}) \sum_{\theta} \prod_{i:\theta(i)>0} \frac{p_D(x^{(i)}) \cdot g(z_{\theta(i)}|x^{(i)})}{(1 - p_D(x^{(i)})) \cdot \lambda_{CC}(z_{\theta(i)})}, \quad (4.19)$$

$$\pi(\emptyset|\mathbf{X}) = \prod_{i=1}^n (1 - p_D(x^{(i)})), \quad (4.20)$$

$$\pi_C(\mathbf{Z}) = e^{-\lambda_C} \prod_{z \in \mathbf{Z}} \lambda_{CC}(z), \quad (4.21)$$

averages over all possible measurement-to-track association hypotheses θ which include the missed detection case [Mah07a, p. 332]. With $\pi(\emptyset|\mathbf{X})$ representing the case that none of the objects is detected and $\pi_C(\mathbf{Z})$ the case that all measurements $z \in \mathbf{Z}$ are false alarms. Figure 4.4 is used in this work. Beside the objects' motion in the state space between consecutive measurements, the multi-object Markov density additionally considers a change in the number of objects in the prediction step (4.16) of the multi-object Bayes filter. This may happen when objects leave or enter the perception area of the system or an object spawns other objects, e.g. an aircraft releasing missiles. Since spawning has not to be considered in the investigated applications, the standard multi-object motion model without spawning [Mah07a, p. 471] is used in this work.

4.2.2 Multi-Object Markov Densities

The standard multi-object motion model implies the following assumptions:

- each object $x \in \mathbf{X}$ either still exists at the arrival time of the next measurement with the survival probability $p_S(x)$ or disappears with the probability $1 - p_S(x)$. Similar to the standard multi-object measurement model, each existing object $x_i \in \mathbf{X}$ generates an RFS $\mathbf{S}(x_i)$ which can take on either $\{x_{i,+}\}$ when the object survives or $\emptyset$ when it disappears. Thus, the survival process of the multi-object measurement model is again a Bernoulli RFS (4.13).
- each object evolves in time independent of other objects according to a single-object Markov transition density $f(x_+|x)$ (3.3).
- the birth of new objects is modeled using a problem specific distribution of the RFS $\mathbf{B}$ which is independent of the existing objects.

Consequently, the predicted set $\mathbf{X}_+$ is composed of the union of surviving objects $\mathbf{S}(\mathbf{X})$ and new born objects $\mathbf{B}$:

$$\mathbf{X}_+ = \mathbf{S}(x_1) \cup \dots \cup \mathbf{S}(x_n) \cup \mathbf{B} = \mathbf{S}(\mathbf{X}) \cup \mathbf{B}, \quad (4.22)$$

where $|\mathbf{X}| = n$. These three cases are illustrated in Figure 4.4. Modeling the birth RFS $\mathbf{B}$ as Poisson distributed, the motion model is exactly analogous to the measurement model (4.19) and given by [Mah07a] as:

$$f(\mathbf{X}_+|\mathbf{X}) = \pi_B(\mathbf{X}_+)\pi(\emptyset|\mathbf{X}) \sum_{\theta} \prod_{i:\theta(i)>0} \frac{p_S(x^{(i)}) \cdot f(x_+^{(\theta(i))}|x^{(i)})}{(1 - p_S(x^{(i)})) \cdot \lambda_B b(x_+^{(\theta(i))})}. \quad (4.23)$$

Similar to (4.19), the summation goes over all possible associations θ where $\theta(i) = 0$ expresses the disappearance of an object i and $\theta(i) > 0$ represents that object i survives. The expected number of new born objects is λ_B which are distributed according to $b(x_+)$. Thus, the possibility that all objects in $\mathbf{X}_+$ just appeared is denoted by the density

$$\pi_B(\mathbf{X}_+) = e^{-\lambda_B} \prod_{i=1}^{|\mathbf{X}_+|} \lambda_B b(x_+^{(i)}) \quad (4.24)$$

and

$$\pi(\emptyset|\mathbf{X}) = \prod_{i=1}^{|\mathbf{X}|} (1 - p_S(x^{(i)})) \quad (4.25)$$

Figure 4.4: Illustration of the standard multi-object motion model as in [Mah07a].

models the case that none of the existing objects survives.

4.3 The Probability Hypothesis Density Filter

The probability hypothesis density (PHD) filter [Mah03] is the multi-object counterpart of the constant gain Kalman filter described in Section 3.2.1. Therefore, in Figure 4.5, which is the multi-object counterpart to Figure 3.1, the PHD filter is depicted at the same position as the constant gain Kalman filter in Figure 3.1. The PHD filter approximates the multi-object Bayes filter by neglecting all statistical moments of the multi-object probability distributions in (4.16) and (4.17) with order higher than one. Propagating only the first-order moment of $\pi(x|\mathbf{Z}^k)$ which is the PHD $v(x|\mathbf{Z}^k)$ instead of the entire multi-object density, significantly reduces the computational complexity. This facilitates the implementation of a tractable filter which does not require measurement-to-track association and enables explicit statistical models for missed detection, sensors' field of view (FOV), and false alarms.

At the beginning of this section the PHD and some of its major properties are described. Afterwards the PHD filter equations and their required assumptions are introduced. Based on this knowledge the Gaussian mixture (GM) implementation of

Figure 4.5: Visualization and overview of common multi-object tracking algorithms, their recursion steps, and their analogy to single-object tracking algorithms of Figure 3.1. In the measurement space objects are represented by ellipses, measurements by stars, and multi-object measurements by dotted ellipses. In the state space tracks are illustrated by crossed ellipses and multi-object states by dotted ellipses.

the PHD filter is given.

4.3.1 The Probability Hypothesis Density

The probability hypothesis density (PHD) $v_k(x|Z^k)$ has been adopted by [Mah03] from point process theory [DV88] where it is denoted as the first-moment density or intensity density. The PHD is not a probability density, since it is uniquely defined by

$$\hat{N}_S = \int_S v(x|Z^k) dx. \quad (4.26)$$

A visualization of a PHD over a two dimensional state space is shown in Figure 4.6. Given an arbitrary region S of the single-object state space, the integral on the right side of (4.26) gives an estimate of the number of objects $\hat{N}_S$ in S . Expanding S to the entire single-object state space, $\hat{N}$ is the expected number of objects based on $v(x|\mathbf{Z}^k)$. Thus, the peaks in Figure 4.6 indicate locations in state space with a high object density.

Figure 4.6: Illustration of a PHD defined over a two dimensional state space.

In [Mah03], Mahler showed that $v(x|\mathbf{Z}^k)$ matches the first-order multi-object moment of an RFS $\mathbf{X}$ with the multi-object probability distribution $\pi(\mathbf{X}|\mathbf{Z}^k)$. Since the naive definition of a first-order multi-object moment

$$E[\mathbf{X}] = \int \mathbf{X} \cdot \pi(\mathbf{X}|\mathbf{Z}^k) \delta \mathbf{X} \quad (4.27)$$

is mathematically undefined, the RFS $\mathbf{X}$ is substituted by the Dirac delta density (4.8) concentrated at $\mathbf{X}$. Thus, the multi-object counterpart of an expected value is the PHD

$$v(x|\mathbf{Z}^k) = \int \delta_{\mathbf{X}}(x) \cdot \pi(\mathbf{X}|\mathbf{Z}^k) \delta \mathbf{X}, \quad (4.28)$$

which is defined over the state space. According to (4.26), $v(x|\mathbf{Z}^k)$ can be interpreted as the density of objects at location x in the state space. The huge advantage of the PHD is that it no longer depends on the multi-object state, so it can be predicted and updated without the computationally costly set integrals in (4.16) and (4.17). Albeit, using not the multi-object density but only its first-order moment is accompanied with a loss of information and of the possibility to model dependencies of objects.

In [Mah03] this issue is illustrated by a suitable example for a PHD of a multi-object distribution of two one dimensional objects located at position a and b . The multi-object distribution is given by

$$\pi(\{x_1, x_2\}) = \mathcal{N}(x_1; a, \sigma^2) \cdot \mathcal{N}(x_2; b, \sigma^2) + \mathcal{N}(x_1; b, \sigma^2) \cdot \mathcal{N}(x_2; a, \sigma^2), \quad (4.29)$$

supposing that $\pi(\mathbf{X}) = 0$ unless $\mathbf{X} = \{x_1, x_2\}$ with $x_1 \neq x_2$. As introduced in

(3.6), $\mathcal{N}(x_1; a, \sigma^2)$ is a Gaussian distribution with mean value a and variance σ^2 . In [Mah03] the corresponding PHD is shown to be

$$v(x) = \mathcal{N}(x; x_1, \sigma^2) + \mathcal{N}(x; x_2, \sigma^2), \quad (4.30)$$

which is depicted by Figure 4.7a for $a = 1$, $b = 4$, and $\sigma = 1$. The two maxima

Figure 4.7: Graph comparison of the PHD $v(x)$ (4.30) with different variances σ .

Figure 4.8: Graph comparison of the multi-object density $\pi(\{x_1, x_2\})$ (4.29) with different variances σ .

of the PHD at $x = 1$ and $x = 4$ are the locations with the highest object density which mark the expected location of the objects in state space. Furthermore, the

expected number of objects is $\int v(x)dx = 2$. However, assuming a higher variance $\sigma = 1.6$, the PHD of (4.30) turns into a unimodal function over the state space which is shown in Figure 4.7b. Here the maximum is at $x = 2.5 = (a + b)/2$ which equals the center between a and b . The multi-object density $\pi(\{x_1, x_2\})$ on the other hand is able to distinguish two targets with a variance of $\sigma = 1$ as well as $\sigma = 1.6$, which is depicted by Figures 4.8a and 4.8b. Comparing the resolution capacity of $v(x)$ and $\pi(\{x_1, x_2\})$ in terms of two objects, leads to the interval $\sqrt{2}\sigma < |a - b| < 2\sigma$ in which the multi-object density is able to separate two objects whereas the PHD is not. Consequently, it is harder to track closely spaced objects which feature a high uncertainty with a PHD filter, than using a filter considering the full multi-object densities. Thus, the PHD filter introduced in the next section is suitable for systems with a low measurement noise and clearly separated objects.

4.3.2 The PHD Filter

The probability hypothesis density (PHD) filter prediction is based on the standard multi-object motion model without spawning (see Section 4.2.1) which contains the following assumptions:

- each individual object moves independently according to the single-object Markov transition density $f(x_+|x)$.
- an existing object at time step k with state x will exist at the time step of the next measurement with the surviving probability $p_S(x)$.
- the PHD for new objects entering the scene is given by $b(x)$, the PHD of the birth RFS is assumed to be statistically independent of existing objects.

Considering these assumptions, the prediction step of the PHD filter yields

$$v_+(x) = \int p_S(\xi) f(x|\xi) v(\xi) d\xi + b(x) \quad (4.31)$$

$$v_+(x) \stackrel{\text{abbr.}}{=} v_+(x|\mathbf{Z}^k) \quad (4.32)$$

$$v(x) \stackrel{\text{abbr.}}{=} v(x|\mathbf{Z}^k) \quad (4.33)$$

with the first term in (4.31) performing the prediction of the existing multi-object PHD by integrating over the state space and the second term adding the PHD for new born objects.

For the update of the PHD filter the standard measurement model (Section 4.2.1) is taken as basis:

- each object generates either a single, state independent measurement which is detected with the probability of detection $p_D(x)$ or no measurement which results in a missed detection and the corresponding probability $1 - p_D(x)$. Thus, the set of measurements generated by one object is represented by a multi-Bernoulli distributed RFS as defined in Section 4.1.1.
- the single-object likelihood function $g(z|x)$ models the likelihood that a measurement z is generated by an object with the state x .
- the density of the false alarm or clutter RFS follows a Poisson distribution with the PHD $\kappa(z) = \lambda_c c(z)$ consisting of an average number of λ_c false alarms whose spatial distribution is given by the probability density $c(z)$. Furthermore, clutter is independent of object generated measurements.

In addition to the assumptions made in the standard measurement model, the predicted multi-object distribution $\pi_+(\mathbf{x}|\mathbf{Z}^k)$ is assumed to be Poisson distributed. The update equation for the PHD filter follows from the foregoing:

$$\begin{aligned} v(x) &= L(\mathbf{Z}|x)v_+(x) \\ &= (1 - p_D(x))v_+(x) + \sum_{z \in \mathbf{Z}} \frac{p_D(x)g(z|x)v_+(x)}{\kappa(z) + \int p_D(\xi)g(z|\xi)v_+(\xi)d\xi}. \end{aligned} \quad (4.34)$$

Here the first term expresses the case of missed detections and the second term corrects the predicted PHD $v_+(x)$ with the current set of measurements $\mathbf{Z}$. Compared to the multi-object Bayes filter, the equations of the PHD filter include no set integrals which reduces the computational complexity to $\mathcal{O}(mn)$, where m is the number of measurements and n corresponds to the number of objects.

4.3.3 The Gaussian-Mixture PHD Filter

The Gaussian mixture probability hypothesis density (GM-PHD) filter is one possibility to realize a computationally tractable implementation of the PHD filter. It is based on the work on Gaussian sum filters [AS72; SA71] and introduced by [VM06]. Since the GM-PHD filter in [VM06] does not ensure the continuity of individual object tracks, it is extended by [CPV06] to include track labels. The convergence analysis of the filter is given in [CV07].

In the GM-PHD filter the posterior and the prior PHD are approximated by the

Gaussian mixtures (GMs):

$$v(x) = \sum_{i=1}^J w^{(i)} \mathcal{N}(x; \bar{x}^{(i)}, \underline{P}^{(i)}) \quad (4.35)$$

$$v_+(x) = \sum_{i=1}^{J_+} w_+^{(i)} \mathcal{N}(x; \bar{x}_+^{(i)}, \underline{P}_+^{(i)}) \quad (4.36)$$

Each mixture is composed of a weighted sum of J Gaussians with the weights $w^{(i)}$. Due to property (4.26), the expected number of objects is

$$\hat{N} = \sum_{i=1}^J w^{(i)} \text{ and } \hat{N}_+ = \sum_{i=1}^{J_+} w_+^{(i)}. \quad (4.37)$$

In addition to the requirements of the standard motion (Section 4.2.2) and measurement model (Section 4.2.1), the closed-form solution of the PHD recursion requires a linear Gaussian multi-object model which includes the following assumptions on the birth, termination, and detection of objects.

- each object follows a linear Gaussian process and measurement model:

$$f(x_+|x) = \mathcal{N}(x; \underline{F}x, \underline{Q}) \quad (4.38)$$

$$g(z|x_+) = \mathcal{N}(z; \underline{H}x_+, \underline{R}) \quad (4.39)$$

- the survival and detection probability are state independent. As described in Section 4.3.5 this assumption can be relaxed which leads to MC specific probabilities:

$$p_S(x) = p_S^{(i)} \quad (4.40)$$

$$p_D(x) = p_D^{(i)}. \quad (4.41)$$

- the PHD of new born objects is a GM:

$$b(x) = \sum_{i=1}^{J_B} w_B^{(i)} \mathcal{N}(x; \bar{x}_B^{(i)}, \underline{P}_B^{(i)}) \quad (4.42)$$

$$\hat{N}_B = \sum_{i=1}^{J_B} w_B^{(i)} \quad (4.43)$$

- each mixture component (MC) is assigned with an unique identifier or label l

to enable object tracking.

The first assumption is very common in object tracking. Similar to the Kalman filter, this requirement ensures the maintenance of the Gaussian distribution through prediction and measurement update of the GMs. The state independence of the survival and detection probability posted in the second assumption is needed to realize a closed-form solution of the PHD filter with GMs. It is reasoned in Section 4.3.5 where a relaxation of this requirement is given as well. In the second last assumption $\bar{x}_B^{(i)}$ with $i = 1, \dots, J_B$ are the peaks of the birth PHD $b(x)$, where the weights specify the expected number $\hat{N}_B$ of new born objects originated from $\bar{x}_B^{(i)}$ and $\underline{P}_B^{(i)}$ determines the spread of the PHD around $\bar{x}_B^{(i)}$. Some birth models are introduced in Section 4.3.6.

In consequence, the i^{th} MC consists of a weighted and labeled Gaussian distribution

$$\left(l^{(i)}, w^{(i)}, \mathcal{N}(x, \bar{x}^{(i)}, \underline{P}^{(i)}) \right) \quad (4.44)$$

and is predicted and updated according to the following rules.

GM-PHD Filter Prediction

The predicted PHD is a GM composed of the persisting objects $v_S(x)$ from the current time step and the new born objects $b(x)$:

$$v_+(x) = v_S(x) + b(x) \quad (4.45)$$

Both, the birth PHD $b(x)$ (4.42) as well as the PHD of the persisting objects $v_S(x)$ are GMs with mixture components (MCs) of the form (4.44). $b(x)$ creates J_B new MCs according to the given birth PHD. The labels $l \in (1, \dots, J_B)$ of the new MCs are required to be new and distinct. All other MCs of the predicted PHD $v_+(x)$ originate from persisting objects modeled by the PHD

$$v_S(x) = \sum_{i=1}^J w_+^{(i)} \mathcal{N}(x; \bar{x}_+^{(i)}, \underline{P}_+^{(i)}) \quad (4.46)$$

$$w_+^{(i)} = p_S^{(i)} w^{(i)} \quad (4.47)$$

$$\bar{x}_+^{(i)} = \underline{F} \bar{x}^{(i)} \quad (4.48)$$

$$\underline{P}_+^{(i)} = \underline{F} \underline{P}^{(i)} \underline{F}^T + \underline{Q} \quad (4.49)$$

$$l_+^{(i)} = l^{(i)}. \quad (4.50)$$

It consists of the predicted MCs of the posterior PHD, thus it has the same number J of MCs. Multiplying the prior weight $w^{(i)}$ with the survival probability $p_S^{(i)}$ facilitates the persistence or termination of MCs. As aforementioned, the survival probability has to be state independent to realize a closed-form solution of the PHD filter. Due to the linear Gaussian assumption, the prediction of the mean $\bar{x}^{(i)}$ and the covariance $\underline{P}^{(i)}$ is performed according to the Kalman filter equations (3.9) and (3.10) with the process matrix $\underline{F}$ and the process noise covariance matrix $\underline{Q}$. The labels of the MC do not change during prediction. Thus, the predicted GM-PHD (4.36) concatenates the predicted MCs $v_S(x)$ of the previous time step with new MCs $b(x)$ representing the new born objects and it consists of

$$J_+ = J + J_B \quad (4.51)$$

MCs.

GM-PHD Filter Innovation

Based on a new measurement set $\mathbf{Z} = \{z^1, \dots, z^m\}$, the predicted PHD $v_+(x)$ which is represented by J_+ MCs

$$\left(l_+^{(i)}, w_+^{(i)}, \mathcal{N}(x; \bar{x}_+^{(i)}, \underline{P}_+^{(i)}) \right) \quad (4.52)$$

is updated. The updated PHD is again a GM and results from (4.34) to be

$$v(x) = \sum_{i=1}^{J_+} (1 - p_D^{(i)}) w_+^{(i)} \mathcal{N}(x; \bar{x}_+^{(i)}, \underline{P}_+^{(i)}) + \sum_{z \in \mathbf{Z}} \sum_{i=1}^{J_+} w^{(i)}(z) \mathcal{N}(x; \bar{x}^{(i)}, \underline{P}^{(i)}). \quad (4.53)$$

The deviation is given in [VM06]. The first term of (4.53) handles the case of missed detections. Here the components of the predicted PHD $v_+(x)$ are multiplied with their probability of missed detection $q_D^{(i)} = (1 - p_D^{(i)})$ which depends on the i^{th} MC. Thus, also the detection probability is not completely state independent. Details demonstrating the correctness of this approach are given in Section 4.3.5. To realize a continuous object tracking, the labels of the MCs representing not detected objects are maintained, i.e. $l^{(i)} = l_+^{(i)}$.

The update of the predicted GM-PHD is realized with the second term in (4.53). Each measurement z_j with $j = (1, \dots, m)$ generates J_+ new MCs representing the possibility that the j^{th} measurement has been created by any of the persisting objects. The likelihood that the j^{th} measurement originates from an object in the vicinity of the i^{th} MC is represented by the measurement likelihood $q(z^{(j)} | x_+^{(i)})$ which increases

or decreases the weight $w^{(i)}$ of the MC:

$$q(z^{(j)}|x_+^{(i)}) = \mathcal{N}\left(z^{(j)}; \hat{z}^{(i)}, S^{(i)}\right) \quad (4.54)$$

$$w^{(i)}(z^{(j)}) = \frac{p_D^{(i)} w_+^{(i)} q(z^{(j)}|x_+^{(i)})}{\kappa(z) + \sum_{i=1}^{J_+} p_D^{(i)} w_+^{(i)} q(z^{(j)}|x_+^{(i)})} \quad (4.55)$$

In addition to the measurement likelihood, the detection probability $p_D^{(i)}$ of the i^{th} MC influences the weight of the updated component. For normalization purposes, the weights of all MCs generated by the same measurement $z^{(j)}$ are divided by the sum of their updated weights. To incorporate the Poisson distributed clutter, the PHD $\kappa(z)$ of the clutter RFS is part of the normalization term. Thus, an averaged update is realized which facilitates an association-free multi-object tracker.

To illustrate the update of the MCs, the following example depicted by Figure 4.9 is considered. The GM-PHD consists of two one dimensional Gaussians which are located at $x_1 = 0$ m and $x_2 = 4$ m with a variance of $\sigma_x^2 = 0.3$ m² and weight $w = 0.5$, respectively. It is updated by a measurement with variance $\sigma_z^2 = 0.3$ m² whose mean value moves between $z = 0$ m and $z = 4$ m (Figure 4.9a). In Figure 4.9b the weights of the updated Gaussians assuming a detection probability $p_D = 0.9$ and variable clutter density of $\kappa = 0, 0.01, 0.02$ are shown. Focusing on the weights for a clutter density of $\kappa = 0$, the weight of the Gaussian which matches the received measurement raises to one, while the weight of the other measurement is reduced to zero. If the measurement can neither be clearly associated to MC1 nor to MC2 the MC closer to the measurement gets the higher weight. In case of $z = 2$ m where the measurement is directly between the two Gaussians, both associations are equally likely, which is expressed by $w_1 = w_2 = 0.5$. In applications with clutter, the probability that a measurement is not originated by an object raises, which results in a lower value for the updated weight even if the measurement exactly matches the Gaussian. This is illustrated by Figure 4.9b with different κ values.

The mean and covariance of the predicted Gaussians (4.52) are determined by the update equations of the Kalman filter (see Section 3.2):

$$\bar{x}^{(i)} = \bar{x}_+^{(i)} + \underline{K}^{(i)} \left[z^{(j)} - \hat{z}^{(i)} \right] \quad (4.56)$$

$$\underline{P}^{(i)} = \underline{P}_+^{(i)} - \underline{K}^{(i)} \underline{S}^{(i)} \underline{K}^{(i)T}. \quad (4.57)$$

The predicted measurement $\hat{z}^{(i)}$, the innovation covariance $\underline{S}^{(i)}$, as well as the Kalman gain $\underline{K}^{(i)}$ are calculated according to (3.11), (3.12), and (3.14). Representing still the same objects, the labels of the updated MCs are identical to those of the predicted MCs.

- (a) Two one dimensional Gaussians of a GM-PHD with mean $x_1 = 0 \text{ m}$ and $x_2 = 4 \text{ m}$ illustrated by blue and green squares. The variance $\sigma_{x_1}^2 = \sigma_{x_2}^2 = 0.3 \text{ m}^2$ is sketched by the blue and green $3\sigma_x$ region respectively. Black crosses mark the evaluated means of the measurement.

- (b) Values of the two updated Gaussians' weights (blue and green) assuming different clutter densities κ .

Figure 4.9: Example to visualize the association-free update of the GM-PHD filter.

Thus, after the innovation of the GM-PHD the GM consists of

$$J = J_+ + m \cdot J_+ \quad (4.58)$$

MCs representing the estimated multi-object state. Here J_+ is the number of MCs of the predicted PHD and m indicates the number of measurements. Finally, the estimates for the states of the tracked objects have to be extracted out of the estimated GM-PHD $v(x)$. As the label of an MC indicates the object which is represented by this MC, all MCs are grouped with respect to their label. Further, each group of MCs contributes its MC with the highest weight to the set of object tracks if its weight is at least 0.5 [PCV09]. Consequently, each element of the set of tracks features a state estimate with related covariances as well as an unique label that facilitates a continuous multi-object tracking.

4.3.4 GM-PHD Filter Pruning and Merging

To truncate the boundless increasing number of MCs, a track management as in [CPV06; PCV09] including pruning less important and merging similar MCs is applied. The importance of MCs is expressed by their weight which is forced to exceed a pruning threshold τ_{prune} . Thus, all MCs with

$$w^{(i)} < \tau_{prune} \quad (4.59)$$

are eliminated from the GM which reduces the number of MCs from J to

$$\check{J} = J - J_{prune}.$$

Subsequently, the weights of the remaining MCs are normalized, so that the weights of the pruned GM-PHD $\check{v}(x)$ still sum to the number of objects in state space:

$$\check{w}^{(i)} = \frac{\sum_{j=1}^J w^{(j)}}{\sum_{j=1}^{\check{J}} w^{(j)}} \cdot w^{(i)}, i = 1, \dots, \check{J}. \quad (4.60)$$

In the following, the $\check{\cdot}$ over the remaining MCs can be neglected, since all other components have been eliminated.

Another way to reduce the number of the GM-PHD's components is to merge similar Gaussians. The similarity of two Gaussians is measured using the Mahalanobis distance (MHD) d_{MHD} (3.35). All MCs whose Gaussians have an MHD lower than a threshold τ_{merge} are merged as described in Algorithm 1.

Algorithm 1 Merge mixture components

Require: set $I = \{i = 1, \dots, J\}$ of MCs sorted with descending weights and a merging threshold τ_{merge} .

```

1:  $J_{merge} = 0$ 
2: while  $I \neq \emptyset$  do
3:    $J_{merge} = J_{merge} + 1$ 
4:    $L = \{i \in I \setminus 1 | (\bar{x}^{(i)} - \bar{x}^{(1)})^T \cdot (\underline{P}^{(1)} + \underline{P}^{(i)})^{-1} \cdot (\bar{x}^{(1)} - \bar{x}^{(i)}) < \tau_{merge}\}$ 
5:    $w^{(1)} = \sum_{i \in L} w^{(i)}$ 
6:    $\bar{x}^{(1)} = \frac{1}{w^{(1)}} \sum_{i \in L} w^{(i)} \bar{x}^{(i)}$ 
7:    $\underline{P}^{(1)} = \frac{1}{w^{(1)}} \sum_{i \in L} w^{(i)} (\underline{P}^{(i)} + (\bar{x}^{(1)} - \bar{x}^{(i)})(\bar{x}^{(1)} - \bar{x}^{(i)})^T)$ 
8:    $I = I \setminus 1$ 
9:    $I = I \setminus L$ 
10: end while
11: return set of MCs  $(l^{(i)}, w^{(i)}, \mathcal{N}(x, \bar{x}^{(i)}, \underline{P}^{(i)}))$  with  $i = (1, \dots, J_{merge})$ .
```

Of course it would be intuitive to merge similar MCs before pruning instead vice versa. However, since pruning reduces the number of components of the GM-PHD, less distances have to be calculated in the merging step.

4.3.5 State Dependent Detection and Survival Probability

As already noted, the GM implementation of the PHD filter necessitates a constant detection and survival probability [VM06]. The reason for this requirement is outlined in [Mah07a, p. 625]. In the PHD innovation (4.34) both $p_D(x)$ and $1 - p_D(x)$ are used. Modeling $p_D(x) = \mathcal{N}_{norm}(x; \bar{x}, \sigma)$ as a normalized Gaussian, $1 - p_D(x)$ is no more Gaussian which would violate the linearity assumption of the GM-PHD filter. Beside a generalization of the probabilities using exponential mixtures proposed in [VM06], an approximation first introduced by [UEW07] is feasible. Here an MC dependent probability of detection $p_D(x) = p_D^{(i)}$ is used to realize a GM-PHD filter with state dependent detection probability. Additionally, assuming a constant $p_D(x)$ in the uncertainty region of a Gaussian component,

$$p_D^{(i)} = p_D(\bar{x}^{(i)}) \approx \int p_D(x) \mathcal{N}(x; \bar{x}^{(i)}, \underline{P}^{(i)}) dx \quad (4.61)$$

can be computed easily. Similar, an MC dependent survival probability $p_S^{(i)} = p_S(\bar{x}^{(i)})$ can be realized. Despite, a PHD/CPHD filter for unknown detection profiles was recently developed by Mahler [MVV11], which regards the $p_D(x)$ as an unknown state variable, the aforementioned assumption is beneficial for some applications.

One case is a static sensor setup with time invariant detection profile which is used in this thesis and described in Chapter 7. In this case the detection profile can be determined accurately based on the object detection results of previously recorded sequences.

4.3.6 Birth models

In the original formulation of the PHD filter the birth PHD is assumed to be known beforehand. Thus, any available information about possible locations for object appearance is used to construct the birth GM-PHD

$$b(x) = \sum_{i=1}^{J_B} w_B^{(i)} \mathcal{N}(x; \bar{x}_B^{(i)}, \underline{P}_B^{(i)}) \quad (4.62)$$

already introduced with (4.42). An example given in [VM06] is airplane tracking where the Gaussians of $b(x)$ are concentrated in the area around airports. However, if an object appears at a region in state space not covered by the concentrated MCs in $b(x)$ the GM-PHD filter will not be able to track this object. Further, lost tracks can not be initialized again. Another way to construct $b(x)$ is to diffuse it over the entire state space [HL10; JDSW12]. This birth model facilitates the tracking of objects appearing at arbitrary locations in state space. Albeit, as indicated in [RCVV12] it requires a huge number of MCs, tends to set up spurious false tracks, and needs longer confirmation times.

Thus, an adaptive birth model similar to [RCVV12] is used in this thesis. Each measurement with a maximum measurement likelihood $\max_{i \in 1:|\mathbf{X}|} \left(q(z^{(j)} | x_+^{(i)}) \right)$ (4.54) less than a birth threshold τ_{birth} contributes a Gaussian to $b(x)$ in the subsequent prediction step. Consequently, the adaptive birth GM-PHD is given by:

$$b(x) = \sum_j w_B^{(j)} \mathcal{N}(x; \underline{H}z^{(j)}, \underline{P}_B^{(j)}), \quad \forall j \mid \max_{i \in 1:|\mathbf{X}|} \left(q(z^{(j)} | x_+^{(i)}) \right) < \tau_{birth}, \quad (4.63)$$

where $w_B^{(j)}$, $\underline{P}_B^{(j)}$, and τ_{birth} are application specific parameters and the indices are $i \in \{1, \dots, |\mathbf{X}|\}$ and $j \in \{1, \dots, |\mathbf{Z}|\}$.

4.4 The Gaussian Mixture Multiple-Model PHD Filter

The prediction step of the discussed PHD filter depends on a single a priori motion model so far. Predicting the motion of each object with the same model is not sufficient to track maneuvering objects or objects with different motion characteristics. Thus, similar to the IMM filter of Section 3.2.4, jump Markov system (JMS) based multiple-model (MM) filters have been developed to address this issue. In [PVTM09] a closed form solution for the PHD filter using a linear Gaussian jump Markov multiple-object model and a GM is introduced. This multiple-model probability hypothesis density (MMPHD) filter, which has been shown by Mahler [Mah12] to be the only mathematically correct MMPHD implementation, facilitates the tracking of objects of different types using appropriate motion models within a single filter.

4.4.1 The Jump Markov System

A jump Markov system (JMS) can be described by a finite set of N_o state space models which are parametrized by an additional discrete state variable $o = 1, \dots, N_o$ denoted as the mode or jump variable.

This leads to an augmented state $\hat{x}^o = (x, o)$, which has already been introduced for the IMM Kalman filter in Section 3.2.4. According to Bayes' rule, the prediction of the augmented state can be factorized:

$$f(x_+, o_+ | x, o) = t_{o_+, o} \cdot f(x_+ | x, o) \quad (4.64)$$

The Markov transition density of a JMS consists of the product of the Markov transition density $f(x_+ | x, o)$ corresponding to the mode variable o and the respective element $t_{o_+, o}$ of the Markov transition matrix

$$T = \begin{pmatrix} t_{o_1, o_1} & \dots & t_{o_1, o_{N_o}} \\ \vdots & \ddots & \vdots \\ t_{o_{N_o}, o_1} & \dots & t_{o_{N_o}, o_{N_o}} \end{pmatrix} \quad (4.65)$$

for the Markov chain defined by the discrete state variable o . Compared to the IMM filter (Section 3.2.4), this is an equivalent but more general formulation of an MM system.

Given this, [Mah12] defines the multi-object state of a JMS as a finite set of augmented

states:

$$\overset{o}{\mathbf{X}} = \left\{ \overset{o}{x}_1, \dots, \overset{o}{x}_n \right\} = \{(x_1, o_1), \dots, (x_n, o_n)\} \quad (4.66)$$

Consequently, each object is subject to an individual motion model mode o but with the addition that the state spaces of all objects are the same. Using the motion model o as an auxiliary variable to improve the accuracy of the object tracking, it is not estimated explicitly.

4.4.2 The GM-MMPHD Filter

Assuming linear Gaussian measurement and motion models, the augmented state can be integrated in the GM-PHD filter of Section 4.3.3 by an expansion of the mixture components (MCs) (4.44) by its individual motion model mode o :

$$\left(l^{(i)}, o^{(i)}, w^{(i)}, \mathcal{N}(x, \bar{x}^{(i)}, \underline{P}^{(i)}) \right) \quad (4.67)$$

and an adaption of the models in the following way:

- the motion model depends on the model mode

$$f(\overset{o}{x}_+ | \overset{o}{x}) = t_{o_+, o} \cdot \mathcal{N}(x; \underline{F}(o)\bar{x}(o), \underline{Q}(o)) \quad (4.68)$$

with the motion model dependent transition probability $t_{o_+, o}$, the process matrix $\underline{F}(o)$, and the process noise covariance matrix $\underline{Q}(o)$.

- the measurement model is mode independent

$$g(z | \overset{o}{x}_+) = \mathcal{N}(z; \underline{H}\bar{x}_+, \underline{R}) \quad (4.69)$$

- the detection and survival probability as well as the birth PHD are motion mode dependent

$$p_S(\overset{o}{x}) = p_S^{(i)}(o) = p_S^{(i)}, \quad (4.70)$$

$$p_D(\overset{o}{x}) = p_D^{(i)}(o) = p_D^{(i)}, \quad (4.71)$$

$$b(\overset{o}{x}) = \sum_{i=1}^{J_B} w_B^{(i)} \mathcal{N}(x; \bar{x}_B^{(i)}(o), \underline{P}_B^{(i)}(o)) = \sum_{i=1}^{J_B} w_B^{(i)} \mathcal{N}(x; \bar{x}_B^{(i)}, \underline{P}_B^{(i)}). \quad (4.72)$$

Since the motion model mode is part of the MC (4.67), the dependency on o is dropped in the following.

- the Poisson clutter process is assumed to be independent of the motion mode. In [VVC08] it was outlined that the innovation step involves a combinatorial sum in case of state dependent clutter. Thus, to realize a computationally inexpensive MMPHD filter, no motion mode dependency is considered.

Based on these considerations, the Gaussian mixture multiple-model probability hypothesis density (GM-MMPHD) filter is an ordinary GM-PHD filter defined on the augmented state space $\overset{o}{x}$. Thus, the GM-MMPHD filter equations can be obtained by substituting the kinematic states x by the augmented states $\overset{o}{x}$ in the GM-PHD equations introduced in Sections 4.3.3 and 4.3.3.

Compared to the prediction of the single model GM-PHD filter, the GM-MMPHD filter incorporates multiple state space models. Consequently, the values of the predicted PHD (4.46) depends on the motion mode o in the MM case:

$$\begin{aligned} v_S(\overset{o}{x}) &= v_S(x, o) \\ &= \sum_o \sum_{i=1}^{J(o)} w_+^{(i)} \mathcal{N}(x; \bar{x}_+^{(i)}, \underline{P}_+^{(i)}), \end{aligned} \quad (4.73)$$

$$\bar{x}_+^{(i)} = \underline{F}(o_+) \bar{x}^{(i)}, \quad (4.74)$$

$$\underline{P}_+^{(i)} = \underline{F}(o_+) \underline{P}^{(i)} \underline{F}^T(o_+) + \underline{Q}(o_+), \quad (4.75)$$

$$w_+^{(i)} = p_S^{(i)} t_{o_+, o} w^{(i)}, \quad (4.76)$$

$$l_+^{(i)} = l^{(i)}. \quad (4.77)$$

Calculating the sum over all model modes o in (4.73), each MC is predicted according to each motion model and weighted with the product of the prior weight $w^{(i)}$, its survival probability $p_S^{(i)}$, and the model transition probability $t_{o_+, o}$. As in the IMM Kalman filter, $t_{o_+, o}$ is an element of the Markovian transition probability matrix T (3.29) which is commonly assumed to be constant and equal for all MCs [BL93; PVTM09].

Concretely, previous to the prediction $N_o - 1$ duplicates of each MC are generated and transformed to the state representation of the predicted motion model $o_+ \in \{o_1, \dots, o_{N_o} \mid o_+ \neq o^{(i)}\}$. Subsequently, the MCs are predicted according to their motion model (4.73)-(4.77). Thus, MCs with different motion models but the same label are possible. This facilitates that each object can be predicted with its individual motion model which corresponds to the preferred approach in [Mah12]. Consequently, the predicted GM-PHD consists of

$$J_+ = J \cdot N_o + J_B \quad (4.78)$$

MCs. Due to the linear increase of the number of MCs with the number of motion models, an appropriate choice of the model transitions probabilities is crucial to preserve the computational tractability of the filter. Additionally, the pruning and merging methods of Section 4.3.4 can be used to reduce the number of MCs. The motion model mode of new born objects is modeled according to a probability distribution of the model modes that is multiplied with the birth PHD introduced in Section 4.3.6 [PVTM09].

The innovation for the GM-MMPHD filter is calculated on the lines of the single model GM-PHD filter discussed in Section 4.3.3. Beside the difference that now every i^{th} MC comprises its individual motion model mode, the weight update (4.79) involves a summation over the model modes:

$$w^{(i)}(z^{(j)}) = \frac{p_D^{(i)} w_+^{(i)} q(z^{(j)} | x_+^{(i)})}{\kappa(z) + \sum_o \sum_{i=1}^{J_+(o)} p_D^{(i)} w_+^{(i)} q(z^{(j)} | x_+^{(i)})}. \quad (4.79)$$

Here a single measurement model per sensor is assumed and the definition of the augmented multi-object state (4.66) requires equal object state spaces for every motion model, the measurement model does not depend on the motion model mode.

In contrast to the IMM Kalman filter (Section 3.2.4), model probabilities are not explicitly required because in the GM-MMPHD filter they are included in the weighted MCs implicitly. Further, as addressed by [Mah12] the introduced version of the MMPHD filter is until now the only theoretically funded MMPHD filter. This filter can not be denoted as an IMM filter, since no interaction between the model dependent PHDs is involved in the prediction step. In [PKS08] an interacting multiple-model PHD (IMMPPHD) filter is proposed, but it has been derived bottom-up from the single object IMM filter and not top-down from an IMMPPHD filter which makes it impossible to prove its mathematical correctness.

As aforementioned, the motion model mode o is only an auxiliary variable to improve the state estimation accuracy. Therefore, the PHD is marginalized over the motion model mode o to obtain the model independent PHD

$$v(x) = \sum_o v(x^o). \quad (4.80)$$

Given this, usual GM-PHD state extraction methods as discussed in Section 4.3.3 can be employed.

4.4.3 The GM-MMPHD Filter Incorporating Different State Spaces

The definition of the augmented multi-object state (4.66) is based on the assumption that the object state space for every motion model is the same. However, in many applications this assumption does not hold because motion models defined on different state spaces are required. One example application is given in Section 3.4 where highly agile objects as well as objects which are limited in their freedom of motion have to be tracked. Hence, the augmented multi-object state is composed of a disjoint union of motion model specific multi-object states:

$$\begin{aligned}\hat{\mathbf{X}} &= \hat{\mathbf{X}}(o_1) \uplus \dots \uplus \hat{\mathbf{X}}(o_{N_o}) \\ &= \{(x_1, o_1), \dots, (x_{n_1}, o_1)\} \uplus \dots \uplus \{(x_1, o_{N_o}), \dots, (x_{n_{N_o}}, o_{N_o})\}.\end{aligned}\quad (4.81)$$

Defining a PHD on each motion model specific multi-object state and knowing the transformations between the states and covariances of the state models, a GM-MMPHD filter as outlined in Section 4.4.2 can be used to estimate the states of the objects. The transformation between the state models is involved in the prediction step (4.73), where each MC is predicted with every motion model and in the merging step where similar MCs are merged.

Motion models with different state spaces additionally require motion mode specific measurements models $\underline{H}(o) = \underline{H}^{(i)}$. This directly affects the prediction of the measurement (3.11), the innovation covariance (3.12), and the Kalman gain (3.14).

Using the modelling introduced in this section, the GM-MMPHD filter is able to track multiple objects which follow motion models defined on different state spaces. This is a crucial feature of the filter for the application described in Chapter 7.

Chapter 5

Multi-Sensor Classifying MMPHD Filter

Figure 5.1: Overview on the proposed perception system.

A common issue in multi-object tracking is that the objects differ in their type, their appearance as well as their motion characteristics. However, these attributes have not been considered in this thesis so far. Depending on the application of the multi-object tracker, a differentiation of the objects based on their type may be required. One use-case is path planning for mobile robots which requires a classification of static and moving objects to consider all feasible paths to reach a destination. Another application is the tracking of objects with different motion characteristics, like pedestrians and vehicles. In this particular case the class information can be exploited to predict pedestrians with a motion model which is able to describe fast

direction changes including even lateral movements while a more rigid single-track model is used for vehicles. Thus, the classifying multiple-model probability hypothesis density (CMMPHD) filter is proposed to realize an integrated measurement and track-feature-based object classification which subsequently enables the adaption of the tracks' motion model according to their estimated object class. Further, the CMMPHD filter facilitates the fusion of multiple classifying or non-classifying sensors.

An overview of the entire filter is sketched in Figure 5.1. Starting at the top of Figure 5.1, the developed CMMPHD filter supports the fusion of measurement sets generated by arbitrary sensors or object detectors as long as a set element can be represented as an oriented bounding box (OBB) and related uncertainties are provided. The fusion of multiple sensors differing in their FOV and their recognition abilities is discussed in Section 5.3. The CMMPHD filter estimating the number of objects in the scene as well as their states and classes is depicted at the bottom of Figure 5.1 and introduced in Section 5.2. Due to the MM capability of the filter and the integrated class estimation, the estimation of the objects' states using appropriate motion models within a single filter is possible. The benefit of this approach compared to classical filter structures is motivated in the following.

Classical approaches exploit parallel filter structures in which each filter was designed to track one object type with characteristic motion and noise models. An example setup with two filters, one for pedestrians (ped) and the other one for vehicles (car) is illustrated in Figure 5.2a. In this case the filters are absolutely independent and the measurement-to-filter assignment rests on a measurement-based object classification like the one described in Section 6.3. A serious issue of the structure in Figure 5.2a is the strong dependency on the classification result. In case of a false classification, e.g. a pedestrian is classified as vehicle, the pedestrian tracker receives no measurement, i.e. a missed detection, while the vehicle tracker receives a measurement of a pedestrian, i.e. which corresponds to a clutter measurement. This might lead to track losses and false tracks. To overcome these drawbacks, the CMMPHD filter has an integrated object classification. As depicted in Figure 5.2b, multiple objects of different classes are tracked using a single filter which supports multiple object classes.

Within the CMMPHD filter feasible object classes are modeled using the Dempster-Shafer theory of evidence (DST). The DST is a general and flexible theory to describe and fuse hypotheses of sensors which can be even contradictory or fuzzy. Subsequently, the fusion of measurements of heterogeneous sensors incorporating class information is possible. In an initial version of the CMMPHD filter the objects' classes were represented by probabilities. Since the DST representation facilitates the resolution of contradicting measurements and the set of distinguished classes can be extended conveniently, only the DST based object classification is discussed

Figure 5.2: Comparison of a parallel (a) and single MM filter structure (b).

in this thesis.

5.1 Introduction to the Dempster-Shafer Theory of Evidence

The Dempster-Shafer theory of evidence (DST) introduced by Dempster [Dem68] and Shafer [Sha76] is a convenient and flexible way to represent uncertain data. Assuming an object O and an attribute a which is related to the object, the true attribute a^* is supposed to be a member of a finite subset of disjoint elementary hypotheses a_i represented by the frame of discernment (FOD)

$$\Omega = \{a_i\}, \quad i = 1, \dots, n. \quad (5.1)$$

In many applications it is not possible to determine the true attribute based on a single statement of a totally reliable information source. Moreover, in general multiple uncertain statements of at least one information source S_j with a characteristic reliability have to be fused to determine a^* . Each source provides its support for so-called focal hypotheses A_j which can be composed of any subset of Ω i.e. an arbitrary element of the power set 2^Ω . In DST the certainty or amount of belief for a focal hypothesis is expressed by a basic belief mass (BBM) denoted by $m(A_j) \geq 0$. The BBM may be a probability but does not have to be a probability necessarily.

The basic belief assignment (BBA) m summarizes the knowledge of a source about the object's attribute and is generally defined as:

$$m(\emptyset) = 0, \quad (5.2)$$

$$\sum_{A \subseteq \Omega} m(A) = 1 \quad (5.3)$$

which performs a mapping of the power set 2^Ω to the interval $[0, 1]$. Postulating the empty BBA $m(\emptyset) = 0$, leads to the closed-world assumption [Sme90] which states that the true attribute of the object is an elementary hypothesis contained in Ω . This is an appropriate assumption for the aim of this thesis, due to the known environment.

In contrast to probability theoretic approaches, the DST facilitates the expression of the belief in every element of 2^Ω and not only the elementary hypotheses. Thus, even information sources can be used which are only able to provide their knowledge about subsets of Ω .

Additionally the reliability of each information source can be modeled using discounting:

$$m^\alpha(A) = \begin{cases} \alpha m(A), & A \neq \emptyset \\ 1 - \alpha + \alpha m(\Omega), & A = \Omega \end{cases} \quad (5.4)$$

The parameter α expresses the reliability of the BBA and can be interpreted as a probability of correctness. Thus, it is feasible to fuse two BBAs received from sources S_1 and S_2 with different reliability using Dempster's rule of combination

$$\begin{aligned} m_{S_1 \oplus S_2}(A) &= m_{S_1}(A) \oplus m_{S_2}(A) \\ &= \frac{\sum_{X \cap Y = A} m_{S_1}(X) m_{S_2}(Y)}{1 - \sum_{X \cap Y = \emptyset} m_{S_1}(X) m_{S_2}(Y)} \quad \forall A \in 2^\Omega, \end{aligned} \quad (5.5)$$

by simply discounting the BBAs with their probability of correctness prior to fusion. In the nominator of (5.5) the sum over all combinations of $m_{S_1}(X) m_{S_2}(Y)$ which have compatible focal hypotheses is calculated. Discounting the BBAs prior to fusion, guarantees that $m(\Omega) > 0$ which effects a preservation of all focal hypotheses of the fused BBAs in the resulting BBA. The denominator normalizes $m_{S_1 \oplus S_2}(A)$ to compensate partly contradicting BBAs which means that in m_{S_1} exists at least one proposition X with $m_{S_1}(X) > 0$ which is not compatible to any proposition in m_{S_2} . Discounting is moreover used to generate a low pass filter effect by applying it to BBAs of the same source but of consecutive points in time.

A special case is the categorical BBA with $m_I(\Omega) = 1$ which is the identity in case of Dempster's rule of combination:

$$m_I(A) \oplus m(A) = m(A) \forall A \in 2^\Omega. \quad (5.6)$$

Furthermore, with $m(\Omega) > 0$ supporting each focal element of Ω , the “Zadeh's paradox” [Zad79] is resolved as proposed in [Mah07a, Sec. 4.5.2]. The “Zadeh's paradox” has been the reason for numerous critics regarding a deficiency of Dempster's rule of combination in handling contradicting BBAs. Therefore, it is reviewed in the following and a resolution of it using discounting is given. Assume two doctors, Dr. D_A and Dr. D_B , both independently examine the same patient and make their diagnosis whether the patient suffers from meningitis μ , a brain tumor τ , or concussion χ , i.e. $\Omega = \{\mu, \tau, \chi\}$. Their diagnosis represented by BBAs is given in the rows of Table 5.1. All remaining BBMs of Ω not listed in Table 5.1 are equal zero. Although both doctors have a very low support for the brain tumor, the result of Dempster's rule of combination states that it is definitely a brain tumor. This counter-example was mainly accepted as a prove for the failure of Dempster's rule of combination in case of contradicting BBAs. Nevertheless, as Mahler points out in [Mah07a, Sec. 4.5.2], the result of Dempster's rule of combination is equivalent to the result of Bayes' rule. It can be applied in the aforementioned example, since the BBAs of D_A and of D_B only contain BBMs on elementary hypotheses. Thus, an interpretation of the BBMs as probabilities is feasible. Consequently, the failure is not in Dempster's rule of combination but in the modeling of the doctors' belief. Mahler concludes in [Mah07a, Sec. 4.5.2] that it is very unwise to assign a zero probability or support to any state. This means in terms of the doctor example that no diagnosis is totally reliable, thus a mass of zero or one to any focal hypothesis is not appropriate. In DST this can be assured using discounting just before fusion. In Table 5.2 the discounted BBAs of the doctors' diagnoses assuming a reliability of $\alpha = 0.95$ are illustrated in the first two rows. The discounting leads to an additional mass $m(\mu\tau\chi)$ on the whole frame of discernment which supports all possible combinations of μ , τ , and χ . Consequently, the fused diagnosis shown in the last row of Table 5.2 features an equivalently high support for μ and χ and just a low support for τ . This indicates a high uncertainty regarding μ and χ which meets one's expectation fusing two equally reliable diagnoses. Thus, the given brief review of Zadeh's paradox shows that fusing discounted BBAs facilitates a correct fusion of even contradicting statements.

The BBM $m(A)$ expresses only the support exactly given to A but not the total support of A in the BBA m . Therefore, the degree of belief, the degree of plausibility and the uncertainty are used to express the total support of A in m . The degree of belief $Bel_m(A)$ can be interpreted as a loose lower limit to the uncertainty interval

doctor	$m(\mu)$	$m(\tau)$	$m(\chi)$
D_A	0.99	0.01	0.0
D_B	0.0	0.01	0.99
$D_A \oplus D_B$	0.0	1.0	0.0

Table 5.1: Zadeh's paradox: the first two lines show the diagnoses of the doctors, in the third line the result of the fusion is given.

doctor	$m(\mu)$	$m(\tau)$	$m(\chi)$	$m(\mu\tau\chi)$
D_A	0.94	0.01	0.0	0.05
D_B	0.0	0.01	0.94	0.05
$D_A \oplus D_B$	0.48	0.01	0.48	0.03

Table 5.2: Zadeh's paradox using discounting: the first two lines show the discounted diagnoses of the doctors, in the third line the result of the fusion is given.

$U_m(A)$ and is defined as:

$$Bel_m(A) = \sum_{B \subseteq A, B \neq \emptyset} m(B). \quad (5.7)$$

Here all BBMs of m are added which imply A . On the contrary, the degree of plausibility $Pl_m(A)$ is a loose upper limit to the uncertainty:

$$Pl_m(A) = \sum_{B \cap A \neq \emptyset} m(B) \quad (5.8)$$

which is composed of all BBMs of m not contradicting A . Finally the uncertainty interval

$$U_m(A) = Pl_m(A) - Bel_m(A) \quad (5.9)$$

is bounded by $Bel_m(A)$ and $Pl_m(A)$. All three measures of the BBA m are depicted in Figure 5.3.

The aforementioned representations of uncertain data are all based on belief functions. In the transferable belief model (TBM) of Smets and Kennes [SK94] this is called the credal level. On this level the uncertain belief of multiple sources is gathered and fused. Furthermore, in [SK94] a pignistic level (from Latin *pignus*, a bet) on which decisions are made is introduced. Here probabilities are induced by beliefs at the credal level. Based on these probabilities decisions can be made. To obtain a probability for a focal hypothesis from the BBA, the pignistic transformation is

Figure 5.3: Belief, plausibility and uncertainty of a BBA [MRWD13].

utilized:

$$BetP_m(A) = \sum_{B \subseteq \Omega, B \neq \emptyset} \frac{|A \cap B|}{|B|} m(B) \quad (5.10)$$

The pignistic transformation $BetP_m(A)$ accounts the support of every BBM $m(B)$ which does not contradict A . In case of B not being an elementary hypothesis of Ω , there is a lack of information which portion of $m(B)$ belongs to which subset of B . Thus, the $m(B)$ is equally distributed among the elementary hypotheses of B achieved by the division with $|B|$ which is the number of elementary hypotheses in B . The multiplication with the number $|A \cap B|$ assures that A and B have a common subset of hypotheses. In this thesis the understanding of the credal level, to handle uncertain data, and the pignistic level, to make decisions is adopted from the TBM [SK94]. Any further assumptions are based on the DST.

5.2 The Classifying MMPHD Filter

Generally, two sources for object classification can be distinguished: firstly, the classification may be based on features of the measurements, like the texture in case of video detections and secondly, features of the tracks, like their velocity, may be used. Albeit, the use of measurement-based features is problematic in case of PHD filters, since no explicit association between measurements and tracks is available. Therefore, the already introduced GM-MMPHD filter is extended to the Gaussian mixture classifying multiple-model probability hypothesis density (GM-CMMPHD) filter which is able to incorporate classified measurements and track-features for track classification. To realize the aforementioned characteristics of the filter, each mixture component (MC) (4.67) is augmented to:

$$\left(l^{(i)}, o^{(i)}, w^{(i)}, \mathcal{N}(x, \bar{x}^{(i)}, \underline{P}^{(i)}), \mathcal{N}(d; \bar{d}^{(i)}, \underline{P}_d^{(i)}), m_C^{(i)} \right). \quad (5.11)$$

Beside the elements approximating the MMPHD in (4.67), each MC additionally holds object attributes which do not belong to the multi-object state. The properties

are: certainty values for possible object classes, modeled by a class-BBA $m_C^{(i)}$ and the object's dimension d which is modeled as normal distributed with mean $\bar{d}$ and covariance $\underline{P}_d$ and includes the width and length of the object. In Figure 5.4 a full filter cycle of a single CMMPHD MC is illustrated. Following (5.11), the MC consists of three components: the state, the dimension, and the class. In Figure 5.4 those components are visualized as parallel layers which are all predicted according to a model and corrected with a measurement in the update step.

The assignment of object attributes to the MCs is an approximation, since an MC can represent multiple objects with different class and dimension attributes in general. However, in many applications this approximation is suitable, due to a good separability of the tracked objects in the state space. Further, the estimation of the objects' state does still not require a measurement-to-track association.

As visualized in Figure 5.4, each MC is updated by a measurement

$$z^{(j)} = \left(\mathcal{N}(z_s; \bar{z}_s^{(j)}, \underline{R}_s^{(j)}), \mathcal{N}(z_d; \bar{z}_d^{(j)}, \underline{R}_d^{(j)}), m_{C,z}^{(j)} \right), \quad j \in \{1, \dots, |\mathbf{Z}|\}, \quad (5.12)$$

composed of the object's 3D position and orientation $\bar{z}_s$, its dimension $\bar{z}_d$ which are both Gaussian distributed with covariances $\underline{R}_s$ and $\underline{R}_d$, and its measurement-based class-BBA $m_{C,z}^{(j)}$.

As indicated by the three separate layers in Figure 5.4, the estimation of the object's state, class, and dimension do not directly influence each other. Thus, the additional properties do not manipulate the result of the PHD filter estimate and can be used to adapt filter parameters like the transition or detection probability as visualized by the dashed arrows in Figure 5.4. The upper layer (dark grey) of Figure 5.4 depicts the standard GM-MMPHD filter prediction and update of the Gaussian and its weight as described in Section 4.4.2. The remaining two layers visualize the attached object attributes: the object's class (grey) and its dimension (light grey). Both do not belong to the multi-object state estimated by the PHD filter, since they are static parameters in general.

However, depending on the viewpoint, perception systems are often not able to detect the entire object which causes variations in the detected dimensions. Thus, the distribution of the object's dimensions is approximated as Gaussian distributed random variable and estimated using a separate, MC individual Kalman filter just as in the classical multi-object tracking described in Section 3.3. Due to the fact that each MC is updated with only one measurement, no data association is required. In the prediction step a pseudo noise is added to model the possible dimension change between measurements. The measurement uncertainty is state dependent and decreases with the ability of the system to detect the entire object. The

Figure 5.4: Structure of the CMMPHD filter cycle for a single MC. On the first layer (dark grey) the standard GM-MMPHD prediction and update of the Gaussian which depends on the transition and detection probability is visualized. The next two layers show additional properties of the MC which are estimated in parallel. The class-BBA represents the class of the object. It is updated with a class-BBA $m_{C,z}$ which is contained in each measurement and BBAs formed out of features of the MC, which is indicated by the dotted arrows on the right. The estimated object dimension is one of these features which are characteristic for the object's class. The dimension estimation is depicted by the bottom layer. The dashed arrows from the middle layer to the transition and detection probability visualize the adaption of these probabilities based on the estimated class.

estimation of the object's class modeled by the class-BBA $m_C^{(i)}$ is discussed in the subsequent Section 5.2.1.

Despite the extension of the MCs, the management of the mixture itself comprising birth, termination, pruning, and merging of MCs as well as the track extraction, is applied as described in Section 4.4.

In [GW12b] a CPHD filter incorporating class information is presented. The so-called classification aided CPHD filter is based on the multiple-model cardinalized probability hypothesis density (MMCPHD) filter [GW12a] and replaces the discrete motion model mode variable by a discrete object class variable. Further, the CPHD update is extended to involve measurement-feature based class probabilities. In contrast to the CMMPHD filter, the filter in [GW12b] does not facilitate the use of class specific motion models. Recently, Vo and Vo [VV13a] proposed a formulation of the joint multi-object tracking, identification and classification problem based on the RFS framework. In their approach, labeled random finite sets are used and the kinematic state is extended by the objects' classes which facilitates the incorporation of the object classification into the multi-object estimation process directly. Thus, the approach in [VV13a] is more general than the independent and parallel estimation of the objects' states and classes proposed in this thesis. Nevertheless, the formulation of [VV13a] has not been tested on either simulated or real data and significantly exceeds the complexity of the proposed CMMPHD filter.

5.2.1 Using BBAs for Track Classification

Using the MC representation (5.11) facilitates an integrated track classification based on measurement as well as track-features in an association-free MMPHD filter. Hence, the name classifying multiple-model probability hypothesis density (CMMPHD) filter derives. Firstly, the frame of discernment (FOD) Ω_T of the tracker is defined whose focal elements correspond to the classes that have to be distinguished. Each $m_C^{(i)}$ of (5.11) is defined on Ω_T and holds the certainty for the object classes in Ω_T .

A major advantage of PHD filters is the missing explicit measurement-to-track association. Thus, computationally expensive data association methods, like the auction algorithm or PDA, are unnecessary. However, without a measurement-to-track association the measurements cannot be used directly to estimate the objects' classes. Using the augmented MC (5.11) which is updated with a single measurement, the GM-CMMPHD filter facilitates the estimation of the MCs' class attributes exploiting features of the measurements but maintaining the freedom of association. As depicted by Figure 5.4, the class is estimated in parallel to

the multi-object state and might be based on measurement information $m_{C,z}$ and characteristic features of the MC, e.g. its maximum velocity $|v_{max}|$.

In order to realize the measurement-based class estimation, the measurements are required to contain a class-BBA $m_{C,z}$ which may be obtained using features of the detected objects or class probabilities of a classifier. However, the approach additionally allows the incorporation of unclassified measurements by assigning the full mass to $m_{C,z}(\Omega_T) = 1$. Since both measurement and track-features are sensor and application specific, detailed examples will be given in the Section 7.2.

In the prediction step the uncertainty of the class-BBA is increased by discounting (5.4):

$$m_{C+}^{(i)} = \left(m_C^{(i)}\right)^{p_+} \quad (5.13)$$

The parameter p_+ expresses the validity of the estimated class-BBA at the time of the next measurement and ensures that the object classification m_C does not focus on a single class proposition.

Regardless of the source of the class-BBA, the class of the i^{th} MC is updated according to (5.5):

$$m_C^{(i)} = m_{C+}^{(i)} \oplus m_C^\alpha \quad (5.14)$$

By discounting the class-BBA m_C prior to fusion with α , the confidence in the classification result of current source can be considered. Hence, α can be dependent on the sensor which provides the class-BBA or the used track-feature.

As outlined in Section 4.3.4, similar MCs are merged to truncate the number of MCs. Assume that the set of MCs to merge is $L = \{l_1, \dots, l_n\}$. To calculate the class-BBA of the merged MC, the contributing weight

$$\tilde{w}^{(l_j)} = \frac{w^{(l_j)}}{\sum_{j \in L} w^{(j)}} \quad (5.15)$$

for each MC is determined. Subsequently, the merged BBA results from the fusion of the BBAs discounted with their contributing weight:

$$m_C^L = \left(m_C^{(l_1)}\right)^{\tilde{w}^{(l_1)}} \oplus \left(m_C^{(l_2)}\right)^{\tilde{w}^{(l_2)}} \oplus \dots \oplus \left(m_C^{(l_n)}\right)^{\tilde{w}^{(l_n)}} \quad (5.16)$$

Transforming the class-BBA to the pignistic level using (5.10), a decision regarding

the class $C^{(i)}$ of the object represented by the i^{th} MC

$$C^{(i)} = \max_{a \in \Omega_T} BetP_{m_C^{(i)}}(a) \quad (5.17)$$

and its class probability

$$p_C^{(i)} = BetP_{m_C^{(i)}}(C^{(i)}) \quad (5.18)$$

can be made.

5.2.2 Using BBAs to Adapt the Transition Matrix

Depending on the type of an object, different motion models are required to accurately predict the objects' motion. Hence, the integrated estimation of the class-BBA m_C in the CMMPHD filter can be used to control the transitions between the motion models in the prediction of the GM-MMPHD filter (4.73). In general, the elements $t_{o_+,o}$ of the motion model transition probability matrix T are constant [BL93; PVTM09], i.e. do not depend on the state or MC. Recently, several approaches to adapt the transition probabilities based on the state [RGV⁺13], map information [DMM⁺11] or lane markings [SW09] have been proposed. All of them use multiple models to model different modes of object motion. In contrast to them, the motion model transition probability matrix of each MC in the GM-CMMPHD filter is adapted with respect to the estimated class-BBA, to sufficiently model the motion characteristics according to the object's type. The dashed arrow between the class-BBA estimation and the transition probability in Figure 5.4 indicates this dependency. Thus, the MC weight prediction (4.76) modifies to

$$w_+^{(i)} = p_S^{(i)} t_{o_+,o}^{(i)} w^{(i)}, \quad (5.19)$$

where $t_{o_+,o}^{(i)}$ is an element of the MC individual transition probability matrix

$$T^{(i)} = \begin{pmatrix} BetP_{m_C^{(i)}}(A_{o_1}) & \dots & BetP_{m_C^{(i)}}(A_{o_{N_o}}) \\ \vdots & \ddots & \vdots \\ BetP_{m_C^{(i)}}(A_{o_1}) & \dots & BetP_{m_C^{(i)}}(A_{o_{N_o}}) \end{pmatrix} \quad (5.20)$$

which is recalculated with each update. The class hypotheses A_{o_j} have to cover at least one object class of the tracker's FOD Ω_T which follow the same motion model o_j ($j \in \{1, \dots, N_o\}$). Additionally, all distinguished classes have to be covered by $\{A_{o_1}, \dots, A_{o_{N_o}}\}$ with $A_{o_i} \cap A_{o_j} = \emptyset$ for $i \neq j$. Consequently, the rows of $T^{(i)}$ sum to unity, which means that one of the motion models is assumed to be appropriate.

Figure 5.5: Perception area of two sensors S_1 and S_2 with different recognition abilities.

Thus, each element of $T^{(i)}$ transforms the currently estimated class-BBA $m_C^{(i)}$ of the i^{th} MC into a probability expressing the likelihood that o_j ($j \in \{1, \dots, N_o\}$) is the appropriate motion model for the MC.

5.3 The Multi-Sensor CMMPHD Filter

Advanced perception systems are often required to redundantly perceive arbitrary objects maneuvering in a large area. Thus, multiple sensors have to be combined to detect all relevant objects. To cover the entire perception area, a network of one sensor type with complementary FOVs is popular. Further, complementary sensor types with redundant FOVs are used to facilitate the recognition of objects of different types. In Figure 5.5 an example for a sensor setup consisting of two sensors S_1 and S_2 with partially redundant FOVs is given.

Being able to fuse synchronous measurement sets of multiple sensors with different characteristics, the single-sensor CMMPHD filter of Section 5.2 is extended to the multi-sensor case. For this extension multiple concepts have been presented in literature which differ in the degree of approximation. Independent of the multi-sensor

PHD concept, the measurement sets of the sensors are assumed to be independent and the joint measurement set $\mathbf{Z}_J$ is given by the disjoint union of the sensors' measurement sets

$$\mathbf{Z}_J = \mathbf{Z}^{S_1} \uplus \dots \uplus \mathbf{Z}^{S_{N_S}}, \quad (5.21)$$

where all measurement sets have the same data acquisition time and N_S is the number of used sensors. Further, the PHD prediction in the single and multi-sensor case possess the same form. Thus, the update step has to be modified to correct the predicted PHD with the joint measurement set $\mathbf{Z}_J$.

5.3.1 Multi-Sensor PHD filters

Due to the huge variety of proposed multi-sensor PHD filters, in the following a brief summary and valuation of them is given.

The general multi-sensor PHD filter

A general form of the multi-sensor PHD filter was presented in [CM12] and [Mah09a]. In the update step, a weighted summation over all possible measurement pairings contained in $\mathbf{Z}_J$ is calculated. Thus, the general multi-sensor PHD suffers from combinatorial complexity and is computationally problematic [Mah10; NCM11].

The iterated-corrector approximation of the multi-sensor PHD filter

Hence, as Mahler poses in [Mah10], the iterated-corrector approximation has become the default approach to solve multi-sensor PHD problems. It was introduced in [Mah03] and provides an approximation of the multi-sensor PHD update. Firstly, the update step (4.34) is performed with the first measurement set to calculate the updated PHD for the first sensor. Afterwards, the measurement set of the second sensor updates that PHD instead of the predicted PHD which leads to an updated PHD involving the first and second sensor. Iterating the update for all synchronously received sensor measurements, yields to the PHD updated by multiple sensors. However, in [NC11] the limitations of the iterated-corrector approximation are demonstrated. It is shown that the value of the updated PHD significantly depends on the order in which the sensors' measurement sets are provided and the ratio between the sensors' detection probabilities p_D^S . However, in this thesis

sensors with different detection probabilities are fused which excludes the use of the iterated-corrector approximation.

The PM-PHD filter

A compromise between the computationally intractable general multi-sensor PHD filter and the approximation in the iterated-corrector PHD filter is the product multi-sensor probability hypothesis density (PM-PHD) filter [Mah10; NCM11]. This filter is a theoretically principled, order-independent, and tractable approximation of the multi-sensor PHD. In the update of the PM-PHD filter the predicted PHD is not updated by iteration but by multiplication of the single-sensor PHD update terms $L(Z^{S_j}|x)$ (4.34) with $j = 1 \dots N_S$ and a normalization term. However, this normalization term includes an infinite sum for large cardinality values. Since in road user tracking large cardinality values are usual, also the PM-PHD filter is not suited for this thesis.

The multi-sensor PHD filter approximation used in the CMMPHD filter

The extension of the CMMPHD filter to handle synchronous measurements of multiple sensors is based on a computationally tractable and order independent multi-sensor approximation first proposed in [Mah03, (105)-(107)]. Here the multi-sensor PHD update is given by:

$$v(x) = L(Z^{S_1}|x) \cdots L(Z^{S_{N_S}}|x) \cdot v_+(x), \quad (5.22)$$

where

$$L(Z^{S_j}|x) = (1 - p_D^{S_j}(x)) + \sum_{z \in \mathbf{Z}^{S_j}} \frac{p_D^{S_j}(x) g^{S_j}(z|x)}{\kappa^{S_j}(z) + \int p_D^{S_j}(\xi) g^{S_j}(z|\xi) v_+(\xi) d\xi} \quad (5.23)$$

is equivalent to the single-sensor PHD update (4.34). The multi-sensor update (5.22) equals that of the PM-PHD filter except for the computationally problematic normalization term. In difference to (4.34), the detection probability $p_D^{S_j}$, single-sensor measurement likelihood $g^{S_j}(z|x)$, as well as the clutter PHD $\kappa^{S_j}(z)$ depend on the currently processed sensor S_j .

In cases in which the number of objects is small, the approximation in (5.22) becomes invalid, since it can not be reduced to the multi-sensor form of Bayes' rule [LHLC11].

Nevertheless, the filter has been applied with success in several applications, e.g. [DDVH11; LHLC11].

Thus, in this thesis the multi-sensor update formula in (5.22) is used, since it is independent of the sensors' p_D^S ratio. That is required due to the already mentioned state and class dependent detection probabilities which are discussed in Sections 5.3.2 and 5.3.3. Consequently, a balanced ratio between the detection probabilities of the sensors can not be guaranteed.

Further, the complexity $\mathcal{O}((|\mathbf{Z}^{S_1}| + \dots + |\mathbf{Z}^{S_{N_S}}|) \cdot |\mathbf{X}|)$ of the multi-sensor PHD update is linear in the number of provided sensor measurements, thus enables a real-time implementation even in applications with a huge number of objects which is an advantage over the PM-PHD filter. Owing to multiplication of multiple single-sensor update terms $L(\mathbf{Z}^{S_j}|x)$ (5.23) in (5.22), the multi-sensor PHD filter can be implemented with GMs just as introduced in Section 4.3.3. In the following, approaches to handle redundant and complementary sensors in the GM implementation of the multi-sensor CMMPHD filter are discussed.

5.3.2 Fusing Sensors with Different FOVs

To fuse sensors with different or redundant FOVs, an accurate model of the sensors' detection performance considering their FOVs is required. In Figure 5.5 two pedestrians are in the FOV of sensor S_1 and one of the pedestrians as well as one car are in the FOV of sensor S_2 . Consequently, the car and the pedestrian on the left can only be detected by one of the sensors respectively. Providing measurements of both sensors to a multi-sensor GM-PHD filter which does not consider the sensors' FOVs, will lead to unstable tracking results. That is because a measurement set of S_2 is able to confirm the car track and raises the weights of the corresponding MCs, while a measurement set of S_1 will not contain a detection of the car which subsequently decreases the MCs' weights.

To overcome this drawback, a sensor and state dependent detection probability

$$p_D^{(i),S_j} = p_D^{S_j}(\bar{x}^{(i)}) \quad (5.24)$$

is introduced. It expresses the probability that the sensor S_j is able to detect the i^{th} MC with mean $\bar{x}^{(i)}$. Here the assumption of [UEW07] introduced in Section 4.3.5 is adopted which assumes that the detection probability of one sensor is constant within the uncertainty region of a Gaussian. Assuming for the example in Figure 5.5 that the p_D inside the sensors' FOV is close to one and outside close to zero, the contradicting object detections of S_1 and S_2 can be resolved. Thus, the influence of

a measurement set on the MCs in the update step (5.22) can be controlled by an accurate modeling of the detection probability depending on the sensor specific FOVs. Examples of how to determine $p_D^{(i),S_j}$ are presented in Section 7.2.

5.3.3 Fusing Sensors with Different Recognition Abilities

Depending on the type of the sensors and the used object detection and classification algorithms, sensors may differ in their recognition ability $\mathcal{RA}^S$. Consider again the example of two sensors with partially redundant FOVs depicted by Figure 5.5. Additionally, in this case sensor S_1 and sensor S_2 feature different recognition abilities

$$\mathcal{RA}^{S_1} = \{P\}, \quad (5.25)$$

$$\mathcal{RA}^{S_2} = \{C\}. \quad (5.26)$$

The expressions in (5.25) and (5.26) express that S_1 is able to detect pedestrians (P) while S_2 only to detects cars (C). In the example of Figure 5.5, one of the pedestrians is in the FOV of S_1 and S_2 . Thus, the FOV-based detection probability (5.24) of both sensors is high. However, according to (5.25), only S_1 is able to detect the pedestrian. Providing measurements of both sensors to a standard multi-sensor GM-PHD filter which does not consider the sensors' $\mathcal{RA}^S$, a measurement of S_1 confirms the MC representing the pedestrian while a measurement of S_2 does not. Thus, the contradicting measurements will result in MCs with low weights which might prevent a continuous object tracking. Modeling the sensor specific $\mathcal{RA}^S$ as a set of class hypotheses in terms of the DST as proposed in (5.25) and (5.26), the tracker's FOD $\Omega_T = \{C, P\}$ can be constructed based on the elementary class hypothesis contained in the sensors' $\mathcal{RA}^S$. Since each MC holds the knowledge about its class, the probability

$$BetP_{m_C^{(i)}}(\mathcal{RA}^{S_j}) \quad (5.27)$$

that the i^{th} MC can be detected by sensor S_j featuring $\mathcal{RA}^{S_j}$ can be determined. A similar approach which uses DST to represent the sensors' recognition ability in a JIPDA framework has been presented in [MMD10]. Consequently, the integration of the object classification in the GM-CMMPHD filter facilitates an adaption of the sensors' detection probability for objects with class BBA m_C by:

$$p_D^{(i),S_j} = BetP_{m_C^{(i)}}(\mathcal{RA}^{S_j}) \cdot p_D^{S_j}(\bar{x}^{(i)}). \quad (5.28)$$

This dependency is visualized by the dashed arrow in Figure 5.4 connecting the estimated class BBA with the detection probability.

However, the $\mathcal{RA}^S$ is not required to model only one class hypothesis. Dependent on the sensor it can contain multiple elementary class hypotheses or even combinations of class hypotheses if the sensor is not able to distinguish single classes. In case of a detection only sensor, the $\mathcal{RA}^S$ equals the tracker's FOD Ω_T .

In general, two types of object recognition strategies can be distinguished. The first strategy detects objects using classification by applying a class specific classifier to the measurements. Thus, the recognition ability is limited to the object class the classifier has been trained for. The second method detects objects independent of their class. Here the object class is unknown or can be determined afterwards using a classifier based on features of the detected objects. Examples for both methods are also given in Section 7.2.

Chapter 6

Laserscanner-Based Detection and Classification of Road Users at Intersections

While the previous chapters concern general methods for object tracking, this chapter describes methods explicitly developed for road user recognition at the public intersection (Chapter 2.2). Nevertheless, the developed object detection and classification methods are not restricted to this particular intersection setup but have been parametrized for this application. Thus, the intersection perception system is best qualified to motivate and explain the entire object recognition approach. Starting with the introduction of information maps as a practical approach for position dependent parametrization in Section 6.1, the methods to detect and classify objects are discussed in Section 6.2 and Section 6.3.

6.1 Information Maps

Information maps are used to store system and position dependent parameters for all algorithms involved in the detection, classification and tracking process. The concept of information maps was introduced in [WMRD13]. Since the intersection perception system outlined in Chapter 2.2 is a static system, many characteristics of the system can be modeled dependent on the position. The areas of the intersection that should be considered for object detection, the likelihood of objects classes, which is e.g. for cars higher on streets than on sidewalks, or the detection probability which decreases with the distance to the intersection center are only some examples. Representing that information discretized in 2D grid maps, facilitates the convenient parameter storage and access even if the underlying functions are highly non-linear or discontinuous. In this thesis the origin of the information maps coincides with

Figure 6.1: Lane markings map (a) and ROI map (b).

the origin of the intersection coordinate system C_I . The basis for each information map is a highly accurate map containing the position of the lane markings at the intersection also given in the C_I system. A picture of this lane markings map is given in Figure 6.1a. Using the lane markings map as a basis, different information maps with certain resolutions containing various parameter information were created. To illustrate the concept of information maps a region of interest (ROI) map is shown as example in Figure 6.1b. In that ROI map locations which should not be considered for object detection containing buildings, vegetation, parking lots, or the intersection egresses are blackened out, while the ROI is bright in color. For reasons of clarity, the lane markings map is visualized in the background. Further examples for information maps modeling parameters are given throughout the thesis in Figures 6.12, 7.1, 7.2, 7.3, and 7.4.

6.2 Object Detection

The issue of the object detection based on laserscanner measurements is to determine which measurement point has been originated by which object. Subsequently, an estimate for the number of objects and their appearance in the current measurement can be given. Until now, a multitude of methods to detect objects in 3D point clouds were published. Most of the methods exploit sensor characteristics [HHW10; RD09], assume object models [Kam07; Mun11; Wen08], or are based on point cloud densities [KKD12]. However, those methods were developed for sensors mounted near the ground and are highly depend on the accuracy and density of the measurements. In this thesis new object recognition methods for sensors with a bird's-eye perspective are proposed. In contrast to the work of Stimming [SR11] in the Inter Safe 2 project [RFW⁺11], the methods proposed in this thesis highlight the issue of detecting objects which are only partially visible or represented by patchy measurements. Two major challenges influenced the development and designing of the road user detection

Figure 6.2: Block scheme of the object detection and generation method. Input is the point cloud of the laserscanner system and output are oriented bounding boxes (OBBs) and point cloud clusters which are associated to the OBBs.

methods: First, in case of high distances between the sensor and the object or objects with poor reflection characteristics, only sparse or patchy measurements of an object might be gathered. A comparison of the reflection characteristics of different colored vehicles is given in Figure B.3. Using conventional 3D clustering methods, leads to missed detections and so-called over-clustering which means that a single object is represented by multiple clusters. Second, due to the large perception area, objects may appear in adverse poses to the sensors. Hence, often only one aspect of an object is visible which makes it nearly impossible to determine the correct oriented bounding box (OBB) of an object in terms of size and orientation. To overcome those shortcomings, a combination of 3D point clusters with tracks of pathways in a 2D grid map is proposed for object detection in this thesis. In Figure 6.2 a block scheme of the 2D/3D object detection method is given.

After the subtraction of background points originated from static environment objects like buildings, lamp-posts, or the street surface, the remaining foreground points are clustered in 3D and tracked in 2D. In 3D the full spatial resolution and measurement accuracy of the laserscanners can be exploited to achieve a detailed clustering result even if objects are close to each other. Therefore, a realtime-capable implementation of the density based spatial clustering for applications with noise (DBSCAN) algorithm [EK SX96] is used. Since the single measurement based 3D clustering entails the risk of over-clustering, simultaneously a 2D pathway tracking is performed. The pathway tracks (PTs) are determined by projecting the 3D measurement points onto a 2D grid map in parallel to the street surface. Incorporating measurements of multiple frames, the pathway tracking is robust against over-clustering and the tracking additionally provides the objects' orientations which improves the OBB determination even if objects are partly occluded. Combining the 3D clusters and 2D PTs results in a reliable and highly accurate road user detection. Although, the object detection method has been developed to detect road users based on laserscanner measurements, the concept can be used to detect arbitrary objects based on any sensor providing 3D point clouds, e.g. a stereo camera or radar

sensor.

6.2.1 Background Subtraction

In case of static sensor setups background subtraction methods to segment foreground and background measurements are usual. In this thesis the foreground is composed of measurements originated from road users and the background contains measurements generated by overall static objects, e.g. buildings and vegetation. Generally, a model for the background is generated which is compared with the measurements in each measurement cycle. Subsequently, measurements matching the background model are subtracted from the measurement set which yields the required foreground measurements.

As a first step, all measurement points that are not in the ROI of the intersection system are subtracted based on the ROI-map depicted by Figure 6.1b. Subsequently, static or adaptive background models might be used dependent on the environment. While static models are trained offline, adaptive models facilitate the consideration of slight or repetitive changes in the environment. In the following sections a static and an adaptive background model are introduced. Both models exploit the fact that the laserscanners scan their environment with a fixed step size along the azimuthal and the elevation angle. Thus, with the angular scanning ranges and resolutions given in Table 2.2, each laserscanner emits a laser pulse in 1368 directions per scan.

Static Background Model

The static background model is based on the assumption that in each scanning direction the detection range is bounded by a background object. In case of the intersection setup of Chapter 2.2 this assumption is valid, since all laserscanners face to the ground due to their high mounting position. In Figure 6.3 the measurements of one layer which provide a height profile of the surrounding are depicted. In this case most of the laser beams are reflected by the street surface but in the middle of the scan, at $\gamma \in [-20^\circ; 0^\circ]$, a vehicle is located. Obviously, the measurement points originated by the vehicle can be segmented by choosing only those measurements which have a shorter distance to the sensor than the surrounding street. The static background model has to be determined offline, since nothing but background objects are allowed to be in the perception area of the laserscanners.

Figure 6.3: Measurement points of one layer and one sensor in polar sensor coordinates.

Adaptive Background Model

The adaptive background model represents the range measurements distribution $p_b(r)$ of each laser beam b with a Gaussian mixture (GM) consisting of N_{GM} weighted univariate Gaussian distributions:

$$p_{b,k}(r) = \sum_{i=1}^{N_{GM}} w_k^{(i)} \mathcal{N}(r; \bar{r}_k^{(i)}, P_{r,k}^{(i)}), \quad (6.1)$$

where $w_k^{(i)}$ is the weight of the i^{th} Gaussian distribution $\mathcal{N}(r; \bar{r}_k^{(i)}, P_{r,k}^{(i)})$ with mean $\bar{r}_k^{(i)}$ and covariance $P_{r,k}^{(i)}$ at time k . Further, the weights sum to unity, due to the normalization of the GM. With the representation of range distributions as GMs, the mixture of Gaussians (MoG) algorithm [SG99] for adaptive background modeling can be used to adaptively segment foreground and background measurements. In contrast to the original purpose of the MoG algorithm [SG99] not pixel values of camera images, but the range measurements of a laserscanner are assumed to be independently distributed. A application of the MoG algorithm to segment laserscanner data was already published by [OA11]. Nevertheless, the MoG segmentation was based on images of an aligned video camera instead of range measurements.

A further, widely used technique to adaptively model the background based on range measurements are 2D occupancy grid maps [Elf89; TBF05]. Due to the mounting positions of the sensors, a common 2D grid map is not sufficient. Since layer individual 2D occupancy grids are not as adaptive as beam individual GMs and 3D maps as in [DMX10; TPB06; WHB⁺10] are too complex for a realtime system with 14 laserscanners, the MoG segmentation for laserscanner measurements

is proposed.

The essence of the adaptive segmentation algorithm of [SG99] is that frequently received range measurements are likely to belong to the background. Thus, they are represented by a Gaussian with high weight $w^{(i)}$ and low variance $P_r^{(i)}$ in the GM of the corresponding laser beam. Hence, the portion $\omega^{(i)} = w^{(i)}/P_r^{(i)}$ was introduced. Increasing with both a high weight and a low variance, this portion is a sufficient indicator for a Gaussian representing background measurements. After ordering the Gaussians of one beam by decreasing $\omega^{(i)}$, the first

$$N_b = \arg \min_{n_b} \left(\sum_{i=1}^{n_b} w^{(i)} \geq \tau_b \right) \quad (6.2)$$

Gaussians are used to model the background, where τ_b is a threshold which expresses the prior probability for background in the measurements [KB01].

The foreground segmentation of measurements with the adaptive GM background model and the accompanied update of the GMs is summarized in Algorithm 2. The input of the segmentation are range measurements r gathered at the azimuthal angle γ and elevation angle β . The parameters of Algorithm 2 are the learning rate $1/\alpha$ which is used to regulate the adaption ability of the background model and the gating parameter δ_B which defines the gating area around the Gaussians' means according to the Mahalanobis distance (MHD) $d_{MHD}(r_{k+1}, \bar{r}_k^{(i)})$ (3.35).

The benefit of the adaptive compared to the static background model (Section 6.2.1) is that it is able to adapt to changes in the background, e.g. a temporal construction site at the intersection or parked cars. Further, it enables the modeling of periodically moving background objects, e.g. branches or leaves moving in the wind. However, as discussed in several publications reviewed in [BEV08], the choice of the learning rate $1/\alpha$ is critical. On the one hand, a learning rate near one facilitates a fast background model adaption but segments cars waiting at traffic lights as background. On the other hand, a large value for $1/\alpha$ retains waiting cars in the foreground but reduces the adaptability to handle moving vegetation which has to be considered in outdoor applications.

Since the static background model turned out to be more robust and computationally efficient for the segmentation of foreground measurements of the laserscanner system, it is used in the following.

Algorithm 2 Foreground Segmentation with the Adaptive Gaussian Mixture Background Model

Require: parameters: learning rate $1/\alpha > 1$ and gating parameter $\delta_B \geq 1$
 measurements: $Z_{k+1} = \{(r_{k+1}^{(1)}, \gamma_{k+1}^{(1)}, \beta_{k+1}^{(1)}), \dots, (r_{k+1}^{(M)}, \gamma_{k+1}^{(M)}, \beta_{k+1}^{(M)})\}$

- 1: **for all** measurements $(r_{k+1}, \gamma_{k+1}, \beta_{k+1}) \in Z_{k+1}$ **do**
- 2: find GM characterizing the laser beam b with $(\gamma_{k+1}, \beta_{k+1})$.
- 3: **for all** Gaussians $i = 1, \dots, N_{GM}$ **do**
- 4: **if** $d_{MHD}(r_{k+1}, \bar{r}_k^{(i)}) \leq \delta_B |P_{r,k}^{(i)}|$ **then**
- 5: mark i^{th} Gaussian as matched and update it:
- 6: $w_{k+1}^{(i)} = (1 - \alpha)w_k^{(i)} + \alpha$
- 7: $\rho_{k+1} = \alpha \mathcal{N}(r_{k+1}; \bar{r}_k^{(i)}, P_{r,k}^{(i)})$
- 8: $\bar{r}_{k+1}^{(i)} = (1 - \rho_{k+1})\bar{r}_k^{(i)} + \rho_{k+1}r_{k+1}$
- 9: $P_{r,k+1}^{(i)} = (1 - \rho_{k+1})P_{r,k}^{(i)} + \rho(r_{k+1} - \bar{r}_{k+1}^{(i)})(r_{k+1} - \bar{r}_{k+1}^{(i)})^T$
- 10: **else**
- 11: $w_{k+1}^{(i)} = (1 - \alpha)w_k^{(i)}$
- 12: **end if**
- 13: **end for**
- 14: normalize weights: $w^{(i)} = w^{(i)} / \sum_{i=1}^{N_{GM}} w^{(i)}$
- 15: **if** all Gaussians are unmatched **then**
- 16: measurement is foreground
- 17: initialize new Gaussian with mean r_{k+1} , initial weight and variance
- 17: replace Gaussian with lowest weight, if GM already contains N_{GM} Gaussians
- 18: **else**
- 19: determine Gaussians modeling the background according to (6.2)
- 20: **if** measurement matched with one of the first N_B Gaussians **then**
- 21: measurement is background
- 22: **else**
- 23: measurement is foreground
- 24: **end if**
- 25: **end if**
- 26: **end for**
- 27: **return** updated background and segmented foreground measurements

6.2.2 3D Clustering

After the segmentation of the foreground measurements originated from road users in the FOV of the laserscanner system, one way to detect objects in the remaining 3D point cloud is clustering. In this thesis the density based spatial clustering for applications with noise (DBSCAN) algorithm developed by Ester *et al.* [EKSX96] is used for this purpose. As suggested by its name, the DBSCAN algorithm sets up clusters in regions of high point densities. Therefore, only two parameters: the search radius ϵ and the minimum number of points C_{min} within ϵ are required to set up a cluster. The algorithm starts with an arbitrary point p in the point cloud and checks if at least C_{min} points are in the ϵ -neighborhood of it. If this is the case the point p is marked as a cluster core point and all points P_C in its ϵ -neighborhood belong to this cluster. Subsequently, this check is performed for each point in P_C . Assuming the point cloud and ϵ -neighborhood depicted in Figure 6.4 as well as $C_{min} = 3$, the iteration of the just described procedure results in the red marked cluster core points. The green points are located in the ϵ -neighborhood of core points, but are no core points themselves, since they have less than C_{min} points in their ϵ -neighborhood. These points are denoted as boundary points. The last class of points are outliers which are also not surrounded by C_{min} points and do not have a cluster core point in their ϵ -neighborhood. These points are visualized in blue in Figure 6.4.

To perform the neighborhood check, the distances between the currently considered point and all other points in the point cloud have to be calculated. Thus, the complexity of the original version of the DBSCAN algorithm [EKSX96] is quadratic in the number of points $\mathcal{O}(N^2)$. For a realtime implementation of the DBSCAN algorithm the density based grid spatial clustering for applications with noise (DB-GridSCAN) algorithms was developed which significantly reduces the number of distance calculations by a discretization of the (x, y) -space in quadratic grid cells. Each grid cell has an edge length of ϵ , thus only the nine grid cells including and surrounding the current point have to be regarded. In the example depicted by Figure 6.5 these grid cells are marked light blue. Consequently, the number of inter point distance calculations can be reduced significantly because all points outside the blue area have not to be considered. As the 2D grid is only exploited for point selection, still the 3D distances between the points are used.

Figure 6.6 compares the runtime of the DBSCAN [EKSX96] and the proposed DBGridSCAN dependent on the number of clustered points. With an increasing number of points the DBGridSCAN significantly outperforms the DBSCAN. While the runtime curve of the DBSCAN shows the expected quadratic increase, the runtime increase of the DBGridSCAN is almost linear. Comparing the runtime of both algorithms clustering 1,500 points, the DBSCAN takes approximately 75 ms whereas the DBGridSCAN takes only as little as 5 ms. So the DBGridSCAN needs

Figure 6.4: Illustration of the DBSCAN principle.

Figure 6.5: Illustration of the DBGridSCAN principle.

Figure 6.6: Runtime comparison of DBSCAN and DBGridSCAN applied on the number of points to cluster.

much less computational power without losing accuracy.

Due to the sensors' mounting positions the resolution of the laserscanner system decreases with the distance to the center of the intersection. Thus, in the approaches of the intersection objects are represented by a much smaller number and density of measurement points compared to the central widespread. To achieve accurate clustering results in the entire field of perception, position dependent parameters $\epsilon(x, y)$ and $C_{min}(x, y)$ are used. Despite the adaption of the parameters based on the location of the measurement points, the DBSCAN algorithm tends to over- or under-clustering dependent on the choice of its parameters. This is a common issue of single measurement-based clustering approaches, especially if objects with different size, e.g. vehicles and pedestrians, have to be detected. Since the use of knowledge about tracked objects in the clustering algorithms, as proposed by [HW12], entails the risk of biased tracking results, the next section presents a method to correct over-clustered objects regarding only measurement information.

6.2.3 2D Pathway Tracking

The 2D pathway tracking is a clustering method in addition to the DBSCAN algorithm which is also based on the foreground measurement points (see Figure 6.2). In contrast to the DBSCAN algorithm, several successive measurements are exploited for object detection. Since it is computationally problematic to track the individual measurement points, they are discretized by the projection onto a 2D grid, which is in the (x, y) -plane, to reduce complexity. As in grid mapping [Elf89], a 2D grid with quadratic cells is spanned in the (x, y) -plane of the local coordinate system. In each frame, the measurements are projected onto the grid which has been initialized with zeros. A grid cell including at least one measurement point is voted with one. Combining the grids of N_{PG} consecutive scans with an *OR* operation per grid cell, pathways unveil on the resulting pathway grid (PG). The generation of the pathway grid is illustrated by Figure 6.7 and follows the remarks in [MSN⁺10].

The benefits of the pathway grid representation of the laserscanner system measurements are: first, it facilitates an object generation which is less prone to over-clustering, since patchy object detections are connected by the motion blur effect in the pathway grid. Second, owing to the motion blur effect each pathway contains history information about its position. Thus, the generally complex data association in multi-object tracking can be solved easily by contour clipping. Hence, a computationally efficient estimation of the position, velocity, and dimension of the maneuvering objects is feasible. A summary of the pathway tracking method which follows the multi-object tracking concept introduced in Figure 3.3 is given by the block scheme in Figure 6.8.

Figure 6.7: Creation of pathway grid (PG). In the 2D to 3D projection step, the 3D measurement points (orange) are projected on the 2D grid (white cells). The pathway grid is determined based on the fusion of N_{PG} consecutive grids using an *OR* operation.

Figure 6.8: Block-scheme of pathway tracking.

After the generation of the pathway grid from N_{PG} consecutive measurement frames, the pathways are detected using the common image labeling algorithm [CCL04] implemented in [Ins14]. Each pathway is described by its contour which is a closed chain of side-by-side grid cells and illustrated in green in Figure 6.8. Since pathway tracking only improves the clustering of larger objects like vehicles, pathways whose extension is too small for vehicles are not considered. The positions of the pathways are equal to the determined mean of the contour points.

Afterwards, the detected pathways are tracked in grid coordinates using a bank of Kalman filters as discussed in Section 3.3. The objects' motion is modeled by a linear constant velocity (LCV) model (3.37). The state x of the pathway track (PT) represents the estimated position and velocity of the corresponding pathway and is updated by the position of the assigned pathway.

The contour of the pathway is not estimated but used as an attribute of the PT which is exploited for measurement-to-track association and dimension determination. Especially in the data association and merge detection module (see Figure 6.8), the contours are essential for the proposed pathway tracking method. As already mentioned, the pathways contain history information about their position due to the motion blur effect in the pathway grid. Consequently, a PT is only updated if it has a non-empty cut set with at least one detected pathway. This case is illustrated by the left image in the data association and merge detection module of Figure 6.8. Here, the non-empty cut set of the predicted PT and the detected pathway is marked in grey. Obviously, the size of the cut set depends on the extent of the considered object and its velocity. Thus, the size of the grid cells and the number of considered history frames has to be adapted according to the use-case. In the case that one PT can be associated to multiple pathways, simply the pathway with the largest cut set is used to update the PT to keep the complexity low. While the state is updated according to the Kalman filter equations (3.15) and (3.16), the contour is adopted from the pathway as it stands.

To realize a reliable pathway tracking, the case that multiple PTs are associated to one detected pathway has to be handled. Since clutter measurements can be almost ruled out due to the filtering of small objects in the pathway grid, two reasons for multiple associations are feasible: First, multiple objects are represented by one pathway or second, one pathway is represented by multiple PTs. Both cases are depicted in the right picture of the data association and merge detection block in Figure 6.8. In the first case (Figure 6.8(a)) the current measurement groups two objects. Thus, the PTs represent the current scene correctly and should be kept. In contrast, the measurement is correct and the PTs should be merged in the second case (Figure 6.8(b)). Consequently, the current case has to be initially identified. By analyzing this issue, an application specific feature to classify the current case was found. Since in this thesis the 2D pathway tracking is used to detect road

users at a public intersection, the maximum width w_{max} of an object, which is the width of a truck, is used to distinguish the cases (a) and (b). If a PT's width exceeds w_{max} (case (a)) it is updated with the grey colored cut set of the detected erroneous pathway and the predicted PT. This is an usual approach in image-based blob tracking to temporary bridge measurement frames with under-clustered objects for instance discussed in [KWM94]. In case (b), w_{max} is not exceeded which leads to a merge of the affected PTs. To merge several PTs, the convex hull around their contours yields the contour of the merged PT. After the merging step a one-to-one association exists and the merged PT can be updated with the associated pathway as described before. Dependent on the application of the pathway tracking even other features to classify cases (a) and (b) can be used.

Finally, with the knowledge of the PTs' direction of motion, the orientation of the corresponding object can be determined. Thus, the pathway tracking enables the determination of the objects' orientations which is a significant benefit especially in cases of patchy object detections. As a result the 2D pathway tracking provides PTs featuring a position, velocity, orientation, and contour.

6.2.4 Correction of Over-Clustered Objects and Object Generation

To complete the laserscanner-based object detection, the results of the DBSCAN in 3D and the 2D pathway tracking are combined. The result of the DBSCAN features clusters of the original point cloud gathered by the laserscanner system while the 2D pathway tracking provides object information in terms of PTs described by their position, velocity, orientation, and contour. The point clusters of the DBSCAN algorithm provide the basis for the combination of the two clustering results because they exploit the full 3D information and high sensor resolution. Since the application requires the separation of objects which are close to each other, e.g. a group of pedestrians, the parameters of the DBSCAN algorithm have to be chosen respectively. Nevertheless, requiring to cluster objects which significantly differ in their extension and reflection characteristics and parameterizing the DBSCAN algorithm for objects close to each other carries the risk of over-clustering. To correct over-clustered objects, the PTs embracing several objects are used. In Figure 6.9 the clusters are visualized with points of the same color and the PTs appear as grey areas in the background. All point clusters embraced by the same PT are replaced by a new cluster which is composed of the affected cluster points. Thus, the PTs facilitate the correction of over-clustered objects. Nevertheless, the final clustering result is still based on the full 3D information and high sensor resolution, since the 2D PTs are only used to correct over-clustered objects.

Figure 6.9: Visualization of 3D point clusters as points with identical color and the corresponding PTs in the background (grey).

The final step in object detection is the object generation which provides oriented bounding boxes (OBBs) for each point cluster as introduced in Section 3.4.2. To determine the objects' lengths and widths, the accuracy of the objects' orientations ϕ is crucial. Thus, three sources to calculate the objects' orientations with increasing uncertainty σ_ϕ are exploited. The first and most confident way to determine ϕ are the PTs. Here, the orientation of the OBB is based on the estimated motion direction of the corresponding PT. The standard deviation of the orientation σ_ϕ is calculated based on the estimated uncertainty of the PT's velocity. If an association of a point cluster and a PT is not possible environment information can be used to determine ϕ . Detecting road users at an intersection, the objects in the approaches or egresses are most likely to move towards the intersection center or away from it respectively. Accordingly, the orientation in the approaches and egresses is position dependent and suitable to be stored in an information map as described in Section 6.1. Being able to consider lane changes and turning maneuvers, a standard deviation of the orientation of $\sigma_\phi = 10^\circ$ turned out to be appropriate. Finally, if an object is too small or moves too slow to create a PT and is not located on the intersection's approaches or egresses, it is assumed to be oriented in the (x, y) -direction of the point cloud's major axis which is determined by singular value decomposition (SVD). Due to patchy and incomplete object detections, the SVD contains the highest uncertainty, i.e. $\sigma_\phi = 90^\circ$. Regardless of the source of the objects' orientation, the length is defined as the extension of the object in orientation direction (x -direction) and the width is defined as the extend perpendicular to the orientation direction (see Figure 3.5). The algorithm to determine the OBB for detected objects is summarized in Algorithm 3.

Especially in the central area of the intersection the pathway tracking improves the accuracy of clustering and the OBBs significantly. An example is given by Figure 6.10. The three grey areas show the area of the PTs. In both images the left object moves

Algorithm 3 Determine Oriented Bounding Boxes for Detected Objects

Require: Set of 3D point cloud clusters $C_{3D} = \{c_{3D}^{(1)}, \dots, c_{3D}^{(N_{3D})}\}$ and set of PTs $C_{PT} = \{c_{PT}^{(1)}, \dots, c_{PT}^{(N_{PT})}\}$.

- 1: $B = \emptyset$
- 2: **for all** $c_{3D} \in C_{3D}$ **do**
- 3: $c_{PT}^* = \text{findAssociatedPT}(c_{3D}, C_{PT})$
- 4: **if** found c_{PT}^* **then**
- 5: $[\phi, \sigma_\phi] = \text{getOrientationFromPT}(c_{PT}^*)$
- 6: **else if** c_{3D} located on an approach or egress **then**
- 7: $[\phi, \sigma_\phi] = \text{getOrientationFromMap}(c_{3D})$
- 8: **else**
- 9: $[\phi, \sigma_\phi] = \text{getOrientationFromSVD}(c_{3D})$
- 10: **end if**
- 11: $OBB = \text{determineOBB}(c_{3D}, \phi)$
- 12: $B = B \cup (OBB, \sigma_\phi)$
- 13: **end for**
- 14: **return** Set B with OBBs and orientation uncertainties corresponding to the entries in C_{3D} .

from the upper left to the lower right and the others move from the lower right to the upper left corner of the image. At the top of the image the OBBs determined by the DBGridSCAN clustering are illustrated. Since in the central widespread no map information is available, the objects' orientation is based on SVD. In this case the object in the center is over-clustered and the orientation of the object on the left faces to the opposite direction. In contrast, the result of the DBGridSCAN with pathway tracking shows no over-clustering and the OBBs are properly oriented. Consequently, the combination of the DBGridSCAN and pathway tracking facilitates the correction of over-clustered objects. A comprehensive evaluation is provided in Section 8.2.

6.3 Road User Classification

The measurement-based road user classification exploits point cloud features of the detected objects to distinguish between the classes bike (B), car (C), pedestrian (P), and truck (T).

In [PLN09] and [AMB07] a broad overview of features mainly used to classify vehicles and pedestrians is given. Further, in [ZK07] part-based features for pedestrian

Figure 6.10: (top) clustering result of the DBGridSCAN. (bottom) clustering result of the DBGridSCAN with pathway tracking. OBBs are marked blue, their orientation is shown by red lines, measurements are marked orange and the area of PTs is marked grey.

classification and in [SATS10] and [GCB⁺10] layer-based approaches for pedestrian recognition are addressed. The majority of those features usually used for laserscanner-based object classification rest upon geometric attributes of the objects. Obviously, e.g. the extent is a strong feature to distinguish vehicles and pedestrians assuming ideal visibility conditions. However, features which are somehow related to the objects' width and length are not always suitable for road user classification. This is not surprising, since due to its viewpoint or occlusions the used sensors are not able to detect the entire objects in most of the cases.

Consequently, the independent point cloud features used for road user classification are:

- height of the cluster

$$f_h^{(i)} = \max_{j \in N_p^{(i)}} p_z^{(j)}, \quad (6.3)$$

where $N_p^{(i)}$ denotes the number of points in the i^{th} cluster and $p_z^{(j)}$ is the z -component of the j^{th} measurement point $p^{(j)} = [p_x^{(j)}, p_y^{(j)}, p_z^{(j)}]^T$

- absolute values of the x -, y - and z -component of the normalized point cloud clusters' major axis $[f_{pc(x)}, f_{pc(y)}, f_{pc(z)}]^T$ determined by SVD
- standard deviation of the point cloud's measurement points

$$f_{std}^{(i)} = \sqrt{\frac{1}{N_p^{(i)} - 1} \sum_{j=1}^{N_p^{(i)}} \|p^{(j)} - \bar{p}^{(i)}\|^2} \quad (6.4)$$

with the mean of the point cloud cluster $\bar{p}^{(i)} = 1/N_p^{(i)} \sum_{j=1}^{N_p^{(i)}} p^{(j)}$

In Figure 6.11 the class dependent distributions $f(\cdot|c)$ of the features are visualized, which are based on several thousand labeled measurements of the laserscanner system.

The distribution of the height feature $p(f_h|c)$ in Figure 6.11a primarily facilitates the discrimination of trucks from all the other classes. Due to the bird's-eye view of the laserscanners, the height determination is not influenced by occlusions which makes it a reliable feature for truck classification even in high distances. The components of the normalized major axis $[f_{pc(x)}, f_{pc(y)}, f_{pc(z)}]^T$ of an object involve the ratio of the object's length, width, and height as well as the distribution of the measurement points within the point cloud. Thus, those features are highly suitable to distinguish objects which are extended in z -direction, i.e. pedestrians and bikes, from objects which are extended in (x, y) -direction, i.e. cars and trucks. That faculty of differentiation is also emphasized by Figures 6.11b, 6.11c, and 6.11d. Finally, the standard deviation feature f_{std} which characterizes the compactness of the point cloud is an indicator for pedestrians. In the majority of cases pedestrians are represented by compact point clouds which results in the peak of the distribution for pedestrians in Figure 6.11e. Additionally, due to their large extent the recognition of trucks is possible. To make use of all features, a feature distribution value never reaches zero but is bounded by $p(\cdot|c) = 1e^{-4}$ which approved as a sufficient lower bound.

The $N_f = 5$ features, summarized in $F = \{f_h, f_{pc(x)}, f_{pc(y)}, f_{pc(z)}, f_{std}\}$ are calculated

Figure 6.11: Class-dependent feature distributions used for measurement-based object classification.

for each detected object and are used for classification with Bayes' rule:

$$p^{L,(i)}(c_j|F) = \frac{p^{L,(i)}(c_j) \prod_{k=1}^{N_f} p^{L,(i)}(f_k|c_j)}{\sum_{j=1}^{N_c} p^{L,(i)}(c_j) \prod_{k=1}^{N_f} p^{L,(i)}(f_k|c_j)}. \quad (6.5)$$

Thus, the probability that the i^{th} object is of class c_j based on the set of laserscanner

Figure 6.12: Information map containing the position dependent prior class probability for the class car (C).

features F is denoted by $p^{L,(i)}(c_j|F)$. Consequently, a probability for each of the $N_c = 4$ classes is determined. To incorporate the position dependent probability of class occurrence, e.g. a vehicle is more likely to occur at the street than at the sidewalk, the prior class probability $p^{L,(i)}(c_j)$ depends on the i^{th} object, concretely on the position of the object at the intersection. Similar to the feature distributions, the position dependent prior class probability was determined based on manually labeled data sets gathered at the investigated intersection. Hence, the prior class probability for each class is stored in an information map as depicted by Figure 6.12 for the class car (C) and in Figure 6.13 for the class pedestrian (P). Summing up the values of identical cells in all four prior class probability maps yields unity.

The probability $p^{L,(i)}(c_j)$ is coded from zero (black) to one (white). Incorporating the position dependent class probabilities, significantly improves the classification result especially in cases of partly occluded object detections or objects approaching the intersection.

Finally, the result of the measurement-based road user classification is summarized by the four class probabilities

$$\left\{ p^{L,(i)}(B|F), p^{L,(i)}(C|F), p^{L,(i)}(P|F), p^{L,(i)}(T|F) \right\} \quad (6.6)$$

which sum to unity for each detected object. The reader should note that with (6.6) no decision about the class of the object is made. Instead of assigning the class with

Figure 6.13: Information map containing the position dependent prior class probability for the class pedestrian (P).

the highest probability to the object, the determined class probabilities are used as masses of the class-BBA introduced in Chapter 5 which represents the knowledge of the laserscanner system about the object's class on the credal level [SK94].

Chapter 7

Application of the CMMPHD Filter to Road User Tracking at Intersections

In this chapter the application of the developed CMMPHD filter to track and classify road users at the urban intersection with the perception setup introduced in Chapter 2.2 is described. The multi-sensor intersection perception system gathers environment information with 19 sensors of three different types. Consequently, 19 different FOVs and three recognition abilities have to be considered in sensor fusion. The sensor specific parameters and their determination are described in Section 7.1. After the sensor models, the parameterization of the CMMPHD filter for road user tracking at the intersection is addressed in Section 7.2. Here the focus lies on the integrated class estimation based on features of the PHD's MCs and the adaptive modeling of the objects' motion.

7.1 Sensor Models

As sketched out in Figure 5.1, each sensor that should be fused within the CMMPHD filter has to provide information about the objects' position, orientation, attributes, and the respective sensor dependent uncertainties. Since each measurement directly includes components of the objects' states, a linear measurement model $\underline{H}$ (3.8) is sufficient to predict the measurements in the PHD update.

The detections as well as the tracks of the road users are represented as oriented bounding boxes (OBBs) whereby the point target assumption involved in the CMMPHD filter is violated. To realize a consistent and reliable extended object tracking with the CMMPHD filter, reference points (RPs) as introduced in Section 3.4.2

are used. Considering the static intersection sensor setup, at each position of the intersection the best observable sensor system specific RP can be determined and stored in an information map which is part of the corresponding sensor model. As the measurement values of each detected object are referred to the RP given in the sensor model, also the measurements are predicted for that RP in the update step of the filter. Although the usage of RPs works well in this thesis, especially in applications with moving sensors or unknown environments more sophisticated extended target RFS tracking approaches are advisable. These approaches are not considered in this thesis but are briefly sketched in the future work Section 9.2.

Apart from the RPs, the sensors' FOV as well as their recognition abilities are required to fuse the sensor measurements to a consistent environment representation. The sensor characteristics are introduced in the following.

7.1.1 Laserscanner

The methods to detect and classify road users based on measurements of the laserscanner system were already described in Chapter 6. The laserscanner system facilitates the detection and classification of road users in the central widespread as well as in the approaches up to high ranges. Due to the large FOV, the system detects objects from various viewpoints which leads to the need of position dependent RPs and uncertainties of the objects' extent. The considered RPs and uncertainty ranges are summarized in Table 7.1. Furthermore, the classification capability of the laserscanner system contains objects which significantly differ in their extent, e.g. trucks and pedestrians, and reflectivity characteristics, e.g. pedestrians and different painted cars. As this directly influences the accuracy of position determination, class specific position uncertainties are sufficient. The used uncertainty ranges are also given in Table 7.1.

Analyzing the object detection performance in a huge set of sequences containing over 100,000 object detections in different environment conditions, the position dependent detection probability of the laserscanner system $p_D^{L,(i)}$ is determined. In Figure 7.1 the detection probability of the laserscanner system stored in an information map is visualized. The bright areas feature a high detection probability $p_D^{L,(i)} = 0.85$ and are flanked by dark areas which are not in the FOV of the laserscanner system and feature a low detection probability $p_D^{L,(i)} = 1e^{-4}$. Furthermore, the bright area in Figure 7.1 reveals the range of the laserscanner system to detect objects. Measured from the corresponding stop lines, a continuous road user detection from a distance of approx. 63 m in the upper left approach and from approx. 40 m in the other two approaches is feasible. The differences owe to the mounting positions of the sensors

Table 7.1: Summary of Sensor Parameters

	description	laserscanner	LR camera	HR camera
$\mathcal{RA}^S$	recognition ability (Section 5.3.3)	$\{B, C, P, T\}$	$\{C, T\}$	$\{P\}$
p_D^S in FOV	detection probability (Section 5.3.2)	0.85	0.8	0.9
RP	reference point (Section 3.4.2)	FC, CC, RC	FC	CC
p_z^S	confidence in classification result (5.14)	0.8	0.9	0.9
$\sigma_{x_p}[m]$	standard deviation of x position	0.2 – 0.5	0.5	0.5
$\sigma_{y_p}[m]$	standard deviation of y position	0.2 – 0.3	0.3	0.2
$\sigma_\phi[^\circ]$	standard deviation of orientation	10 – 90	10	90
$\sigma_l[m]$	standard deviation of length	0.2 – 4	4	0.1
$\sigma_w[m]$	standard deviation of width	0.2 – 1	0.2	0.1

and branches of trees hanging above the street.

According to (6.6), the laserscanner system is able to provide class probabilities for the classes bike (B), car (C), pedestrian (P), and truck (T). Using the definition of the sensors' recognition ability $\mathcal{RA}^S$ in Section 5.3.3, the recognition ability of the laserscanner system is:

$$\mathcal{RA}^L = \{B, C, P, T\}. \quad (7.1)$$

Thus, the class-BBA $m_{C,z}^{L,(j)}$ of the j^{th} measurement (5.12) is defined by the masses:

$$\begin{aligned} m_{C,z}^{L,(j)}(B) &= p^{L,(j)}(B|F) \\ m_{C,z}^{L,(j)}(C) &= p^{L,(j)}(C|F) \\ m_{C,z}^{L,(j)}(P) &= p^{L,(j)}(P|F) \\ m_{C,z}^{L,(j)}(T) &= p^{L,(j)}(T|F) \end{aligned}$$

In order to reduce the sensitivity to classification errors and give a mass to the combination of all distinguished classes, which avoids Zadeh's paradox, the class-BBA

Figure 7.1: Position dependent detection probability of the laserscanner system. Objects within the bright areas have a high detection probability, while the detection probability in the dark areas is close to zero.

$m_{C,z}^{L,(j)}$ is discounted. The experimentally determined discounting factor $p_z^L = 0.8$ expresses the confidence in the classification result of the laserscanner system. Since the discrimination of pedestrians and bikes as well as the road user classification in high ranges based on point cloud features is challenging, the confidence in the classification of the laserscanner system is lower than that of the camera systems described in the next sections. A summary of the parameters is provided in the first column of Table 7.1.

7.1.2 Low Resolution Camera

Three low resolution (LR) cameras are used to detect vehicles in the approaches of the intersection. For a visualization of their FOV see Figures A.2a, A.2b, and A.2c. Due to the temporal synchronization of the sensors, the LR camera system is considered as a whole. Thus, the system is modeled as one virtual sensor which consists of three cameras whose detected objects are provided in one go. Therefore, a vehicle detector based on a paper of Viola and Jones [VJ01] is used. The classifier structure guarantees high detection rates at low computational costs by combining Haar-like features with a cascaded processing scheme. To improve both detection rate and computational complexity of the detector, more advanced techniques like

Figure 7.2: Position dependent detection probability of the LR camera system. Objects within the bright areas have a high detection probability, while the detection probability in the dark areas is close to zero.

inter-stage information propagation, more sophisticated features, and an adaptive coarse-to-fine object search was added by [GLW⁺13]. For vehicle verification and assignment of the posterior class probability $p^{C_{SD},(j)}(V)$ (vehicle or background) a Local Receptive Field with a feed-forward Neural Network is trained [GLW⁺13]. To determine the pose of the vehicle detections, the planar homography between the image plane and the ground plane of the intersection coordinate system is exploited. Given the rectangular vehicle detection in image coordinates, the bottom center point of the rectangle is projected on the lane and the vehicles world orientation is defined based on the lane orientation. Since only the detections of approaching vehicles are considered, all measurements are referred to the front center (FC) RP illustrated in Figure 3.5. The detection performance was again analyzed to determine the detection probability map of the LR camera system in Figure 7.2. The bright areas mark $p_D^{C_{SD},(i)} = 0.8$ and dark areas represent again $p_D^{C_{SD},(i)} = 1e^{-4}$. Compared to the laserscanner system, the detection probability in the bright areas is lower. This is due to the fact that the vehicle detector fails in dense traffic situations, where vehicles occlude each other. This issue might be addressed by occlusion modeling which has not been implemented within this thesis. The detection range of the LR camera system is larger than that of the laserscanner system and facilitates the detection of vehicles up to approximately 80 m from the intersection center.

In contrast to the laserscanner system, the object detection in case of the LR cameras is based on vehicle (V) classification. A vehicle can be either a car (C) or a truck (T), consequently the recognition ability of the LR cameras is $\mathcal{RA}^{CSD} = \{CT\}$ and the class-BBA of the measurement is defined by:

$$\begin{aligned} m_{C,z}^{CSD,(j)}(CT) &= p^{CSD,(j)}(V) \\ m_{C,z}^{CSD,(j)}(BP) &= 1 - p^{CSD,(j)}(V). \end{aligned}$$

Similar to the class-BBAs for the laserscanner measurements, $m_{C,z}^{CSD,(j)}$ is discounted. The classification result of the LR cameras appeared to be more confident as that of the laserscanners and is set $p_z^{CSD} = 0.9$. Amongst the others, this parameter is included in Table 7.1.

7.1.3 High Definition Camera

To detect pedestrians at a neuralgic area of the intersection (see Figure A.2d), a wide angle stereo is realized with two high resolution (HR) cameras. Analog to the LR cameras, the pedestrian detection is based on classification. Exploiting histogram of oriented gradients (HOG) features in linear and Gaussian kernel support vector machines (SVMs) [WKH⁺11], the pedestrians are detected and a class probability $p^{CHD,(j)}(P)$ is provided. For the HR camera system, a constant value for $p^{CHD,(j)}(P)$ was experimentally determined. The detection results of both cameras are fused and OBBs referred to the center center (CC) RP are determined. For more details about the pedestrian detector, refer to [WKH⁺11]. The detection probability of the HR camera system is given in Figure 7.3. In the FOV of the system $p_D^{CHD,(i)} = 0.9$ and outside $p_D^{CHD,(i)} = 1e^{-4}$.

Since the HR cameras are only able to detect pedestrians (Ps), their recognition ability is $\mathcal{RA}^{CHD} = \{P\}$ and the class-BBA yields

$$\begin{aligned} m_{C,z}^{CHD,(j)}(P) &= p^{CHD,(j)}(P) \\ m_{C,z}^{CHD,(j)}(BCT) &= 1 - p^{CHD,(j)}(P). \end{aligned}$$

As for the LR cameras, the confidence of $m_{C,z}^{CHD,(j)}$ was set to $p_z^{CHD} = 0.9$. The parameters of the HR camera system are summarized in the last column of Table 7.1.

Figure 7.3: Position dependent detection probability of the HR camera system. Objects within the bright areas have a high detection probability, while the detection probability in the dark areas is close to zero.

7.2 Parameterization of the CMMPHD Filter

7.2.1 Survival Probability

To fuse the measurements gathered by the three sensor types differing in their FOVs and recognition abilities, special care in modeling the detection and survival probability is essential. While the sensor specific and position dependent detection probabilities are part of the sensor model, the survival probability has to be considered for the entire perception system. Similar to the detection probability the survival probability is position dependent. However, because it is involved in the prediction of the MCs' state, it is common for all sensors.

The survival probability map in Figure 7.4 contains the likelihoods that a currently existing object will also exist at the receiving time of the next measurement. The survival probability depends on the FOVs of the individual sensors and the locations in which objects might appear, disappear, or persist. Consequently, the survival probability map in Figure 7.4 results from the combination of the sensor individual survival probability maps, which equal in appearance the respective detection probability maps and environment specifics. The maps are combined by taking the bright

Figure 7.4: Position dependent survival probability for the objects at the intersection. Objects within the bright areas have a high survival probability, while the survival probability in the dark areas is close to zero.

map values as regions of high survival probability $p_S = 0.99$, while the remaining areas are filled with low values $p_S = 0.01$. Since the considered objects in the bright region of Figure 7.4 are not able to disappear, the choice of the p_S is straight forward. Afterwards, a high survival probability is also assigned to regions in which objects are not able to disappear, i.e. gaps in the FOV of the intersection system.

7.2.2 Road User Classification

According to the recognition abilities $\mathcal{RA}^S$ of the used sensors, the tracker's set of distinguished classes is

$$\Omega_T = \{B, C, P, T\} \quad (7.2)$$

which is denoted as the tracker's frame of discernment (FOD).

As indicated by the dotted arrow in the right part of Figure 5.4, the CMMPHD filter facilitates the incorporation of mixture component (MC) features to classify objects. In case of the distinguished road users in (7.2), the objects' maximum velocity v_{max} as well as their length l and width w showed significance. Due to the

characteristics of the intersection sensors, it is hardly possible to set up a reliable road user classification which only relies on the class-BBAs of the sensor measurements. Therefore, apart from the features of the measurements, attributes of the MCs are used. Due to the filtering, those attributes are characterized by a high robustness.

The first attribute is the maximum absolute value of the MC's velocity

$$v_{max} = \max_{t \in 1:k} \sqrt{\dot{x}_{p,t}^2 + \dot{y}_{p,t}^2} \quad (7.3)$$

during its lifetime $1 : k$. In [GGZ⁺13; JROR10; KPN96] the velocity of pedestrians and cyclists in urban environments was analyzed. In reference to those studies, no class evidence can be derived if $v_{max} < 2$ m/s. Consequently, a high mass is given to the neutral element $m(BCPT)$. Furthermore, objects exceeding $v_{max} = 2$ m/s are unlikely to be pedestrians. Summarizing the velocity studies and modeling the transitions as sigmoid function, results in the BBA masses $m(A)$ for the v_{max} feature in Figure 7.5a.

The second attribute is composed of the length l and width w represented by an MC. Since the measured length and width of a detected object is very unstable due to the sensors' viewpoints, occlusions, and patchy detections, they are not involved in the laserscanner-based road user classification introduced in Section 6.3. Albeit, estimating the objects' extent as an additional attribute of the MCs, the objects' length and width can be used to enforce the classification result. Considering that the intersection perception system is not able to detect the length and width of the objects in the approaches, the extent features are only used for objects that have already passed the stop line. Based on usual road user dimensions the BBAs for l and w are determined and depicted in Figures 7.5b and 7.5c.

Similar to the sensor dependent class-BBAs, even the confidence given to the class-BBAs of the MCs can be considered by discounting. Due to the high persistence of the road user tracking, the class-BBA based on v_{max} is integrated with a high confidence level of 0.9. On the contrary, the BBAs of the extent features are discounted with 0.7.

7.2.3 Motion Models

Considering the set of distinguished classes (7.2), it is appropriate to use two different motion models. On the one hand pedestrians are able to move sideways, their motion direction does not depend on their orientation, and are in general agile road users. Thus, the motion of pedestrians is modeled by a linear constant velocity

(a)

(b)

(c)

Figure 7.5: Masses of the class-BBA for the maximum velocity (a), length (b), and width (c) of the object.

(LCV) model introduced in Section 3.4. On the other hand bikes, cars, and trucks are not able to move laterally and feature rigid motion. Thus, their motion is modeled with a constant turn rate and velocity (CTRV) model (3.40). Consequently, the MCs' individual transition probability matrix (5.20) yields

$$T^{(i)} = \begin{pmatrix} \text{Bet}P_{m_C^{(i)}}(P) & \text{Bet}P_{m_C^{(i)}}(BCT) \\ \text{Bet}P_{m_C^{(i)}}(P) & \text{Bet}P_{m_C^{(i)}}(BCT) \end{pmatrix}. \quad (7.4)$$

In (7.4), the belief regarding the object's class on the credal level induces a motion model probability on the pignistic level. Due to the flexibility of the Dempster-Shafer theory of evidence (DST), the probability for arbitrary class combinations out of the power set 2^{Ω_T} can be determined by the pignistic transformation (5.10). Thus, the set of considered models can be modified whenever needed.

Involving the object's orientation in the LCV model, the state vectors of the two models are:

$$x^{LCV} = \begin{pmatrix} x_p \\ y_p \\ \dot{x}_p \\ \dot{y}_p \\ \phi \end{pmatrix} \quad \text{and} \quad x^{CTRV} = \begin{pmatrix} x_p \\ y_p \\ |v| \\ \phi \\ \dot{\phi} \end{pmatrix}. \quad (7.5)$$

To realize the mixing (4.73) and merging (Algorithm 1) of states and covariances of different motion models in the CMMPHD filter, the transformation of Section 3.4.1 to combine different state models is used.

Chapter 8

Evaluation

In this chapter, the performance of the proposed laserscanner-based object detection algorithms as well as the introduced CMMPHD filter are evaluated. Since the aim of the intersection perception system is to recognize and track road users at an urban intersection, the entire evaluation is based on real data gathered at the intersection described in Chapter 2.2.

After the introduction of common single and multi-object performance measures in Section 8.1, the object detection results using the DBSCAN algorithm with pathway tracking are compared with the results of the original DBSCAN algorithm. Subsequently, the road user tracking performance is evaluated in Section 8.3. Therein, the benefit of the multiple-model approach, the accuracy of estimating the cars' states and extents, the performance in multi-object scenarios, as well as the multi-sensor capability of the CMMPHD filter is addressed. Finally, in Section 8.4 the achieved accuracy in terms of road user classification is revealed.

For the evaluation two sources of ground truth data are available. On the one hand, laserscanner measurements are labeled manually to provide a ground truth for the multi-object tracking and road user classification results. On the other hand, cars are equipped with an automotive dynamic motion analyzer (ADMA) which is a real-time kinematic global positioning system (RTK-GPS) with an inertial measurement unit. Under good satellite conditions, the ADMA is able to achieve a position accuracy of 0.02 m. However, due to tall buildings and trees around the intersection, the accuracy degrades to a standard deviation of approximately 0.15 m in most of the cases. Hence, this standard deviation is used for the consistency checks in Section 8.3.3. Compared to the identified measurement uncertainties summarized in Table 7.1, the reference system is not one magnitude but at least more accurate as the involved sensors.

8.1 Performance Measures

To evaluate the result of the state estimation, performance measures are required. Usually, the performance of state estimators is evaluated using absolute estimation errors and consistency checks which are reviewed in the following.

8.1.1 Absolute Estimation Errors

The calculation of the absolute estimation error is only possible if ground truth data for the estimated states is available. In this case, the difference between the true and the estimated state at every time step k can be calculated:

$$\tilde{x}_k = x_k - \hat{x}_k. \quad (8.1)$$

To express the absolute estimation accuracy of a tracker, typically the root mean square error (RMSE) [BL93, p. 105]

$$RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^N \|\tilde{x}_k\|_2^2} \quad (8.2)$$

which averages the euclidean distance

$$\|\tilde{x}\|_2 = \sqrt{\sum_{i=1}^{N_x} \tilde{x}_i^2} \quad (8.3)$$

over N time steps is used. Since this is the formulation of the RMSE to evaluate tracking algorithms with real world data, originally the RMSE is calculated over N Monte-Carlo runs and based on simulated data. The RMSE is only meaningful if all components used in (8.2) have equal units. Thus, the accuracy of the estimated states like the position, velocity, or orientation is evaluated using a separate RMSE.

8.1.2 Optimal Subpattern Assignment Metric

The optimal subpattern assignment (OSPA) metric developed by Schuhmacher et al. [SVV08] is a metric to evaluate both the spatial as well as the cardinality errors of two sets of objects. In contrast to other multi-object distances like the optimal mass transfer (OMAT) [HM04b], the OSPA distance overcomes prevalent limitations. As

pointed out by [SVV08], one of these limitations is a lack of physical interpretation if the cardinalities of two sets differ.

Regardless of the multi-object tracking algorithm the OSPA distance between the set of true tracks $\mathbf{Y} = \{y^{(1)}, \dots, y^{(n)}\}$ and the set of tracks $\mathbf{X} = \{x^{(1)}, \dots, x^{(m)}\}$ is

$$\bar{d}_p^{(c)}(\mathbf{X}, \mathbf{Y}) := \begin{cases} \left(\frac{1}{n} \left(\min_{\pi \in \Pi_n} \sum_{i=1}^m d^{(c)}(x_i, y_{\pi(i)})^p + c^p(n-m) \right) \right)^{1/p} & \text{if } m \leq n \\ \bar{d}_p^{(c)}(\mathbf{Y}, \mathbf{X}) & \text{if } m > n \end{cases}. \quad (8.4)$$

Here the first term under the sum handles the spatial error, the second term the cardinality error, and the symbol Π_n denotes the set of permutations on $\{1, \dots, n\}$. The parameters of the OSPA metric are the cut off $c > 0$ and the order $p \in [1; \infty]$. Taking a closer look at the definition of the distance between two state vectors

$$d^{(c)}(x, y) := \min(c, d(x, y)), \quad (8.5)$$

the influence of c becomes apparent. In (8.5) an arbitrary metric $d(x, y) \in [0; \infty]$ could be used. Thus, $d^{(c)}(x, y) \in [0; c]$ penalizes spatial errors at the maximum with c . Furthermore, in combination with the second term in (8.4), the cut off parameter prevents that a high spatial distance between two state vectors has a stronger influence on the OSPA distance than a cardinality mismatch. Finally, with a high order p of the OSPA distance the penalization of outliers could be increased [SVV08].

To identify the reason for high OSPA distances or evaluate a method only with respect to its state or cardinality estimation performance, the consideration of only the localization error

$$\bar{d}_{p,loc}^{(c)}(\mathbf{X}, \mathbf{Y}) = \left(\frac{1}{n} \cdot \min_{\pi \in \Pi_n} \sum_{i=1}^m d^{(c)}(x_i, y_{\pi(i)})^p \right)^{1/p} \quad (8.6)$$

or the cardinality error

$$\bar{d}_{p,card}^{(c)}(\mathbf{X}, \mathbf{Y}) = \left(\frac{1}{n} \cdot c^p(n-m) \right)^{1/p} \quad (8.7)$$

involved in the OSPA distance may be helpful. Beside the evaluation of multi-object tracking systems, the OSPA distance (8.4) as well as the cardinality error (8.7) were used as a common measure to compare different clustering algorithms and to show the improvement between successive processing steps of multi-object perception systems [MRD12; MRD13].

In this thesis the OSPA distance between the estimated set of tracks and a manually labeled set of objects is calculated for each timestep. To avoid high OSPA distances due to cardinality errors at the border of the sensor systems' detection area (e.g. Figure 7.1), an evaluation area which encloses only the locations with maximal detection probability is used. Assuming that an object enters the detection area but is not detected immediately due to a low detection probability. In this case the cardinality error will produce a high OSPA distance and falsify the impression of the multi-object tracking performance. Since objects enter and leave the detection area continuously, the OSPA cannot be interpreted without restriction to the evaluation area. Even with the evaluation area the interpretation of the OSPA distance in scenarios with many objects entering and leaving the detection area is challenging as discussed in Section 8.3.4.

8.1.3 Filter Consistency

Another method to evaluate the performance of state estimators are consistency checks. The definition of consistency is given in [BL93]:

$$E\{\tilde{x}\} \stackrel{!}{=} 0 \quad (8.8)$$

$$E\{\tilde{x}\tilde{x}^T\} \stackrel{!}{=} \underline{P}. \quad (8.9)$$

While (8.8) claims that the expectation value of the estimation error has no bias, (8.9) describes the requirement that the estimated covariance $\underline{P}$ matches the true mean square error. Since the mean square error increases in case of an estimation bias, requirement (8.9) implicitly includes (8.8). Thus, the consistency of a state estimator assuming Gaussian distributed signals can be proved using the normalized estimation error squared (NEES) or normalized innovation squared (NIS) [BL93, p.237 ff.]. The tested tracking approach is consistent if the NEES

$$\epsilon_x = \tilde{x}^T \underline{P}^{-1} \tilde{x} \quad (8.10)$$

is χ^2 distributed with $\dim(x)$ degrees of freedom. In the majority of applications it is not possible to provide reference data for all estimated states online. Therefore, the NIS, a measurement-based counterpart to the NEES, was defined:

$$\epsilon_\gamma = \gamma^T \underline{S}^{-1} \gamma. \quad (8.11)$$

Both the measurement residuum γ (3.13) and the innovation covariance $\underline{S}$ (3.12) are calculated in every iteration of the PHD filter. Thus, the filter is consistent if ϵ_γ is between the upper and lower bound of the χ^2 distribution with $\dim(z)$ degrees of freedom in most of the cases. The bounds are determined by the required confidence

level.

8.2 Improvement of Object Detection Using Pathway Tracks

The performance of the road user perception system essentially depends on the precision of the object detection methods. As discussed in Chapter 6, the relevant objects in the FOV of the laserscanner system have to be detected and their oriented bounding boxes (OBBs) including the reference points (RPs) have to be determined correctly. To evaluate the laserscanner-based object detection and creation accuracy of the DBSCAN algorithm with and without pathway tracking, two indicators are used. First, comparing the cardinality of the detected objects with the true cardinality of objects in the sequence reveals if an object detection method tends to produce clutter measurements, missed detections, as well as over- and under-clustered objects. Second, the OSPA metric (8.4) is used to evaluate the determined object cardinality as well as the position accuracy of the OBBs reference points. A sequence of laserscanner measurements with a duration of 97.5 s containing bikers, pedestrians, cars, and trucks passing the intersection was processed using the DBSCAN algorithm with and without pathway tracking. The results in terms of cardinality and OSPA distance are shown by Figure 8.1 and are both based on manually labeled reference data. The sequence is sequence 1d of the published Ko-PER Intersection Laserscanner and Video Dataset [SMS⁺14].

For the comparison of the two object detection methods only the central area of the intersection is considered, where the objects' orientation cannot be determined based on map information (see Algorithm 3). The DBSCAN algorithm requires at least $C_{min} = 5$ points in a search radius of $\epsilon = 0.3$ m to build clusters. Thus, a robust pedestrian detection is guaranteed even if they are close to each other. Nevertheless, since even bigger objects like cars and trucks have to be detected, the clustering result includes a lot of over-clustered vehicles. This becomes apparent in the cardinality comparison in Figure 8.1a. Here the cardinality determined by the DBSCAN algorithm is persistently over the true cardinality. In contrast, using pathway tracks to correct over-clustered objects as introduced in Section 6.2.4, the cardinality of the clustering result mostly coincides with the true cardinality in Figure 8.1a. A similar impression is illustrated by the comparison of the OSPA distances in Figure 8.1b. In addition to the cardinality error, the OSPA distance involves the correctness of the reference points' positions. In this case, the DBSCAN algorithm with pathway tracking benefits from the explicit estimation of the objects' orientation which ensures appropriate RPs. As demonstrated by this example, the

(a) Cardinality comparison of the object detection results with the reference.

(b) Comparison of the OSPA metric courses with $p = 1$ and $c = 10$ m.

Figure 8.1: Comparison of the DBSCAN algorithm and the DBSCAN algorithm with pathway tracking.

OSPA metric (8.4) originally developed to evaluate multi-object trackers is also highly suitable to compare and evaluate different object detection approaches.

8.3 Road User Tracking

8.3.1 Comparison of a Single-Model and a Multiple-Model Filter

The major benefits of the developed classifying multiple-model probability hypothesis density (CMMPHD) filter in tracking multiple objects of different classes are twofold: On the one hand, the multiple-model (MM) capability enables the realistic prediction of the objects' states regarding class dependent motion characteristics. On the other hand, the integrated classification is exploited to adapt the motion model transition probabilities (5.20) of each PHD component based on its estimated class-BBA. Using an example sequence which contains one left turning car, as well as one pedestrian and one cyclist both crossing the street in opposite directions, the CMMPHD filter is compared to its single-model version, the classifying single-model probability hypothesis density (CSMPHD) filter. The color-coded temporal evolution of the laserscanner foreground measurements is illustrated by Figure 8.2.

The CMMPHD filter is parameterized according to Table 7.1 and for the CSMPHD filter an LCV model (3.37) with a standard deviation of 0.5 m is used. Due to its flexibility, the LCV model is sufficient to model both the motion of bikes, pedestrians and cars. However, the initialization takes more time since the uncertainties are designed to represent the agility of pedestrians as well as the high velocities of cars. Due to this trade-off, the initial uncertainty is too low for fast driving cars and too high for slow moving pedestrians. This drawback can be observed in the course of the OSPA distance in Figure 8.3. Here, in the time interval $[2.9s; 5.8s]$ the car enters the FOV of the laserscanner system and consequently the CMMPHD filter outperforms the CSMPHD filter during this period. Due to its high velocity of approximately 13.5 m/s, the single model filter requires 2.9 s to initialize the car track, while the class specific motion models enable the CMMPHD filter to initialize the track just before the car enters the evaluation area of the OSPA distance. The long initialization time of the car track in case of the single model filter is due to the high state uncertainty of the car which enters the evaluation area with a high velocity.

In the remaining part of the sequence, the OSPA distance is below 0.5 m which is an excellent mean position error for tracking extended objects during maneuvers.

Figure 8.2: Evolution of background subtracted laserscanner measurements in time. It shows the measurements originated by a left turning car, a cyclist, and a pedestrian.

Figure 8.3: OSPA distance of a multiple model (CMMPHD) and a single model (CSMPHD) filter with $p = 1$ and $c = 10$ m.

This is only one example in which the CMMPHD filter outperforms its single-model counterpart. Further benefits are a more sufficient track prediction and trajectory estimation which are essential to predict driving maneuvers for ADAS.

8.3.2 Tracking Results for Representative Maneuvers

To give an impression of the achieved tracking accuracy, representative maneuvers were chosen. By means of those example maneuvers the performance of the CMMPHD filter in terms of the position, orientation, velocity, and extent estimation is discussed in detail. The measurements were gathered during night time, to avoid disturbances by other road users and all involved cars were equipped with the reference system introduced in the introduction of Chapter 8.

The first scenario is the one depicted by Figure 8.2. Since the overall tracking performance was already discussed by means of Figure 8.3, now the focus is shifted to the accuracy of position and extent estimation. The tracking results based on the laserscanner measurements of Figure 8.2 are illustrated by Figure 8.4. The estimated oriented bounding boxes (OBBs) of the car, the pedestrian, and the cyclist are depicted in blue, magenta, and cyan respectively.

Since Figure 8.4 contains a lot of information, each object is discussed separately. For the car, the estimated OBBs as well as the temporal corresponding true OBBs are depicted in blue and green. Considering that the laserscanner system is neither able to detect the entire width nor the length of the car in the approaches, an accurate position estimation during the approaching phase is only possible because of the used RP model introduced in Section 3.4.2. In the approaches the cars' states are referred to the front center (FC) RP which results in a highly accurate match of the estimated and true front of the cars even in high distances. In the central area of the intersection, the estimated OBBs nearly perfectly match the reference OBBs. That is because of the high sensor coverage in this area which enables the perception of the car's entire extent. Finally, the classification is continuously correct which is illustrated by the correspondence of the edge (true class) and fill colors (estimated class).

For the bike and the pedestrian tracks no reference data is available. However, Figure 8.4 depicts that both objects are tracked continuously during their way through the intersection, even when they closely pass each other. While the pedestrian is classified correctly all the time, the bike is confused with a pedestrian at the beginning of the sequence. Knowing that the bike just starts to the beginning of the sequence, a velocity-based classification according to the BBA illustrated by Figure 7.5a is not possible in this time slot. Considering also that the point clouds of pedestrians and

Figure 8.4: Tracking results of the CMMPHD filter including the estimated OBBs. For the car reference positions and OBBs are depicted in green. The edge color of the OBBs indicates the true object class and the filling the estimated class. For the car only every 13th and for the pedestrian and bike every 7th OBB is plotted to keep the figure clear.

bikes are often too similar to realize a discrimination, the class confusion can be explained. Nevertheless, reaching higher velocities, the bike is classified correctly. Consequently, the example points out the importance of the integrated classification in the CMMPHD filter to realize a reliable classification of road users.

In the next sequence, a car approaches the intersection, stops at approximately $(-18m, 20m)$, starts again, and turns right. To get an impression for the input data, the evolution of the laserscanner measurements is given by Figure 8.5. Additionally, the tracking results in terms of position, orientation, and extent are shown by Figure 8.6. The car is continuously tracked from approximately 61 m before the stop line until the egress of the intersection. Similar to the preceding maneuver, in the center of the intersection the estimated OBBs match the OBBs of the reference almost perfectly. In the approach the visible part of the car is referred to the correct RP which results in a very accurate position estimation up to high ranges. This impression is confirmed by the RMSE of the position, orientation, and velocity in Figure 8.10. In interpreting the RMSE plots, it has to be considered that for their calculation the estimated and the reference states were transformed to a common RP. Since for the RP transformation the correct orientation and extent of the car is required, the reference state is transformed to the RP of the track and not vice versa. As mentioned before, in the approach the car's state is referred to the front center (FC) RP which compensates the missing length information. However, since

Figure 8.5: Evolution of background subtracted laserscanner measurements over time. It shows the measurements originated by a right turning car.

In general, the objects in the approaches originate only a few measurement points, which can be seen in Figures 8.5 and 8.8, the determination of the objects' width and for this reason the lateral determination of the FC RP is uncertain which directly influences the accuracy of the estimated car's state. Thus, the position error shrinks with the distance to the intersection and finally floats around 0.4 m. The velocity and orientation errors show the same behaviour but slightly raise again during the right turning maneuver. In summary, the RMSEs are low, especially for tracking maneuvering extended targets.

After a left and a right turning car, a straight ahead driving maneuver is investigated. The foreground measurements of the laserscanner system are depicted by Figure 8.8 and the tracking results by Figure 8.9. Also in this case the positions as well as the OBBs of the car are precisely estimated. Even in the approach the position error (Figure 8.10) is below 0.4 m which is due to the fortunate circumstances that the center part of the car's front originates the measurements during the approaching phase. Consequently, the uncertain width determination does not falsify the state

Figure 8.7: RMSEs between the car's estimated position, velocity, and yaw angle and their reference for the maneuver depicted in Figure 8.6.

Figure 8.8: Evolution of background subtracted laserscanner measurements over time. It shows the measurements originated by a car passing the intersection.

estimation in this scenario which results in the overall low position, velocity and orientation errors shown by Figure 8.10.

8.3.3 Accuracy of Cars' State Estimation

For the evaluation of the state estimation accuracy of cars, the RMSE and the consistency check introduced in Section 8.1 are exploited. The evaluation is based on 30 driving maneuvers which cover all permitted driving routes through the intersection. For each maneuver highly accurate reference data is available. Due to the restricted GPS reception at the intersection, only 30 reference position sequences out of over 300 reached the required standard deviation of 0.15 m. Beside the uncertainty and the quality of the GPS signal provided by the ADMA, the validity of the reference position was double-checked with the highly accurate lane markings map (see Figure 6.1a). Only reference trajectories running in accordance with the

Figure 8.9: Tracking results of the CMMPHD filter including the estimated OBBs of a car driving straight across the intersection. Reference positions and OBBs are depicted in green. To keep the Figure clear, only every 13th OBB is illustrated.

Figure 8.10: RMSEs between the car's estimated position, velocity, and yaw angle and their reference for the maneuver depicted in Figure 8.9.

Table 8.1: RMSEs over 30 Driving Maneuvers

$RMSE_{pos}$ [m]	$RMSE_{vel}$ [m/s]	$RMSE_{yaw}$ [°]
0.39	0.42	1.71

driving lanes in the map were selected for evaluation.

Similar to that for the representative maneuvers in Section 8.3.2, the RMSE for the position, the velocity, and the orientation is calculated. The RMSEs over all considered maneuvers are summarized in Table 8.1. The low residues confirm the excellent results discussed in the previous section.

To complete the evaluation of the state accuracy, Figure 8.11 shows box plots, also called Box-Whisker-Plots, which illustrate various statistical distribution and range measures of data. Beside the median of the RMSEs (red horizontal line), a box is shown which starts at the 0.25-quartile and ends at the 0.75-quartile. Consequently, the length of the box corresponds to the interquartile range (IQR). The extremal points, not considered as outliers, are marked by the end of the dashed lines, the so called Whiskers. The Whiskers are 1.5 IQRs apart from the quartiles. All points with a higher distance to the quartiles are outliers and marked by red crosses. With the box plots the good impression of the state estimation accuracy is completed, since the majority of position estimates is better than 0.5 m and the maximum error is at 0.92 m. Similar results were achieved for the velocity, which is mostly estimated with an error lower 0.6 m/s and a maximum of 1.32 m/s. Additionally, stating that 75 % of the yaw angle RMSEs are smaller than 2.0° and a maximum error of 4.1°, even the precision of the yaw angle estimation is remarkable for extended object tracking.

Figure 8.11: Box plots of the RMSE in terms of position, velocity, and yaw angle.

Since an accurate estimation of the cars' state is only one side of the medal, the estimated uncertainties are evaluated following the consistency check introduced in Section 8.1.3. Due to the availability of ground truth data for the states, the NEES (8.10) with respect to the position, velocity, and yaw angle is utilized for evaluation. In Figure 8.12 the χ^2 distribution with 4 degrees of freedom is plotted in red, which represents the expected distribution of the NEES values for filter consistency. The histogram of the NEES values, visualized in blue, is similar to the χ^2 distribution which shows that the filter is overall consistent. Using a significance level of 0.05, a NEES value has to lie within the 95 % interval $[0.48; 11.14]$ of the χ^2 distribution with 4 degrees of freedom to represent a consistent state estimation result. The ratio of over estimated ($\text{NEES} < 0.48$), consistent ($0.48 \leq \text{NEES} \leq 11.14$), and under estimated ($\text{NEES} > 11.14$) uncertainties is summarized by the bar plot in Figure 8.13. It states that nearly 90 % of the time the filter is consistent and the reasons for inconsistency equally result from over or under estimated uncertainties. This is an usual behaviour of a filter which models multiple motion states of different object types with one motion model. In this thesis the constant turn rate and velocity (CTRV) model is used for cars. The CTRV assumption is violated during turning, breaking and accelerating maneuvers of cars. Hence, a trade-off between high model uncertainties to prevent losing car tracks during unmodeled maneuvers and low model uncertainties during modeled maneuvers is required. This may result in both under and over estimated uncertainties. Further, in the CMMPHD filter the motion of bikes, cars, and trucks is modeled with a single CTRV model. Consequently, the model parameters have to be optimized to represent the motion characteristics of all three object types. This may also lead to consistency violations. The easiest way to achieve a consistent tracking is to increase the uncertainties. However, in multi-object PHD tracking this will lead to track losses, ID switches and long track initialization times. A more promising approach to improve the consistency of the CMMPHD filter in tracking road users would be to increase the set of motion models from two to four, so the motion of every object type is predicted with its individual motion model or even introduce state dependent motion models. Nevertheless, this would significantly increase the computational complexity and opposes the realization of a real time capable intersection perception system.

8.3.4 Laserscanner-Based Multi-Object Tracking Performance

The multi-object tracking performance is evaluated with the OSPA metric (8.4). Figure 8.15a shows the OSPA distance of sequence 1a published in [SMS⁺14] and Figure 8.15b gives the objects' ground truth cardinality. In this sequence, various cars and pedestrians cross the intersection on different routes and even one truck is included. As visualized by Figure 8.15b, a maximum of 18 road users is perceived

Figure 8.12: Histogram of NEES values (blue) and the respective χ^2 distribution with 4 degrees of freedom (red).

Figure 8.13: Ration of over estimated ($\text{NEES} < 0.48$), consistent ($0.48 \leq \text{NEES} \leq 11.14$), and under estimated ($\text{NEES} > 11.14$) uncertainties.

simultaneously by the intersection system and the entire sequence is processed in real time with a cycle time of 80 ms. The OSPA distance is calculated for the error in (x, y) -space and the manually labeled object data included in the data set [SMS⁺14] is used as ground truth. Since in this sequence two pedestrians hold hands while they cross the intersection and the laserscanner system is consequently not able to separate these pedestrians, their labels provided in [SMS⁺14] are substituted by one label for both pedestrians. This prevents a false impression of the tracking performance due to a cardinality error which can not be solved by the laserscanner system. The relevant pedestrians are shown in the right part of Figure 8.14.

Figure 8.14: Projection of foreground measurements of the laserscanner system (orange) in a camera image.

Conspicuous are the peaks of the OSPA distance in Figure 8.15a. These peaks are mainly due to cardinality errors which arise during traffic phases in which various road users enter and leave the FOV of the perception system. Consequently, they appear at volatile periods of the true cardinality illustrated by Figure 8.15b. The reason for the cardinality errors are delays in track initialization or early track termination. Both root causes may appear if objects are occluded at the edge of the systems FOV or if objects feature poor reflection characteristics. However, in the majority of cases, the OSPA distance is lower than 0.5 m which corresponds to the identified RMSE of the position in Table 8.1. Thus, the developed intersection perception system is able to track a multitude of arbitrary road users continuously and highly accurate on their way through the intersection. Additionally, the position histories (red) in Figure 8.16a confirm the high persistence of the road user tracks

(a) OSPA course with $p = 1$ and $c = 10$ m.

(b) Ground truth for object cardinality.

Figure 8.15: OSPA course and ground truth cardinality for evaluation sequence.

which are initialized far in the intersection approaches and not lost until they reached an egress of the intersection. The reliable and continuous road user tracking is due to both a sensitive object detection and robust CMMPHD filter design. Further, Figure 8.16b illustrates that even huge objects like trucks are tracked reliably and both Figures state a highly accurate road user localization and classification.

8.3.5 Benefit of Multi-Sensor Fusion

To point out the benefit of multi-sensor fusion in general and the proposed multi-sensor CMMPHD filter, an example with pedestrians in the neuralgic area of the intersection (see Figure A.2d) is used. In this sequence three pedestrians move in this area and pass each other closely. The positions of the pedestrians over time are illustrated by Figure 8.17. As already mentioned in the previous section, the

(a) CMMPHD filter sets up tracks at high distances and tracks all road users continuously through the intersection.

(b) CMMPHD filter correctly estimates the dimension and class of trucks.

Figure 8.16: Snapshot of the tracking result. The color of the tracks represents the objects' class: cars (blue), pedestrians (green), bikes (cyan), trucks (yellow). In red the history of the objects' position is illustrated.

laserscanner system is not able to separate closely spaced pedestrians. In contrast, the HR camera system introduced in Section 7.1.3 is based on pedestrian classification and facilitates the recognition of nearby pedestrians. Therefore, the result of the laserscanner-based CMMPHD filter is compared with the result of the laserscanner and HR camera-based multi-sensor CMMPHD filter in Figure 8.18. While the single-sensor approach suffers from track losses and as a consequence from cardinality errors, the multi-sensor approach facilitates the continuous tracking of all three pedestrians through the entire sequence. Due to the fact that the laserscanner and the HR camera system feature different FOVs and recognition abilities, the multi-sensor extensions of the CMMPHD filter discussed in Section 5.3 are the key to the improved pedestrian tracking. Here the FOV of the HR camera system is completely enclosed by the FOV of the laserscanner system. Therefore, an accurate model of the detection probability as well as the recognition ability of the involved sensors is required to achieve the tracking result shown in Figure 8.18. Otherwise, the fusion of complementary sensors with different FOV providing contradicting measurements cannot lead to a correct perception result.

Analogously, the vehicle detections of the LR cameras can be fused to detect vehicles earlier and improve the object classification. Albeit, in dense traffic situations missing occlusion handling for the LR camera system, already addressed in Section 7.1.2, prevents a reliable fusion of laserscanner and LR camera detections. Thus, it is not considered in the evaluation.

8.4 Road User Classification Performance

Apart from the visual inspection of the classification correctness in Figure 8.4, the class confusion matrix (CCM) is suitable to evaluate the classification performance in a huge data set. The evaluation is based on a set of over 35.000 labeled bikes, cars, pedestrians, and trucks which are all tracked and classified by the CMMPHD filter. The CCM depicted by Figure 8.19 summarizes the performance of the classification which is based on the point cloud features introduced in Section 6.3 and the class-BBAs of Figure 7.5. The probabilities on the main diagonal express the class specific probability of correctly classified road users while the other entries stand for the confusion probabilities of the respective classes. The high classification accuracy, which is in case of pedestrians, cars, trucks at least 96 % can be explained with the skillful combination of reasonable point cloud (Section 6.3) and track-features (Section 7.2.3). While the point cloud feature based class-BBA (6.6) achieves a reliable distinction of small road users, e.g. bikes and pedestrians, and huge road users, e.g. cars and trucks, the track-features facilitate the refinement of the classification. Especially the distinction of pedestrians and bikes is challenging, since their point

Figure 8.17: Evolution of pedestrians' position over time. In the region around $x = -3$ m and $y = 0$ m they come close to each other.

Figure 8.18: OSPA course of the single-sensor and multi-sensor CMMPHD filter in tracking three pedestrians whose positions are shown by Figure 8.17

	Bike	Car	Pedestrian	Truck
Bike	0.8000	0.0227	0.1773	0.0000
Car	0.0289	0.9694	0.0013	0.0004
Pedestrian	0.0020	0.0073	0.9907	0.0000
Truck	0.0000	0.0015	0.0000	0.9985

Figure 8.19: CCM for road user classification. The entries indicate the class specific probability of correctly classified road users. The rows represent the true classes and the columns the estimated classes.

clouds are very similar. In this connection, the v_{max} (Figure 7.5a) is a strong feature to improve the bike and pedestrian classification. Nevertheless, bikes can only be classified correctly in 80 % of all cases, since the feature has no effect in cases of stationary or slow moving bikes. This leads to the probability of approx. 18 % for confusing bikes with pedestrians.

8.5 Conclusion

The evaluation illustrates that based on a reliable road user recognition, the CMM-PHD filter achieves highly accurate tracking results. In case of the laserscanner-based object detection and OBB generation, the proposed combination of 3D point clusters and pathway tracks significantly improved the cardinality and orientation determination.

Providing a set of these classified OBBs to the CMMPHD filter, the number of objects, their states, dimensions, and classes are estimated reliably. The benefit of

the integrated classification based on the DST, which is a novelty in the field of PHD filters, was shown in three cases: First, in Section 8.3.1 the class dependent multiple-model (MM) capability of the CMMPHD filter facilitates the sufficient motion modeling of different object types. Thus, track initialization times and estimation errors are reduced. Second, the introduction of sensor specific object recognition abilities $\mathcal{RA}^S$ in combination with the integrated object classification enables the fusion of sensor systems with different FOVs and recognition abilities, as demonstrated in Section 8.3.5. Third, with the integrated classification even the distinction of bikes and pedestrians and an improvement of the overall classification is achieved. In all three described cases, the representation of the objects' classes with class-BBAs according to DST enables the fusion of different even undecided sources of classification and the deferral of the decision upon a concrete class to the very end, where the tracks are extracted out of the PHD. As shown by means of the CCM, the classification performance is at least 96 % for all classes except bikes which are confused with pedestrians if they move slow.

In Sections 8.3.2, 8.3.3, and 8.3.4 the results state a reliable, persistent and highly accurate cardinality and state estimation of maneuvering extended objects without exception. All results were generated in real-time (80 ms cycle-time) and are based on real sensor and reference data from the multi-sensor intersection perception system described in Chapter 2.2.

Chapter 9

Conclusion and Future Work

9.1 Conclusion

Particularly in urban areas the raising volume of traffic and people places a demand on coming driver assistance systems to cope with complex driving situations, a huge variety of road user types, and high traffic densities which implicate highly restricted views for the driver and in-vehicle sensors. Consequently, upcoming advanced driver assistance systems (ADAS) require the combination of complementary sensor systems, to recognize the different objects in the vehicles' surrounding and achieve the robustness required for highly automatic driving. However, even with an optimal setup of in-vehicle sensors the risk of occlusions in dense traffic cannot be resolved. One possible solution is to share environment information gathered by vehicles or road side systems with other vehicles to complement locally missing pieces of information about the current traffic situation.

In this thesis an intersection-based perception system creating a dynamic representation of the current traffic situation at a public urban intersection is presented. The system fuses object detections of spatially distributed laserscanners and video cameras which are complementary in their FOVs and recognition abilities. For the laserscanners highly robust and innovative object recognition algorithms are developed. One novelty is the application of the mixture of Gaussians (MoG) segmentation algorithm, which is very popular in image processing, to segment laserscanner measurements originated from static or dynamic objects. Subsequently, the measurements representing dynamic objects are clustered by exploiting both the highly accurate range measuring capability of the laserscanners in 3D as well as 2D pathways appearing after projecting and cumulating the laser measurements on a 2D grid. Consequently, the proposed laserscanner-based object recognition method can detect objects even if they are only partially visible or represented by patchy measurements. It is shown that this approach significantly outperforms an

established 3D clustering algorithm.

The key contribution of this thesis is the classifying multiple-model probability hypothesis density (CMMPHD) filter which is a general-purpose object tracking algorithm to estimate the states and classes of multiple objects of differing types which are observed by an arbitrary number of sensors with different FOVs and recognition abilities. The filter is a computationally inexpensive approximation of the multi-object Bayes filter which avoids explicit data associations by averaging over all possible associations. To realize a robust classification of the objects between bikes, cars, pedestrians, and trucks, features of the measurements as well as dynamic features of the tracks, here mixture components (MCs), are used. Therefore, a BBA modeling feasible object classes is added to each MC. Thus, even measurement-based object classification results can be integrated without the need of an explicit measurement-to-track association. Obviously, a car features other motion characteristics than a pedestrian. To predict the motion of the objects with a sufficient model, the CMMPHD filter considers multiple motion models which are weighted according to the estimated object class probability. Hence, an MC whose class-BBA supports the class car is predicted with a motion model which is sufficient for cars and so forth. It is shown that the multiple model approach has an advantage over a single-model filter during track initialization and in the estimation of realistic trajectories. The representation of the objects' classes as BBAs, the consideration of the sensors' recognition abilities, and the position dependent modeling of the sensors' detection probabilities facilitate the fusion of multiple classifying sensors with different characteristics using DST calculus. The enhancement in terms of track continuity and accuracy is demonstrated.

Beside the illustration of the tracking improvements achieved by the CMMPHD filter compared to state-of-the-art multi-object trackers, the general tracking performance of the filter is evaluated in detail. Both during representative maneuvers of single vehicles and during realistic traffic situations in which the intersection is crossed by a multitude of road users on various routes, the CMMPHD filter estimates the states and their uncertainties precisely. Also the classification accuracy is sufficient to realize advanced driver assistance systems (ADAS) for preventive safety at urban intersections. Additionally, the performance of the intersection perception algorithms was demonstrated during the final demonstration of the research initiative Ko-FAS. In the demonstrations of the joint project Ko-PER almost all cooperative driver assistances relied on the information about the dynamic objects at the intersection provided by the CMMPHD filter.

9.2 Future Work

During the ongoing work, some aspects turned out to require further investigations which are not considered in this thesis. These are summarized in the following and may be a starting point for further research.

The first aspect concerns the multi-object tracking algorithm. Recently, the δ -generalized labeled multi-Bernoulli (δ -GLMB) filter [VV13b] which is an analytic implementation of the multi-object Bayes filter was proposed. Whereas the prediction of the δ -GLMB filter is combinatorial, the LMB filter published in [RVVD14b] approximates the posterior δ -GLMB distribution by an LMB RFS which enables a prediction with linear complexity. Additionally, in [RVVD14a] the real-time capability of a multi-sensor LMB filter is demonstrated by tracking multiple vehicles. Beside the fact that the LMB filter significantly outperforms the PHD filter [Reu14], the LMB filter features an integrated estimation of the tracks' existence probability. To the knowledge of the author, currently no multiple-model (MM) version of the LMB filter is published. So this might be a promising research topic to achieve further performance gains.

Road user tracking always includes the issue of extended object tracking. Usually and also in this thesis extended object models are used which refer the states of an object to a reference point (RP). Using this model, filters based on the point target assumption are sufficient to track extended objects. Nevertheless, extended object models work well in the majority of cases, problems occur if the perception system is not able to correctly determine the currently observed RP. In case of static sensor systems considered in this thesis, this is not a big issue, but in case of moving sensor platforms like in-vehicle perception systems the risk of confusing RPs is high. Therefore, the author contributed to [SGM⁺14] and [GRMS14], in which the Gamma-Gaussian-Inverse Wishart (GGIW) PHD filter which estimates an object's kinematic state, its extent and the number of measurements that it generates [GO12a; GO12b] is applied. Since the GGIW PHD filter directly processes the laserscanner measurements, parameterization dependent clustering algorithms as well as error prone extended object models are no longer required. Although, until now no real-time implementation for a sizeable number of objects is available, the good results in [GRMS14; SGM⁺14] indicate that further research would be worthwhile.

Finally, the issues in fusing video-based vehicle detections in dense traffic are addressed. In dense traffic a vehicle approaching the intersection behind another vehicle may be occluded in the camera image. Thus, the camera is not able to detect the occluded vehicles. Consequently, a static and position dependent modeling of the detection probability based on offline analysis is not sufficient. Therefore, an

explicit occlusion modeling as proposed by [LCB12; RD11] is required to fuse the video-based detections in dense traffic situations within the multi-sensor CMMPHD filter. This leads to a state dependent detection probability which can only be used in an approximate manner in the PHD filter (see Section 4.3.5). Since the update step of the LMB filter allows state dependent detection probabilities, this is a further reason for the development of a classifying multiple model labeled multi-Bernoulli (CMMLMB) filter.

Appendix A

Sensor Planning for Intersection Sensors

Perceiving road users at a major urban intersection with spatially distributed sensors mounted at least 4 m above the road level to realize an almost occlusion free bird's-eye view and prevent vandalism, the optimal mounting positions can not be determined by trial-and-error. Hence, a sensor planning framework was developed which facilitates the simulation of laserscanner and camera data as well as their field of view (FOV). Prerequisites for an accurate and realistic sensor planning are: first, an highly accurate environment model in terms of its topology and geometry. Therefore, the within the Ko-PER project created lane markings map (Figure 6.1a), a surveying map provided by the land surveying office of Aschaffenburg, and manually measured distances and heights of buildings were used to determine the 3D model of the intersection's environment depicted by Figure 2.1 and Figure A.1. With this model, feasible mounting positions were determined and architectural conditions were considered. Second, the sensors' measurement characteristics are represented by sensor models. A lasersanner is modeled by ray tracing. According to the laserscanner's characteristics (Table 2.2), a ray or beam is emitted in each possible direction and the distance between the sensor and the reflection point on the 3D environment model is calculated. Regarding only the four extremal beam directions, the reflection points span the FOV of a laserscanner as illustrated in Figure A.1. The cameras' FOVs (Figure A.2) are also spanned by four reflection points. In contrast to the laserscanner, the direction of the emitted rays is defined by the boundaries of the cameras' vertical and horizontal opening angle. The cameras' image is simulated by using the transformation (2.2) which considers the intrinsic and extrinsic camera parameters.

Adjusting the sensors' pose manually, suitable mounting positions and orientations of the sensors regarding occlusions robustness, high spatial resolution, and complete sensor coverage were determined. Additionally, in [Dag12] methods to determine the

optimal sensor poses were integrated in the simulation framework and investigated. However, due to the complexity of the optimization problem further research has to take place to plan the sensor poses automatically. Thus, the sensor planning framework was the key to realize the intersection perception system, introduced in Chapter 2.2.

(a) central widespread

(b) three approaching lanes of the major road on one side

(c) three approaching lanes of the major road on the other side

(d) two approaching lanes of the smaller road

(e) sidewalks

Figure A.1: Simulated FOVs of the laserscanners (blue boxes) at the intersection.

(a) LR camera observing the first approach of the major road.

(b) LR camera observing the second approach of the major road.

(c) LR camera observing the approach of the smaller road.

(d) HR cameras observing the neuralgic area.

Figure A.2: Simulated FOVs of the LR and HR cameras at the intersection.

Appendix B

Sensor Calibration

The spatial calibration of the distributed intersection sensors can be divided in three succeeding steps: First, the calibration of each individual sensor network, i.e. the laserscanners, LR cameras, and HR cameras to a common network specific coordinate system. Second, the cross-calibration of the networks to an intersection coordinate system C_I and third, the transformation of this C_I system to the ENU system whose origin is known in the world geodetic system 1984 (WGS 84) for global association.

For the calibration of the laserscanner and the camera networks conducted in the first calibration step, sensor specific characteristics are exploited. Thus, in the field of camera calibration the checkerboard is an established pattern to perform the intrinsic and extrinsic camera calibration. To determine the parameters of the camera intrinsic matrix (2.2), the method introduced by [Bou13; Zha00] was used successfully. For the extrinsic calibration which aims the identification of the pose parameters also a checkerboard pattern was exploited. Finding the edges of the checkerboard in different cameras and knowing the association between these edges, the pose parameters can be determined. Considering a multitude of different locations and sizes of the pattern at the intersection, the extrinsic parameters were determined by the minimization of the retrojection error of all checkerboards in every view [GSM⁺12; SMD13]. As motivated in Chapter 2.2, beside the two HR cameras only the three LR cameras observing the intersection approaches are used in this thesis. However, at the intersection additional four cameras are installed to observe the intersection widespread [GSM⁺12]. Hence, the LR camera network features large areas with redundant FOVs which facilitates the just described extrinsic calibration by matching a pattern observed by different cameras.

In contrast, the FOVs of the laserscanners, especially in the approaches do not overlap in such large areas and the laserscanner is not able to detect texture patterns. Thus, a calibration object (Figure B.1a) was designed whose geometry facilitates the determination of the transformation between sensors scanning this object. Knowing

(a) Calibration object for the spatial calibration of the laserscanners and the cross calibration of laserscanners and cameras.

(b) Calibration object of Figure B.1a sketched in orange and laser measurements (black) of one laserscanner originated from the street and the calibration object.

Figure B.1: Calibration object for spatial calibration of intersection sensors.

the transformation between the sensors with neighbouring FOVs, the calibration of the entire laserscanner network to a common coordinate system by chaining the inter-sensor transformations is straight forward. The entire laserscanner calibration method is described in [MD10].

With the checkerboard pattern on the top of the laserscanner calibration object, the setup is suitable to cross-calibrate the camera networks with the laserscanner network. Since the locations of all checkerboard edges on the calibration object are known (Figure B.1a) and the location of the scan points on the calibration object can be determined due to its geometry (Figure B.1b), the transformation between the laserscanner network's, the LR camera network's, and the HR cameras' coordinate systems can be calculated. An impression of the accuracy of the spatial calibration is given by the image of Figure B.3, where the foreground laserscanner measurements are projected into the camera image. More details and an accuracy evaluation is provided in [GSM⁺12].

Knowing the transformation between all intersection sensors, the remaining step is to determine the transformation in the WGS 84 to provide the road users' states in global coordinates. Therefore, the lane markings map in Figure B.2 is exploited. Since the positions of the lane markings are referred to the origin of the map which is given in global coordinates, the lane markings in the camera images facilitate the identification of the transformation between the coordinate system of the camera network and the global WGS 84. Making use of the cross-calibration, the entire intersection perception system is given in WGS 84.

Figure B.2: Lane markings map used for spatial calibration.

(a) Illustration of point cloud representation of cars, pedestrians, a cyclist, and a bus.

(b) Illustration of reduced laser reflectivity of black cars. Compared to the white car at the bottom of Figure B.3a the black car at the same location in this Figure originates significantly less measurement points.

Figure B.3: Projection of foreground measurements of the laserscanner network (orange) in a camera image illustrates the accuracy of the cross-calibration. Further, a comparison of a white and a black car's reflection characteristics is given. Obviously, the white car in Figure B.3a originates significantly more laserscanner measurements than the black car located at the same position in Figure B.3b.

Acronyms

ABS	anti-lock breaking system	1
ACC	adaptive cruise control	1
ADAS	advanced driver assistance systems	1
ADMA	automotive dynamic motion analyzer	113
B	bike	95
BBA	basic belief assignment	66
BBM	basic belief mass	65
C	car	95
CA	constant acceleration	23
CCM	class confusion matrix	135
CICAS-SSA	cooperative intersection collision avoidance systems-stop sign assist	12
CMMLMB	classifying multiple model labeled multi-Bernoulli	142
CMPMPHD	classifying multiple-model probability hypothesis density	64
CP	constant position	23
CPHD	cardinalized probability hypothesis density	39
CSMPHD	classifying single-model probability hypothesis density	119
CTRV	constant turn rate and velocity	29
CV	constant velocity	23, 29
DBGridSCAN	density based grid spatial clustering for applications with noise	88
DBSCAN	density based spatial clustering for applications with noise	83
δ -GLMB	δ -generalized labeled multi-Bernoulli	141
DST	Dempster-Shafer theory of evidence	65
EKF	extended Kalman filter	20
ENU	east-north-up	11
ESC	electronic stability control	1

FISST	finite set statistics	34
FOD	frame of discernment	65
FOV	field of view	1, 7, 143
GGIW	Gamma-Gaussian-Inverse Wishart	141
GLMB	generalized labeled multi-Bernoulli	39
GM	Gaussian mixture	43, 49, 85
GM-CMMPHD	Gaussian mixture classifying multiple-model probability hypothesis density	69
GM-MMPHD	Gaussian mixture multiple-model probability hypothesis density	59
GM-PHD	Gaussian mixture probability hypothesis density	48
GNN	global nearest neighbor	27
GPS	global positioning system	11
HOG	histogram of oriented gradients	106
HR	high resolution	8, 10, 106
I2V	infrastructure-to-vehicle	5
IMM	interacting multiple-model	23
IMMPHD	interacting multiple-model PHD	60
IPDA	integrated probabilistic data association	28
IQR	interquartile range	129
JIPDA	joint integrated probabilistic data association	28
JMS	jump Markov system	23, 57
JPDA	joint probabilistic data association	28
Ko-FAS	cooperative driver assistance systems	5
Ko-PER	cooperative perception	6
LCV	linear constant velocity	29, 92
LIDAR	light detection and ranging	8
LMB	labeled multi-Bernoulli	39
LR	low resolution	8, 10, 104
MC	mixture component	49, 69
MHD	Mahalanobis distance	26
MHT	multi-hypothesis tracking	28
MM	multiple-model	23, 31

MMCPHD	multiple-model cardinalized probability hypothesis density	72
MMPHD	multiple-model probability hypothesis density	57
MoG	mixture of Gaussians	85, 139
NEES	normalized estimation error squared	116
NIS	normalized innovation squared	24
NN	nearest neighbor	27
OBB	oriented bounding box	32, 94, 121
OMAT	optimal mass transfer	114
OSPA	optimal subpattern assignment	114
P	pedestrian	95
PDA	probabilistic data association	28
PDF	probability density function	16
PG	pathway grid	90
PHD	probability hypothesis density	33, 43, 47
PM-PHD	product multi-sensor probability hypothesis density	77
PT	pathway track	83, 92
RFS	random finite set	34
RMSE	root mean square error	114
ROI	region of interest	82
RP	reference point	32
RTK-GPS	real-time kinematic global positioning system	113
SMC	sequential Monte-Carlo	39
SVD	singular value decomposition	94
SVM	support vector machine	106
T	truck	95
TBM	transferable belief model	68
UKF	unscented Kalman filter	21
UT	unscented transformation	21
UTC	universal time, coordinated	11
V	vehicle	106
V2V	vehicle-to-vehicle	5

WGS 84	world geodetic system 1984	147
--------	----------------------------------	-----

List of Symbols

Notations

a	vector or scalar value
a^T	transpose of a vector
$\underline{A}$	matrix
$\underline{A}^{-1}$	inverse of a matrix
$\underline{A}^J$	Jacobian of a matrix
$\underline{A}^T$	transpose of a matrix
A	set
$\mathbf{A}$	random finite set
$ a $	dimension of a vector
$ \mathbf{A} $	cardinality of a random finite set

Subscripts

k	time step k
$+$	predicted value of a variable or density
B	variable or density corresponding to a new born object
C	variable or density corresponding to clutter process
S	variable or density corresponding to a surviving object

Scalars and vectors

γ	residual of predicted and actual measurement
λ	expected value of the Poisson distribution
μ	IMM mixing probability
o	motion model
χ	sigma points of the unscented transformation
x	state vector
$\hat{x}$	expected value of the spatial distribution $p(x)$

Matrices

$\underline{A}$	assignment matrix
$\underline{F}$	process (or state transition) matrix
$\underline{H}$	measurement matrix
$\underline{I}$	identity matrix
$\underline{K}$	Kalman gain
$\underline{P}$	estimation error covariance matrix
$\underline{Q}$	covariance matrix of the process noise
$\underline{R}$	covariance matrix of the measurement noise
$\underline{S}$	innovation covariance matrix
$\underline{T}$	motion model transition probability matrix

Miscellaneous functions

f	process (or state transition) function
h	measurement function
d_{MHD}	Mahalanobis distance (MHD)

Densities

p	spatial distribution p
t	model transition density t
π	multi-object probability density
$c(z)$	probability density of clutter measurements
$\delta_{\mathbf{Y}}$	multi-object Dirac delta probability distribution centered at $\mathbf{Y}$
$f(x_+ \cdot)$	Markov transition density
$\kappa(z)$	probability hypothesis density of the clutter process
$v(x)$	probability hypothesis density (equivalent to intensity density)
$\mathcal{N}(x; \bar{x}, \sigma)$	normal distribution over x with mean $\bar{x}$ and standard deviation σ
b	proposal density for birth distribution

Specific probabilities

$p_D(x)$	detection probability of an object with state x
$q_D(x)$	missed detection probability of an object with state x
$p_S(x)$	survival (or persistence) probability of an object with state x
r	existence probability

Measurement likelihoods

$g(z x)$	single-object likelihood for a measurement z given an object with state x
----------	---

$g(\mathbf{Z} \mathbf{X})$	multi-object likelihood for a multi-object measurement $\mathbf{Z}$ given an multi-object state $\mathbf{X}$
----------------------------	--

Bibliography

- [ACG⁺06] Alexander, Lee; Cheng, Pi-Ming; Gorjestani, Alec; et al.: *The Minnesota Mobile Intersection Surveillance System*. In: *Proceedings of the IEEE Intelligent Transportation Systems Conference*, pages 139–144, 2006.
- [AMB07] Arras, K.O.; Mozos, O.M.; and Burgard, W.: *Using Boosted Features for the Detection of People in 2D Range Data*. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3402–3407, 2007.
- [AS72] Alspach, D.L. and Sorenson, H.W.: *Nonlinear Bayesian estimation using Gaussian sum approximations*. In: *IEEE Transactions on Automatic Control*, volume 17, no. 4, pages 439–448, 1972.
- [BB00] Bar-Shalom, Y. and Blair, W.D.: *Multitarget-Multisensor Tracking: Applications and Advances*. From the series *Artech House Radar Library*. Artech House, 2000.
- [BB88] Blom, H.A.P. and Bar-Shalom, Y.: *The interacting multiple model algorithm for systems with Markovian switching coefficients*. In: *IEEE Transactions on Automatic Control*, volume 33, no. 8, pages 780–783, 1988.
- [BBP⁺12] Broughton, Jeremy; Brandstatter, Christian; Pace, Jean-Francois; et al.: *Basic Fact Sheet "Main Figures", Deliverable D3.9 of the EC FP7 project DaCoTA*. Tech. rep. European Road Safety Observatory, 2012. URL: http://ec.europa.eu/transport/road_safety/pdf/statistics/dacota/bfs2012_dacota-trl-main_figures.pdf.
- [Ber88] Bertsekas, Dimitri P.: *The auction algorithm: A distributed relaxation method for the assignment problem*. In: *Annals of Operations Research*, volume 14, no. 1, pages 105–123, 1988. URL: <http://dx.doi.org/10.1007/BF02186476>.
- [BEV08] Bouwmans, Thierry; El Baf, Fida; and Vachon, Bertrand: *Background modeling using mixture of gaussians for foreground detection-a survey*. In: *Recent Patents on Computer Science*, volume 1, no. 3, pages 219–237, 2008.

- [BF88] Bar-Shalom, Y. and Fortmann, T.: *Tracking and Data Association*. Academic Press, Inc., 1988.
- [BL71] Bourgeois, François and Lassalle, Jean-Claude: *An extension of the Munkres algorithm for the assignment problem to rectangular matrices*. In: *Communications of the ACM*, volume 14, no. 12, pages 802–804, 1971. URL: <http://doi.acm.org/10.1145/362919.362945>.
- [BL93] Bar-Shalom, Y. and Li, X.-R.: *Estimation and Tracking: Principles, Techniques, and Software*. Artech House Inc., 1993.
- [BL98] Bar-Shalom, Y. and Li, X.-R.: *Estimation and Tracking: Principles, Techniques, and Software*. YBS Publishing, 1998.
- [Bou13] Bouguet, J. Y.: *Camera calibration toolbox for Matlab*. 2013. URL: http://www.vision.caltech.edu/bouguetj/calib_doc/.
- [BP99] Blackman, S. and Popoli, R.: *Design and Analysis of Modern Tracking Systems*. Artech House Publishers, 1999.
- [BPT⁺12] Broughton, Jeremy; Pace, Jean-Francois; Tormo, Maria Teresa; et al.: *Basic Fact Sheet "Junctions", Deliverable D3.9 of the EC FP7 project DaCoTA*. Tech. rep. European Road Safety Observatory, 2012. URL: http://ec.europa.eu/transport/road_safety/pdf/statistics/dacota/bfs2012_dacota-trl-junctions.pdf.
- [BT75] Bar-Shalom, Yaakov and Tse, Edison: *Tracking in a cluttered environment with probabilistic data association*. In: *Automatica*, volume 11, no. 5, pages 451–460, 1975.
- [CCL04] Chang, Fu; Chen, Chun-Jen; and Lu, Chi-Jen: *A linear-time component-labeling algorithm using contour tracing technique*. In: *computer vision and image understanding*, volume 93, no. 2, pages 206–220, 2004.
- [CM12] Clark, D. and Mahler, R.: *Generalized PHD filters via a general chain rule*. In: *Proceedings of the 15th International Conference on Information Fusion*, pages 157–164, 2012.
- [CPV06] Clark, D.E.; Panta, K.; and Vo, B.N.: *The GM-PHD Filter Multiple Target Tracker*. In: *Proceedings of the 9th International Conference on Information Fusion*, 2006.
- [CV07] Clark, D. and Vo, Ba-Ngu: *Convergence Analysis of the Gaussian Mixture PHD Filter*. In: *IEEE Transactions on Signal Processing*, volume 55, no. 4, pages 1204–1212, 2007.
- [Dag12] Dagli, Gökhan: *Sensorplanung zur Überwachung von Kreuzungen mit Laserscannern*. Studienarbeit, Ulm University, 2012.

- [DDVH11] Delande, E.; Duflos, E.; Vanheeghe, P.; and Heurguier, D.: *Multi-sensor PHD: Construction and implementation by space partitioning*. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3632–3635, 2011.
- [Dem68] Dempster, A. P.: *A Generalization of Bayesian Inference*. In: *Journal of the Royal Statistical Society. Series B (Methodological)*, volume 30, no. 2, pages, 1968. URL: <http://www.jstor.org/stable/2984504>.
- [DMM⁺11] Dyckmanns, H.; Matthaei, R.; Maurer, M.; et al.: *Object Tracking in Urban Intersections Based on Active Use of a Priori Knowledge: Active Interacting Multi Model Filter*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 625–630, 2011.
- [DMX10] Dryanovski, I.; Morris, W.; and Xiao, Jizhong: *Multi-volume occupancy grids: An efficient probabilistic 3D mapping model for micro aerial vehicles*. In: *International Conference on Intelligent Robots and Systems (IROS)*, pages 1553–1559, 2010.
- [DV88] Daley, D.J. and Vere-Jones, D.: *An Introduction to the Theory of Point Processes*. Springer, New York, 1988.
- [Ehr10] Ehrlich, J. et al.: *Final Report: Results on test and validation*. Tech. rep. D2.5.2. SafeSpot SP2-INFRASENS-SP Infrastructure Platform, 2010.
- [EKSX96] Ester, Martin; Kriegel, Hans-Peter; Sander, Jörg; and Xu, Xiaowei: *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*. In: *International Conference on Knowledge Discovery and Data Mining*, pages 22–231, 1996.
- [Elf89] Elfes, A.: *Using occupancy grids for mobile robot perception and navigation*. In: *Computer*, volume 22, no. 6, pages 46–57, 1989.
- [FBS83] Fortmann, T.; Bar-Shalom, Y.; and Scheffe, M.: *Sonar tracking of multiple targets using joint probabilistic data association*. In: *IEEE Journal of Oceanic Engineering*, volume 8, no. 3, pages 173–184, 1983.
- [GCB⁺10] Gidel, S.; Checchin, P.; Blanc, C.; Chateau, T.; and Trassoudaine, L.: *Pedestrian Detection and Tracking in an Urban Environment Using a Multilayer Laser Scanner*. In: *Transactions on Intelligent Transportation Systems*, volume 11, no. 3, pages 579–588, 2010.
- [GGZ⁺13] Goldhammer, M.; Gerhard, M.; Zernetsch, S.; Doll, K.; and Brunsmann, U.: *Early prediction of a pedestrian's trajectory at intersections*. In: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, pages 237–242, 2013.

- [GLW⁺13] Gabb, Michael; Löhlein, Otto; Wagner, Raimar; et al.: *High-Performance On-Road Vehicle Detection in Monocular Images*. In: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, pages 336–341, 2013.
- [GMC⁺08] Gorjestani, Alec; Menon, Arvind; Cheng, Pi-Ming; Shankwitz, Craig; and Donath, Max: *The Design of an Optimal Surveillance System for a Cooperative Collision Avoidance System - Stop Sign Assist: CICAS-SSA Report*. Tech. rep. 2. Intelligent Vehicles Laboratory ITS Institute University of Minnesota, 2008.
- [GMN97] Goodman, I.R.; Mahler, R.P.; and Nguyen, H.T.: *Mathematics of Data Fusion*. From the series *Theory and decision library: Series B : Mathematical and Statistical Methods*. Springer, 1997.
- [GO12a] Granström, K. and Orguner, U.: *Estimation and maintenance of measurement rates for multiple extended target tracking*. In: *Proceedings of the 15th International Conference on Information Fusion*, pages 2170–2176, 2012.
- [GO12b] Granström, K. and Orguner, U.: *A PHD Filter for Tracking Multiple Extended Targets Using Random Matrices*. In: *IEEE Transactions on Signal Processing*, volume 60, no. 11, pages 5657–5671, 2012.
- [GRMS14] Granstrom, Karl; Reuter, Stephan; Meissner, Daniel; and Scheel, Alexander: *A multiple model PHD approach to tracking of cars under an assumed rectangular shape*. In: *17th International Conference on Information Fusion*, pages 1–8, 2014.
- [GSM⁺12] Goldhammer, Michael; Strigel, Elias; Meissner, Daniel; et al.: *Cooperative multi sensor network for traffic safety applications at intersections*. In: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, pages 1178–1183, 2012.
- [GW12a] Georgescu, R. and Willett, P.: *The Multiple Model CPHD Tracker*. In: *IEEE Transactions on Signal Processing*, volume 60, no. 4, pages 1741–1751, 2012.
- [GW12b] Georgescu, Ramona and Willett, Peter: *Classification aided cardinalized probability hypothesis density filter*. In: volume 8392, pages, 2012. URL: <http://dx.doi.org/10.1117/12.917729>.
- [HHW10] Himmelsbach, M.; v Hundelshausen, F.; and Wuensche, H.: *Fast segmentation of 3D point clouds for ground vehicles*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 560–565, 2010.
- [HL10] Houssineau, J. and Laneuville, D.: *PHD filter with diffuse spatial prior on the birth process with applications to GM-PHD filter*. In: *Proceedings of the 13th International Conference on Information Fusion*, pages 1–8, 2010.

- [HL64] Ho, Y.-C. and Lee, R.: *A Bayesian approach to problems in stochastic estimation and control*. In: *IEEE Transactions on Automatic Control*, volume 9, no. 4, pages 333–339, 1964.
- [HM04a] Hall, David L. and McMullen, Sonya A. H.: *Mathematical Techniques in Multisensor Data Fusion*. Artech House Inc., Norwood, 2004.
- [HM04b] Hoffman, J.R. and Mahler, R.P.S.: *Multitarget miss distance via optimal assignment*. In: *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, volume 34, no. 3, pages 327–336, 2004.
- [HW12] Himmelsbach, M. and Wuensche, H. -J.: *Tracking and classification of arbitrary objects with bottom-up/top-down detection*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 577–582, 2012.
- [Ins14] Inspecta, S.L.: *OpenCVBlobsLib*. <http://opencvblobslib.github.io/opencvblobslib/>. 2014.
- [JDSW12] Jonsson, R.; Degerman, J.; Svensson, D.; and Wintenby, J.: *Multitarget tracking with background discrimination using PHD filters*. In: *Proceedings of the 15th International Conference on Information Fusion*, pages 854–860, 2012.
- [JROR10] Jensen, Pablo; Rouquier, Jean-Baptiste; Ovtracht, Nicolas; and Robardet, Celine: *Characterizing the speed and paths of shared bicycle use in Lyon*. In: *Transportation Research Part D: Transport and Environment*, volume 15, no. 8, pages 522–524, 2010. URL: <http://www.sciencedirect.com/science/article/pii/S136192091000101X>.
- [JU96] Julier, S. and Uhlmann, J.: *A General Method for Approximating Nonlinear Transformations of Probability Distributions*. In: 1996. URL: <http://www.smpp.northwestern.edu/savedLiterature/JulierUhlmannUnscented.pdf>.
- [JUD95] Julier, S.J.; Uhlmann, J.K.; and Durrant-Whyte, H.F.: *A new approach for filtering nonlinear systems*. In: *Proceedings of the American Control Conference*. Volume 3, pages 1628–1632, 1995.
- [Kal60] Kalman, Rudolph Emil: *A New Approach to Linear Filtering and Prediction Problems*. In: *Transactions of the ASME - Journal of Basic Engineering*, volume 82, no. Series D, pages 35–45, 1960.
- [Kal84] Kalata, Paul R.: *The Tracking Index: A Generalized Parameter for $\alpha - \beta$ and $\alpha - \beta - \gamma$ Target Trackers*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 20, no. 2, pages 174–182, 1984.
- [Kam07] Kämpchen, Nico: *Feature-level fusion of laser scanner and video data for advanced driver assistance systems*. PhD thesis, Ulm University, 2007. URL: <http://vts.uni-ulm.de/doc.asp?id=5958>.

- [KB01] Kaewtrakulpong, P. and Bowden, R.: *An Improved Adaptive Background Mixture Model for Realtime Tracking with Shadow Detection*. In: *European Workshop on Advanced Video Based Surveillance Systems*, 2001.
- [KGB⁺13] Koehler, S.; Goldhammer, M.; Bauer, S.; et al.: *Stationary Detection of the Pedestrian's Intention at Intersections*. In: *Intelligent Transportation Systems Magazine*, volume 5, no. 4, pages 87–99, 2013.
- [KKD12] Kellner, Dominik; Klappstein, Jens; and Dietmayer, Klaus: *Grid-based DBSCAN for clustering extended objects in radar data*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 365–370, 2012.
- [KoF14] Ko-FAS: *research initiative Ko-FAS*. <http://www.ko-fas.de>. 2014.
- [KPN96] Knoblauch, Richard L.; Pietrucha, Martin T.; and Nitzburg, Marsha: *Field studies of pedestrian walking speed and start-up time*. In: *Transportation Research Record: Journal of the Transportation Research Board*, volume 1538, no. 1, pages 27–38, 1996.
- [Kuh55] Kuhn, H. W.: *The Hungarian Method for the Assignment Problem*. In: *Naval Research Logistics Quarterly*, volume 2, pages 83–97, 1955.
- [KWM94] Koller, Dieter; Weber, Joseph; and Malik, Jitendra: *Robust multiple car tracking with occlusion reasoning*. In: Eklundh, Jan-Olof (editor): *Computer Vision*, volume 800. From the series *Lecture Notes in Computer Science*, pages 189–196, Springer Berlin Heidelberg, 1994. URL: http://dx.doi.org/10.1007/3-540-57956-7_22.
- [LB96] Li, Xiao-Rong and Bar-Shalom, Y.: *Multiple-model estimation with variable structure*. In: *IEEE Transactions on Automatic Control*, volume 41, no. 4, pages 478–493, 1996.
- [LCB12] Lamard, L.; Chapuis, R.; and Boyer, J.-P.: *Dealing with occlusions with multi targets tracking algorithms for the real road context*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 371–376, 2012.
- [LHLC11] Lian, F.; Han, C.; Liu, W.; and Chen, H.: *Joint spatial registration and multi-target tracking using an extended probability hypothesis density filter*. In: *Radar, Sonar Navigation*, volume 5, no. 4, pages 441–448, 2011.
- [Mah03] Mahler, Ronald: *Multitarget Bayes filtering via first-order multitarget moments*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 39, no. 4, pages 1152–1178, 2003.
- [Mah04] Mahler, Ronald: *"Statistics 101" for multisensor, multitarget data fusion*. In: *IEEE Aerospace and Electronic Systems Magazine*, volume 19, no. 1, pages 53–64, 2004.

- [Mah07a] Mahler, Ronald: *Statistical Multisource-Multitarget Information Fusion*. Artech House Inc., Norwood, 2007.
- [Mah07b] Mahler, Ronald: *PHD filters of higher order in target number*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 43, no. 4, pages 1523–1543, 2007.
- [Mah09a] Mahler, Ronald: *The multisensor PHD filter: I. General solution via multitarget calculus*. In: volume 7336, pages, 2009. URL: <http://dx.doi.org/10.1117/12.818024>.
- [Mah09b] Mählich, Mirko: *Filtersynthese zur simultanen Minimierung von Existenz-, Assoziations- und Zustandsunsicherheiten in der Fahrzeugumfelderfassung mit heterogenen Sensordaten*. PhD thesis, Ulm University, 2009.
- [Mah10] Mahler, Ronald: *Approximate multisensor CPHD and PHD filters*. In: *Proceedings of the 13th International Conference on Information Fusion*, pages 1–8, 2010.
- [Mah12] Mahler, Ronald: *On Multitarget Jump-Markov Filters*. In: *Proceedings of the 15th International Conference on Information Fusion*, pages 149–156, 2012.
- [MD10] Meissner, Daniel and Dietmayer, Klaus: *Simulation and calibration of infrastructure based laser scanner networks at intersections*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 670–675, 2010.
- [ME04] Musicki, D. and Evans, R.: *Joint Integrated Probabilistic Data Association: JIPDA*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 40, no. 3, pages 1093–1099, 2004.
- [MES94] Musicki, D.; Evans, R.; and Stankovic, S.: *Integrated probabilistic data association*. In: *IEEE Transactions on Automatic Control*, volume 39, no. 6, pages 1237–1241, 1994.
- [MMD10] Munz, Michael; Mählich, Mirko; and Dietmayer, Klaus: *Generic Centralized Multi Sensor Data Fusion Based on Probabilistic Sensor and Environment Models for Driver Assistance Systems*. In: *IEEE Intelligent Transportation Systems Magazine*, volume 2, no. 1, pages 6–17, 2010.
- [MRD12] Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Real-time detection and tracking of pedestrians at intersections using a network of laserscanners*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 630–635, 2012.

- [MRD13] Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Combining the 2D and 3D World: A New Approach for Point Cloud Based Object Detection*. In: *Proceedings of the Intelligent Signal Processing Conference*, 2013.
- [MRSD14] Meissner, D.; Reuter, S.; Strigel, E.; and Dietmayer, K.: *Intersection-Based Road User Tracking Using a Classifying Multiple-Model PHD Filter*. In: *IEEE Intelligent Transportation Systems Magazine*, volume 6, no. 2, pages 21–33, 2014.
- [MRWD13] Meissner, Daniel; Reuter, Stephan; Wilking, Benjamin; and Dietmayer, Klaus: *Road User Tracking Using a Dempster-Shafer Based Classifying Multiple-Model PHD Filter*. In: *Proceedings of the 16th International Conference on Information Fusion*, pages 1236–1242, 2013.
- [MSN⁺10] Mori, T.; Sato, T.; Noguchi, H.; et al.: *Moving objects detection and classification based on trajectories of LRF scan data on a grid map*. In: *International Conference on Intelligent Robots and Systems (IROS)*, pages 2606–2611, 2010.
- [Mun11] Munz, Michael: *Generisches Sensorfusionsframework zur gleichzeitigen Zustands- und Existenzschätzung für die Fahrzeugumfeldererkennung*. PhD thesis, Ulm University, 2011.
- [Mun57] Munkres, James: *Algorithms for the Assignment and Transportation Problems*. In: *Journal of the Society for Industrial and Applied Mathematics*, volume 5, no. 1, pages 32–38, 1957. URL: <http://www.jstor.org/stable/2098689>.
- [MVV11] Mahler, Ronald; Vo, Ba-Tuong; and Vo, Ba-Ngu: *CPHD Filtering With Unknown Clutter Rate and Detection Profile*. In: *IEEE Transactions on Signal Processing*, volume 59, no. 8, pages 3497–3513, 2011.
- [NC11] Nagappa, Sharad and Clark, Daniel E.: *On the ordering of the sensors in the iterated-corrector probability hypothesis density (PHD) filter*. 2011. URL: <http://dx.doi.org/10.1117/12.884618>.
- [NCM11] Nagappa, S.; Clark, D.E.; and Mahler, R.: *Incorporating track uncertainty into the OSPA metric*. In: *Proceedings of the 14th International Conference on Information Fusion*, pages 1–8, 2011.
- [OA11] Ortega, Agustin and Andrade-Cetto, Juan: *Segmentation of dynamic objects from laser data*. In: *Proc. Eur. Conf. Mobile Robot*, pages 115–121, 2011.
- [PCV09] Panta, Kusha; Clark, Daniel E.; and Vo, Ba-Ngu: *Data Association and Track Management for the Gaussian Mixture Probability Hypothesis Density Filter*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 45, no. 3, pages 1003–1016, 2009.

- [PKS08] Punithakumar, K.; Kirubarajan, T.; and Sinha, A.: *Multiple-model probability hypothesis density filter for tracking maneuvering targets*. In: *Aerospace and Electronic Systems, IEEE Transactions on*, volume 44, no. 1, pages 87–98, 2008.
- [PLN09] Premebida, C.; Ludwig, O.; and Nunes, U.: *Exploiting LIDAR-based features on pedestrian detection in urban scenarios*. In: *Intelligent Transportation Systems, 2009. ITSC '09. 12th International IEEE Conference on*, pages 1–6, 2009.
- [PTS⁺12] Pace, Jean-Francois; Tormo, Maria Teresa; Sanmartin, Jaime; et al.: *Basic Fact Sheet "Urban Areas", Deliverable D3.9 of the EC FP7 project DaCoTA*. Tech. rep. European Road Safety Observatory, 2012. URL: http://ec.europa.eu/transport/road_safety/pdf/statistics/dacota/bfs2012_dacota-intras-urbanareas.pdf.
- [PVTM09] Pasha, S.A.; Vo, Ba-Ngu; Tuan, Hoang Duong; and Ma, Wing-Kin: *A Gaussian Mixture PHD Filter for Jump Markov System Models*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 45, no. 3, pages 919–936, 2009.
- [RCVV12] Ristic, B.; Clark, D.; Vo, Ba-Ngu; and Vo, Ba-Tuong: *Adaptive Target Birth Intensity for PHD and CPHD Filters*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 48, no. 2, pages 1656–1668, 2012.
- [RD09] Reuter, Stephan and Dietmayer, Klaus: *Fuzzy Estimation and Segmentation for Laser Range Scans*. In: *Proceedings of the 12th International Conference on Information Fusion*, pages 1974–1981, 2009.
- [RD11] Reuter, Stephan and Dietmayer, Klaus: *Pedestrian Tracking Using Random Finite Sets*. In: *Proceedings of the 14th International Conference on Information Fusion*, pages 1–8, 2011.
- [Rei79] Reid, D.: *An algorithm for tracking multiple targets*. In: *IEEE Transactions on Automatic Control*, volume 24, no. 6, pages 843–854, 1979.
- [Reu14] Reuter, Stephan: *Multi-Object Tracking Using Random Finite Sets*. PhD thesis, Ulm University, 2014.
- [RFW⁺11] Roessler, Bernd; Fuerstenberg, Kay; Westhoff, Daniel; Ahlers, Florian; and Kulmala, Risto: *INTER SAFE 2 Cooperative Intersection Safety Final Report*. Tech. rep. D1.2. 2011.
- [RGV⁺13] Rezatofighi, Seyed Hamid; Gould, Stephen; Vo, Ba-Ngu; et al.: *A multiple model probability hypothesis density tracker for time-lapse cell microscopy sequences*. In: *Information Processing in Medical Imaging*, 2013.

- [RVVD14a] Reuter, Stephan; Vo, Ba-Tuong; Vo, Ba-Ngu; and Dietmayer, Klaus: *Multi-Object Tracking Using Labeled Multi-Bernoulli Random Finite Sets*. In: *Proceedings of the International Conference on Information Fusion*, 2014.
- [RVVD14b] Reuter, Stephan; Vo, Ba-Tuong; Vo, Ba-Ngu; and Dietmayer, Klaus: *The Labeled Multi-Bernoulli Filter*. In: *IEEE Transactions on Signal Processing*. (under review), 2014.
- [RWW⁺13] Reuter, Stephan; Wilking, Benjamin; Wiest, Jürgen; Munz, Michael; and Dietmayer, Klaus: *Real-Time Multi-Object Tracking using Random Finite Sets*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 49, no. 4. to appear, pages 2666–2678, 2013.
- [SA71] Sorenson, H.W. and Alspach, D.L.: *Recursive Bayesian estimation using Gaussian sums*. In: *Automatica*, volume 7, no. 4, pages 465–479, 1971. URL: <http://www.sciencedirect.com/science/article/pii/0005109871900975>.
- [SAO⁺11] Schubert, R.; Adam, C.; Obst, M.; et al.: *Empirical evaluation of vehicular models for ego motion estimation*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 534–539, 2011.
- [SATS10] Spinello, Luciano; Arras, Kai Oliver; Triebel, Rudolph; and Siegwart, Roland: *A Layered Approach to People Detection in 3D Range Data*. In: *Conference on Artificial Intelligence*, 2010.
- [SG99] Stauffer, C. and Grimson, W.E.L.: *Adaptive background mixture models for real-time tracking*. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Volume 2, pages 637–663, 1999.
- [SGM⁺14] Scheel, Alexander; Granström, Karl; Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Tracking and Data Segmentation Using a GGIW Filter with Mixture Clustering*. In: *Proceedings of the International Conference on Information Fusion*, 2014.
- [Sha76] Shafer, Glenn: *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, 1976.
- [SK94] Smets, Philippe and Kennes, Robert: *The transferable belief model*. In: *Artificial Intelligence*, volume 66, no. 2, pages 191–234, 1994. URL: <http://www.sciencedirect.com/science/article/pii/0004370294900264>.
- [SMD13] Strigel, Elias; Meissner, Daniel; and Dietmayer, Klaus: *Vehicle Detection and Tracking at Intersections by Fusing Multiple Camera Views*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 882–887, 2013.

- [Sme90] Smets, Philippe: *The combination of evidence in the transferable belief model*. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 12, no. 5, pages 447–458, 1990.
- [SMS⁺14] Strigel, Elias; Meissner, Daniel; Seeliger, Florian; Wilking, Benjamin; and Dietmayer, Klaus: *The Ko-PER Intersection Laserscanner and Video Dataset (Extended Abstract)*. In: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, 2014.
- [SR11] Stimming, Christian and Roessler, Bernd: *Cooperative Infrastructure Safety Systems using Infrastructure Laserscanner*. In: *Proceedings of the International Workshop on Intelligent Transportation*, pages 97–102, 2011.
- [SSM62] Smith, Gerald L.; Schmidt, Stanley F.; and Mc Gee, Leonard A.: *Application of Statistical Filter Theory to the Optimal Estimation of Position and Velocity on Board a Circumlunar Vehicle*. Tech. rep. NASA Ames Research Center, Tech. Rep. TR R-135, 1962.
- [Sta14] Statistisches Bundesamt: *Fahrzeugbestand*. online. 2014. URL: <https://www.destatis.de/DE/ZahlenFakten/Wirtschaftsbereiche/TransportVerkehr/UnternehmenInfrastrukturFahrzeugbestand/Tabellen/Fahrzeugbestand.html>.
- [SVV08] Schuhmacher, D.; Vo, B.-T.; and Vo, B.-N.: *A Consistent Metric for Performance Evaluation of Multi-Object Filters*. In: *IEEE Transactions on Signal Processing*, volume 56, no. 8, pages 3447–3457, 2008.
- [SW09] Schubert, R. and Wanielik, G.: *Unifying Bayesian Networks and IMM Filtering for Improved Multiple Model Estimation*. In: *Proceedings of the 12th International Conference on Information Fusion*, pages 810–817, 2009.
- [SWBH12] Schueler, K.; Weiherer, T.; Bouzouraa, E.; and Hofmann, U.: *360 Degree multi sensor fusion for static and dynamic obstacles*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 692–697, 2012.
- [SWP⁺14] Seeliger, Florian; Weidl, Galia; Petrich, Dominik; et al.: *Advisory warnings based on cooperative perception*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 246–252, 2014.
- [TBF05] Thrun, Sebastian; Burgard, Wolfram; and Fox, Dieter: *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.
- [TPB06] Triebel, R.; Pfaff, P.; and Burgard, W.: *Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing*. In: *International Conference on Intelligent Robots and Systems*, pages 2276–2282, 2006.

- [UEW07] Ulmke, M.; Erdinc, O.; and Willett, P.: *Gaussian mixture cardinalized PHD filter for ground moving target tracking*. In: *Proceedings of the 10th International Conference on Information Fusion*, pages 1–8, 2007.
- [VJ01] Viola, P. and Jones, M.: *Rapid object detection using a boosted cascade of simple features*. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 511–518, 2001.
- [VM06] Vo, Ba-Ngu and Ma, Wing-Kin: *The Gaussian Mixture Probability Hypothesis Density Filter*. In: *IEEE Transactions on Signal Processing*, volume 54, no. 11, pages 4091–4104, 2006.
- [VSD05] Vo, B.-N.; Singh, S.; and Doucet, A.: *Sequential Monte Carlo Methods for Multitarget Filtering with Random Finite Sets*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 41, Issue 4, pages 1224–1245, 2005.
- [VV13a] Vo, Ba-Tuong and Vo, Ba-Ngu: *Tracking, identification, and classification with random finite sets*. In: *Proceedings of the SPIE 8745, Signal Processing, Sensor Fusion, and Target Recognition XXII*. International Society for Optics and Photonics, 2013.
- [VV13b] Vo, Ba-Tuong and Vo, Ba-Ngu: *Labeled Random Finite Sets and Multi-Object Conjugate Priors*. In: *IEEE Transactions on Signal Processing*, volume 61, no. 13, pages 3460–3475, 2013.
- [VVC08] Vo, Ba-Tuong; Vo, Ba-Ngu; and Cantoni, Antonio: *Bayesian Filtering With Random Finite Set Observations*. In: *IEEE Transactions on Signal Processing*, volume 56, no. 4, pages 1313–1326, 2008.
- [VVC09] Vo, Ba-Tuong; Vo, Ba-Ngu; and Cantoni, A.: *The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and Its Implementations*. In: *IEEE Transactions on Signal Processing*, volume 57, no. 2, pages 409–423, 2009.
- [Wen08] Wender, Stefan: *Multisensorsystem zur erweiterten Fahrzeugumfelderfassung*. PhD thesis, Ulm University, 2008.
- [WHB⁺10] Wurm, K. M.; Hornung, A.; Bennewitz, M.; Stachniss, C.; and Burgard, W.: *OctoMap: A Probabilistic, Flexible, and Compact 3D Map Representation for Robotic Systems*. In: *Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation*. Software available at <http://octomap.sf.net/>, Anchorage, AK, USA, 2010. URL: <http://octomap.sf.net/>.
- [WK11] Wertheimer, Reiner and Klanner, Felix: *Cooperative perception to promote driver assistance and preventive safety*. In: *8th International Workshop on Intelligent Transportation*, 2011.

- [WKH⁺11] Weimer, D.; Kohler, S.; Hellert, C.; et al.: *GPU architecture for stationary multisensor pedestrian detection at smart intersections*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 89–94, 2011.
- [WMRD13] Wilking, Benjamin; Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Information Maps: A Practical Approach to Position Dependent Parameterization*. In: *arXiv:1312.3808v1 [cs.CE]*, 2013.
- [Zad79] Zadeh, Lotfi A: *On the validity of Dempster’s rule of combination of evidence*. Electronics Research Laboratory, University of California, 1979.
- [ZCZ⁺09] Zhao, Huijing; Cui, Jinshi; Zha, Hongbin; et al.: *Sensing an intersection using a network of laser scanners and video cameras*. In: *Intelligent Transportation Systems Magazine, IEEE*, volume 1, pages 31–37, 2009.
- [Zha00] Zhang, Zhengyou: *A flexible new technique for camera calibration*. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, volume 22, no. 11, pages 1330–1334, 2000.
- [ZK07] Zivkovic, Z. and Krose, B.: *Part based people detection using 2D range data and images*. In: *International Conference on Intelligent Robots and Systems*, pages 214–219, 2007.

Publications

Parts of this thesis were pre-released in author's journal articles and conference papers.

Meissner, Daniel and Dietmayer, Klaus: *Simulation and calibration of infrastructure based laser scanner networks at intersections*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 670–675, 2010.

Meissner, Daniel and Dietmayer, Klaus: *High-Precision 3D Traffic Recognition at Intersections*. In: *Proceedings of the International Workshop on Intelligent Transportation*, pages 35–40, 2011.

Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Multi-Model-PHD-Filter zur Verfolgung von Verkehrsteilnehmern an Kreuzungen mit Laserscannern*. In: *8. Workshop Fahrerassistenzsysteme FAS*, pages 57–66, 2012. URL: <http://www.unidas.de/de/Veranstaltungen/fas2012.php>.

Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Real-time detection and tracking of pedestrians at intersections using a network of laserscanners*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 630–635, 2012.

Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Combining the 2D and 3D World: A New Approach for Point Cloud Based Object Detection*. In: *Proceedings of the Intelligent Signal Processing Conference*, 2013.

Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Road User Tracking at Intersections Using a Multiple-Model PHD Filter*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 377–382, 2013.

Meissner, D.; Reuter, S.; Strigel, E.; and Dietmayer, K.: *Intersection-Based Road User Tracking Using a Classifying Multiple-Model PHD Filter*. In: *IEEE Intelligent Transportation Systems Magazine*, volume 6, no. 2, pages 21–33, 2014.

Meissner, Daniel; Reuter, Stephan; Wilking, Benjamin; and Dietmayer, Klaus: *Road User Tracking Using a Dempster-Shafer Based Classifying Multiple-Model PHD Filter*. In: *Proceedings of the 16th International Conference on Information Fusion*, pages 1236–1242, 2013.

Granstrom, Karl; Reuter, Stephan; Meissner, Daniel; and Scheel, Alexander: *A multiple model PHD approach to tracking of cars under an assumed rectangular shape*. In: *17th International Conference on Information Fusion*, pages 1–8, 2014.

Goldhammer, Michael; Strigel, Elias; Meissner, Daniel; et al.: *Cooperative multi sensor network for traffic safety applications at intersections*. In: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, pages 1178–1183, 2012.

Reuter, Stephan; Meissner, Daniel; and Dietmayer, Klaus: *Multi-object tracking at intersections using the cardinalized probability hypothesis density filter*. In: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, pages 1172–1177, 2012.

Reuter, Stephan; Meissner, Daniel; Wilking, Benjamin; and Dietmayer, Klaus: *Cardinality Balanced Multi-Target Multi-Bernoulli Filtering Using Adaptive Birth Distributions*. In: *Proceedings of the 16th International Conference on Information Fusion*, pages 1608–1615, 2013.

Reuter, Stephan; Vo, Ba-Tuong; Wilking, Benjamin; Meissner, Daniel; and Dietmayer, Klaus: *Divergence Detectors for the δ -Generalized Labeled Multi-Bernoulli Filter*. In: *8th Workshop Sensor Data Fusion: Trends, Solutions, and Applications*, Bonn, Germany, 2013.

Scheel, Alexander; Granström, Karl; Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Tracking and Data Segmentation Using a GGIW Filter with Mixture Clustering*. In: *Proceedings of the International Conference on Information Fusion*, 2014.

Strigel, Elias; Meissner, Daniel; and Dietmayer, Klaus: *Vehicle Detection and Tracking at Intersections by Fusing Multiple Camera Views*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 882–887, 2013.

Strigel, Elias; Meissner, Daniel; Seeliger, Florian; Wilking, Benjamin; and Dietmayer, Klaus: *The Ko-PER Intersection Laserscanner and Video Dataset (Extended Abstract)*. In: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, 2014.

Wilking, Benjamin; Meissner, Daniel; Reuter, Stephan; and Dietmayer, Klaus: *Information Maps: A Practical Approach to Position Dependent Parameterization*. In: *arXiv:1312.3808v1 [cs.CE]*, 2013.

Supervised Theses

Dagli, Gökhan: *Sensorplanung zur Überwachung von Kreuzungen mit Laserscannern*. Studienarbeit, Ulm University, 2012.

Knill, Christina: *Modellierung von Freiraum und Verdeckung in 3D Laserscannerdaten*. Bachelor Thesis, Ulm University, 2012.

Michalka, Tobias: *Adaptive und robuste Hintergrundschätzung in Laserscannerdaten an Kreuzungen*. Bachelor Thesis, Ulm University, 2011.

Sailer, Christian: *Entwurf und Aufbau des Praktikumsversuchs "schwebender Körper im Magnetfeld"*. Diplomarbeit, Ulm University, 2010.

Schwab, Heiko: *Identifikation der Parameter eines generischen Formmodells für Pkws*. Studienarbeit, Ulm University, 2013.