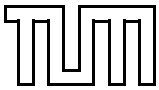


Technische Universität München
Fakultät für Elektrotechnik und Informationstechnik
Lehrstuhl für Datenverarbeitung

Scalable Map-based Environment Representations for Highly Automated Vehicles

Tobias Weiherer, M.Sc.

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr. techn. Josef A. Nossek
Prüfer der Dissertation:	1. Univ.-Prof. Dr.-Ing. Klaus Diepold
	2. Univ.-Prof. Dr.-Ing. Matthias Althoff

Die Dissertation wurde am 22.12.2014 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 16.06.2015 angenommen.

Vorwort

Die vorliegende Arbeit entstand im Rahmen meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Datenverarbeitung der Technischen Universität München, wobei ich im Rahmen eines INI.TUM-Projektes dauerhaft im „Projekthaus Fahrerassistenzsysteme“ der AUDI AG beschäftigt war.

Mein Dank gilt zunächst meinem Doktorvater Prof. Dr.-Ing. Klaus Diepold für seine Unterstützung in den vergangenen Jahren sowie seine wertvollen Anregungen zu dieser Arbeit und über deren Rahmen hinaus.

Herrn Prof. Dr.-Ing. Matthias Althoff danke ich für sein Interesse an meiner Arbeit und die Übernahme der Aufgabe des Zweitprüfers.

Großen Anteil am Erfolg dieser Arbeit haben meine Ansprechpartner bei der AUDI AG, mein Betreuer Dr.-Ing. Ulrich Hofmann und Dr.-Ing. Mohamed Essayed Bouzouraa. Die zahlreichen gemeinsamen, teils intensiven Diskussionen brachten unzählige interessante Ansätze hervor, von denen nur ein kleiner Teil die Grundlage der vorliegenden Arbeit bildet - vielen Dank dafür!

Ein Dank gilt ferner allen Kollegen im „Projekthaus Fahrerassistenzsysteme“, für dessen außergewöhnlich angenehme Arbeitsatmosphäre hier stellvertretend Dr.-Ing. Karl-Heinz Siedersberger gedankt sei. Vor allem den Kollegen im Wahrnehmungsteam Kai Schüler, Christopher Demiral, Caterina Vitadello, Dr.-Ing. Ekaterina Timoshenko, Martin Kellner, Marvin Raaijmakers, Max Schmidt und Jens Storz möchte ich für die produktive und unterhaltsame Zusammenarbeit danken. Dominik Seibert danke ich für seine wichtigen Beiträge im Rahmen einer Masterarbeit. Ein besonderer Dank gilt Dr.-Ing. Stephan Neumaier, der für mathematische Fragestellungen stets ein offenes Ohr und entsprechend fundiertes Fachwissen vorzuweisen hatte.

Meinen zahlreichen Englisch-Korrekturlesern danke ich für ihre Hilfe und ihre gründliche Arbeit.

Zuletzt gilt mein Dank meiner Familie und meinen Freunden für die Unterstützung und Ablenkung in den vergangenen Jahren. Meiner Freundin Cornelia danke ich für ihre Unterstützung bei der Korrektur, ihr Verständnis und ihre unbeirrbare Zuversicht, ohne der diese Arbeit nicht möglich gewesen wäre - Danke!

Ingolstadt, im Dezember 2014

Tobias Weiherer

Abstract

The perception of a vehicle's environment is one of the key issues on the road towards highly automated driving. To further increase the robustness and degree of automation of current Advanced Driver Assistance Systems, the environment perception needs to represent all relevant environment information in sufficient accuracy while taking into consideration the limited computing capabilities of automotive control units. Map-based environment representations have been used to describe individual unstructured environment features at a high level of detail but also generate a high computational effort. This thesis therefore provides several contributions to optimize the resource requirements and to extend the contents of map-based environment representations.

The core of this thesis is the development of a generic interval-based environment representation, which particularly simplifies the interpretation of the described environment features. In the first part, the interval-based approach is used to realize an efficient representation of occupancy information, which considers the inherent uncertainties of the applied laser sensor and incorporates moving objects. In order to compare the quality of the developed map against state-of-the-art approaches a novel map-based evaluation scheme is developed. The experimental results indicate similar map qualities of grid- and interval-based approaches, while the computational requirements can be diminished by applying the new representation.

To improve the robustness of automated vehicles' lateral control, this thesis further addresses, for the first time, the abstract environment feature of common moving object behavior. The proposed approach uses the interval-based map for a location-dependent description of object motions, which can then be used to extract convoy tracks. Based on an analysis of measurement characteristics in dense traffic scenarios, the necessary sensor models for radar and laser sensors are implemented. The developments are concluded by an evaluation concept which allows for measuring the quality of object motion maps.

The last part of the thesis presents a concept to combine map-based environment representations with varying contents and levels of detail. It is shown how cooperative sensor models can improve map estimation by simultaneously using the information of different map content layers. Furthermore, the proposed concept allows to combine and exchange information between interval- and grid-based maps, which increases the scalability of the entire map-based representation. Finally, a comparison to purely grid- and interval-based representations illustrates the advantages of this approach.

Kurzfassung

Die Wahrnehmung des Fahrzeugumfelds ist eine der Schlüsselaufgaben auf dem Weg zu hochautomatisiertem Fahren. Um die Robustheit und den Automatisierungsgrad heutiger Fahrerassistenzsysteme weiter zu erhöhen, muss die Umfeldwahrnehmung sämtliche relevanten Umfeldmerkmale in ausreichender Güte repräsentieren und dabei die begrenzten Rechenkapazitäten automobiler Steuergeräte berücksichtigen. Zur Beschreibung einzelner, unstrukturierter Umfeldmerkmale haben sich kartenbasierte Umfeldrepräsentationen etabliert, stellen allerdings hohe Anforderungen an Rechenleistung und Speicherbedarf. Die vorliegende Arbeit liefert mehrere Beiträge hinsichtlich der Reduktion des Ressourcenbedarfs und der Erweiterung des Inhalts kartenbasierter Umfeldrepräsentationen.

Den Kern der Arbeit bildet die Entwicklung einer generischen, intervallbasierten Umfeldrepräsentation, die sich besonders durch eine vereinfachte Interpretation der beschriebenen Umfeldmerkmale auszeichnet. Im ersten Teil wird dieser Ansatz zur Entwicklung einer Belegungskarte genutzt, die sowohl die Unsicherheiten der verwendeten Sensorik als auch dynamische Objekte korrekt behandelt. Um einen Vergleich zu gitterbasierten Repräsentationen zu ermöglichen, werden anschließend neuartige Bewertungsmethoden entwickelt. Die damit bestimmten Ergebnisse zeigen, dass der Einsatz von Intervallbelegungskarten eine Verringerung des Rechen- und Speicheraufwands bei ähnlichen Kartenqualitäten erlaubt.

Zur robusteren Planung der Querführung automatisierter Fahrzeuge wird in dieser Arbeit anschließend erstmalig das abstrakte Umfeldmerkmal der Kollektivbewegung anderer Verkehrsteilnehmer betrachtet. Der vorgestellte Ansatz basiert auf einer ortsabhängigen Beschreibung von Objektbewegungen in einer Intervallkarte, aus der anschließend Kolonnenspuren extrahiert werden können. Die benötigten Sensormodelle für Radar- und Lasersensorik werden auf Basis einer Analyse von Messdaten in charakteristischen Verkehrsszenarien entwickelt. Abschließend werden Bewertungsmethoden vorgestellt, mit Hilfe derer sich die Qualität der erstellten Karten quantifizieren lässt.

Im letzten Teil der Arbeit wird ein Konzept zur Kombination kartenbasierter Umfeldrepräsentationen mit verschiedenen Inhalten und Detaillierungsgraden erarbeitet. Es wird gezeigt, wie durch den Einsatz kooperativer Sensormodelle die gleichzeitige Schätzung unterschiedlicher Kartenebenen verbessert werden kann. Das entwickelte Konzept erlaubt weiterhin, Intervall- und Gitterkarten zu kombinieren und zwischen den Karten Informationen zu übertragen. Durch diesen Ansatz lässt sich die Skalierbarkeit der Gesamtrepräsentation erhöhen, was durch einen abschließenden Vergleich mit rein gitter- und intervallbasierten Karten verdeutlicht wird.

Contents

List of Figures

List of Tables

List of Abbreviations

ADAS	Advanced Driver Assistance System
ESC	Electronic Stability Control
ACC	Adaptive Cruise Control
LDW	Lane Departure Warning
LKS	Lane Keeping Support
USK	Environment Sensor Coordinate System (German: *Umfeldsensorkoordinatensystem*)
KF	Kalman Filter
EKF	Extended Kalman Filter
RADAR	Radio Detection and Ranging
FMCW	Frequency Modulated Continous Wave
LRR3	Third Generation Long Range Radar
SRR	Short Range Radar
LIDAR	Light Detection and Ranging
PMD	Photonic Mixer Device
GPS	Global Positioning System
DGPS	Differential Global Positioning System (GPS)
IMU	Inertial Measurement Unit
ADTF	Automotive Data and Time-Triggered Framwork
pdf	probability density function
cdf	cumulative distribution function
GM	Grid Map
2DIM	2D Interval Map
GNSS	Global Navigation Satellite System

SLAM	Simultaneous Localization and Mapping
DATMO	Detection and Tracking of Moving Objects
SLAMMOT	Simultaneous Localization, Mapping and Moving Object Tracking
DST	Dempster-Shafer Theory
RP	Reference Point
IMF	Information Matrix Fusion
TP	True Positives
TN	True Negatives
FP	False Positives
FN	False Negatives

List of Symbols

${}^{A}\boldsymbol{p}$	Coordinate p in coordinate system A
${}^{A}\tilde{\boldsymbol{p}}$	Homogeneous coordinate p in coordinate system A
${}^{B}\mathcal{P}_{A}$	Homogeneous transformation matrix (pose) between coordinate systems A and B
ψ	Yaw angle
θ	Pitch angle
γ	Roll angle
$\mathcal{R}$	Rotation matrix
$\mathcal{T}$	Translation matrix
$\boldsymbol{o}_k$	Moving object estimation at time step k
v_x, v_y	Moving object velocity in x- and y-direction
a_x, a_y	Moving object acceleration in x- and y-direction
δ	Orientation of a moving object bounding box
L	Length of a moving object bounding box
W	Width of a moving object bounding box
k	Time step of a discrete-time system
$\boldsymbol{x}_k$	State vector at time step k
$\boldsymbol{z}_k$	Measurement at time step k
$\boldsymbol{z}_{1:k}$	Measurements between time steps 1 and k
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
$\boldsymbol{\Phi}$	State transition matrix / Jacobian of state transition function
$\boldsymbol{C}$	Measurement matrix / Jacobian of measurement function
$\boldsymbol{Q}$	Process noise covariance matrix
$\boldsymbol{R}$	Measurement noise covariance matrix
$\overset{*}{\boldsymbol{x}}_k$	Predicted state vector at time step k
$\overset{*}{\boldsymbol{P}}_k$	Predicted error covariance at time step k
$\hat{\boldsymbol{x}}_k$	Estimated state vector at time step k
$\hat{\boldsymbol{P}}_k$	Estimated error covariance at time step k
$\boldsymbol{K}_k$	Kalman gain at time step k
o	Occupancy state
$\boldsymbol{M}_k$	Map-based environment representation at time step k
$\boldsymbol{I}^{(n)}$	N-th interval of a 2DIM
$I^{(n)}$	Coordinate system of n-th interval
$\psi^{(n)}$	Yaw angle of n-th interval
$\boldsymbol{OC}$	2DIM occupancy cell

$r^{(p)}$	P-th laser range measurement distance
$\vartheta^{(p)}$	P-th laser range measurement vertical angle
$\varphi^{(p)}$	P-th laser range measurement horizontal angle
α	Horizontal laser beam divergence
d	Occupancy cell dynamic state
R_i	Reference map cell probability
E_i	Cell probability of map under evaluation
$\boldsymbol{MC}$	2DIM motion cell
$c^{(n,m)}$	Center of m-th motion cell in n-th 2DIM interval
$l^{(n,m)}$	Left border of m-th motion cell in n-th 2DIM interval
$r^{(n,m)}$	Right border of m-th motion cell in n-th 2DIM interval
$\phi^{(n,m)}$	Direction of m-th motion cell in n-th 2DIM interval
$v^{(n,m)}$	Average velocity of m-th motion cell in n-th 2DIM interval
$\boldsymbol{u}_k$	Equivalent measurement at time step k
$\boldsymbol{U}_k$	Equivalent measurement error covariance at time step k
L_i	I-th content layer of a map-based environment representation

1. Introduction

1.1. Automated Vehicles

The vision of autonomous vehicles has been fascinating car manufacturers and scientists from different research areas for decades. The following introduction will motivate the research on vehicle automation and briefly describe the historical development of increasing vehicle automation. The remaining challenges in realizing highly automated vehicles and the contributions of this thesis in this context will be introduced.

1.1.1. Motivation

A historical imagination about the vision of autonomous driving can be identified in figure 1.1, which depicts a famous newspaper advertisement of America's Independent Electric Light and Power Companies from the 1950's. The authors of this sketch predicted that

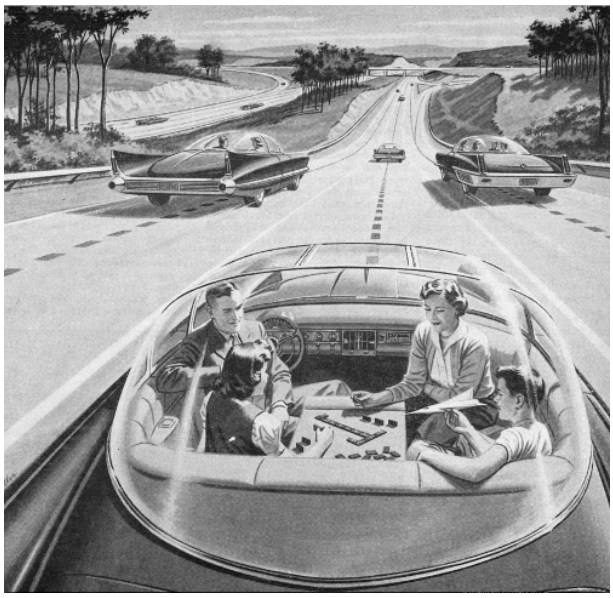


Figure 1.1.: 1950's newspaper commercial from America's Independent Electric Light and Power Companies, from [2].

"electronic devices" would make completely autonomous vehicles possible, which allow the passengers to spend time on side tasks like board games during driving. Besides the emotional fascination and the pioneering spirit that is addressed by the self driving vehicle, two major advantages of automated driving are advertised: "Travel will be more enjoyable. It will be made safe." [2].

Even from today's point of view, there is still enormous potential in approaching the goals of improving the level of comfort and safety for drivers. According to the German General Statistics Office, 3600 people were killed in road traffic accidents in Germany 2012 [156]. Although this figure has never been lower since 1950, it still means that on average about ten people are killed in road traffic every day. A "misbehavior of the vehicle's driver" was at least one of the recorded accident reasons in 86 % of all road traffic injuries [156]. Approaching higher degrees of vehicle automation is one of the measures to reach the ambitious goal of further decreasing the number of fatalities.

However, the global megatrends of urbanization and increased mobility requirements are increasingly changing the characteristics of road traffic. In major German cities, the delay per hour driven in peak periods was up to 38 minutes in 2012, according to TomTom's annual congestion index [166]. Over one year, a driver loses up to 89 hours on a daily 30 minute commute due to traffic jams, the situation in other European and especially Asian countries is even worse. In these situations, the task of driving is not perceived as enjoyable. If a vehicle was able to assist or take over the driving task in tedious situations, the driver would be able to deal with side tasks, which can help to improve his relaxation and productivity.

1.1.2. Driver Assistance Systems

Since the historical vision from the 1950's and today, major progress towards comfortable and safe driving has been made. First of all, the *passive safety* of vehicles has been increased enormously, for example by introducing airbag systems and optimizing the cabin. However, it is assumed that the full potential of passive safety systems is almost realized today [86]. The introduction of driver assistance systems has delivered further contributions to improve safety and comfort. Concerning safety issues, several *active safety* systems, which already try to avoid the emergence of critical situations, have been realized. Popular examples are Electronic Stability Control (ESC) and advanced pre-crash systems that warn the driver or even try to decelerate the vehicle after having detected a critical situation [185]. The combination of active and passive safety approaches is commonly denoted as *integral safety* [86]. Popular examples of systems that increase the driver's comfort are Adaptive Cruise Control (ACC) [187], Lane Departure Warning (LDW), Lane Keeping Support (LKS), park assists and lane change assists [185]. Especially ACC, a cruise control system that automatically adjusts the vehicle's speed according to distances and velocities of detected vehicles ahead, has reached significant market penetration and is also available in compact cars nowadays.

Maurer [103] classifies driver assistance according to the complexity of the sensor data interpretation into conventional systems, e.g. anti-lock braking systems, and systems with machine perception, e.g. ACC. Similarly, the term Advanced Driver Assistance System (ADAS) has become established for systems which use complex signal processing algorithms in their environment perception [36]. In [111], Naab and Reichart propose a different classification depending on the level of intervention. They distinguish driver assistance systems into *pure warning* systems, *automatically intervening* systems and *automatically operating* systems. Whereas automatically intervening systems only interfere in dedicated, uncontrollable situations, automatically operating systems take over certain subtasks of driving. From this point of view, ESC and pre-crash systems can be considered automatically intervening systems, while ACC or LKS are regarded as automatically operating systems.

Another important categorization with regard to the legal assessment of automated vehicles was elaborated by Gasser et al. in a project group of the BaSt[1] [56]. Similar definitions can be found in the current draft J3016 of the SAE[2] International [138]. The most important taxonomies of both reports and corresponding examples are illustrated in figure 1.2. According to that, warning systems are assigned to the lowest level of automation, as they

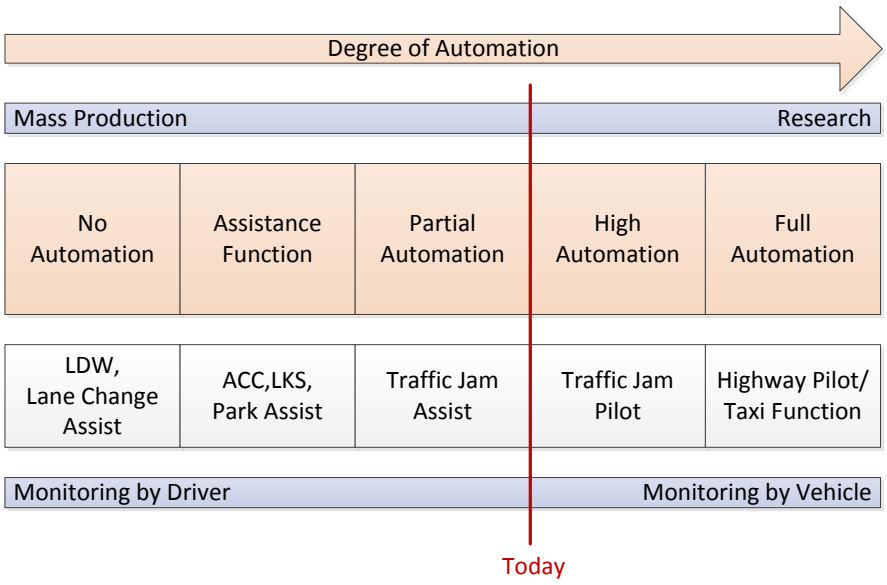


Figure 1.2.: Degrees of automation in driver assistance systems, adapted from [56, 138].

do not contribute to vehicle operation directly. By contrast, pure *assistance functions* are characterized by taking over either longitudinal (e.g. ACC) or lateral control (e.g. LKS) within certain system boundaries, while the remaining task is up to the driver. In such

[1]Bundesanstalt für Straßenwesen - German Federal Highway Research Institute
[2]Society of Automotive Engineers

systems, the driver is responsible for monitoring the mode of operation and has to be able to take over control whenever the system boundaries are reached. *Partially automated* systems take over both longitudinal and lateral control for a certain period of time. Examples for such systems are traffic jam assists, which are currently entering the market. Still, the monitoring task and hence the responsibility to interfere in critical situations remains with the driver. This restriction is removed in the category of *High Automation*, where the vehicle additionally takes over the monitoring task and automatically recognizes when the system limits are reached. Nevertheless, the driver has to finally take over control in these situations after a limited period of time, as the system is not guaranteed to autonomously reach a so called *state of minimal risk*. By contrast, *Full Automation* is indicated by the ability to return to a safe state, whenever the driver is not willing or able to take over control. Hence, the vehicle is completely responsible for the monitoring task. In this thesis, the term *highly automated vehicle* will be used to describe a system that is able to reliably take over longitudinal and lateral control for a longer period of time. A detailed study of the required monitoring concepts is not within the scope of this thesis.

1.1.3. Highly Automated Vehicles

On their road towards the vision of full automation, car manufacturers followed the strategy of incrementally increasing the available degree of automation [33]. As also visualized in figure 1.2, today's market-leading driver assistance systems can be located at the difficult transition between partially and highly automated systems. In the robotics and intelligent vehicle's research community, several concepts of higher degrees of automation have been examined and demonstrated in the last decades. Apart from early visions and concepts of autonomous cars, substantial progress was first made in the 1980's. From a historical point of view, especially the pioneering work of Ernst Dickmanns' group should be mentioned [34]. In the mid 80s, they presented their first autonomous vehicle "VaMoRs"[149], which was able to drive at speeds up to 96 km/h on empty roads by using real-time computer vision. Another major step forward was made by the driverless car competitions of the DARPA[3] from 2004 to 2007, the *DARPA Grand Challenges* [147]. In 2007, the winning vehicle "Boss" from Carnegie Mellon University [170], drove over 4 hours automated in an urban environment. Since then, Google has gained great popularity [50] by its concepts of a driverless car. For a more detailed discussion of especially European research projects on automated vehicles within the last decades, the interested reader is referred to [133].

The main criticism of the presented research vehicles concerns to the applied hardware devices. First of all, most of the research vehicles' sensors are still too expensive and error-prone for serial deployment, especially in approaches that are based on accurate global positioning. Furthermore, the applied sensors, e.g. tall 3D laser scanners, can hardly be integrated in mass production cars without abandoning today's standards of automotive design. Finally, the energy consumption of the required sensors and also

[3] Defense Advanced Research Projects Agency

computing units has to be considered, especially in the context of the CO2 reduction targets the European automotive industry will be faced with. Car manufacturers all over the world are currently trying to close this gap between research and mass production. Recently, Germany's leading premium vehicle manufacturers have also shown concept cars which use pre-production sensor sets and computing units. Audi has demonstrated show cars (see figure 1.3) which allow *Piloted driving* in traffic jams and parking maneuvers [59]. The expression piloted driving emphasizes that the system is able to assume the driving task whenever the driver perceives driving as tedious, comparable to the auto-pilot in a plane. In 2013, Daimler demonstrated a reengineered Mercedes-Benz S 500 "Intelligent Drive". According to [29], this vehicle was able to autonomously drive about 100km on both interurban and urban routes, but on the downside required highly accurate and up-to-date digital map information, e.g. including the positions of traffic lights, lanes and crossings.

Figure 1.3.: Audi's 2013 concept car for piloted driving in traffic jams [59].

One of the main reasons, why highly automated vehicles haven't entered the market yet is the legal situation. The Vienna Convention on Road Traffic from 1968 requires that every moving vehicle must have a driver that is able to control the vehicle "at all times" and "in all circumstances" [168]. As this convention has been ratified by several states, including all states of the European Union, similar principles can be found in current standards. The current version of the European effective norm for steering systems (ECE R79) only allows hands-off steering in low speed situations of up to 10 km/h [169]. In the United States, there is no uniform jurisdiction concerning automated vehicles. Up to now, several states have started to enact laws which regulate the testing of autonomous vehicles [59]. Besides these general homologation constraints, also possible obligations to provide proof in the case of an accident have not decisively been clarified [97].

On the other hand, further technical issues have to be solved to bring higher degrees of automation closer to production vehicles. The reliability and the scope of state-of-the-art methods has to be increased, while at the same time the resource requirements must

remain within reasonable bounds. In order to approach an analysis of the limitations of existing technologies, the high-level system architecture of a highly automated vehicle is introduced in figure 1.4. In the literature, detailed architectures of autonomous vehicles have been proposed, especially under consideration of the different driving task levels according to Rasmussen [132]. The interested reader is referred to the works of Maurer [102], Siedersberger [149] and Stiller et al. [158]. For the purpose of this thesis, an abstract functional architecture adapted from [133] will be used. Although there exist different taxonomies for particular modules and tasks in the literature, the basic principles of the major components are widely accepted and can e.g. also be found in [170] and [77]. The terminology in this thesis will conform to the expressions used in figure 1.4.

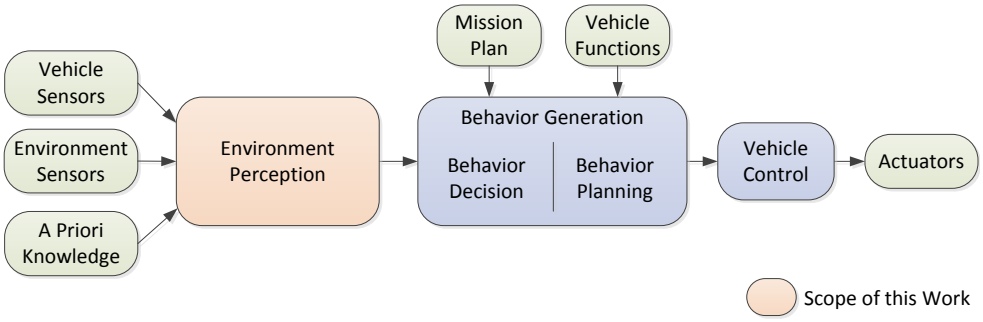


Figure 1.4.: High-level system architecture of a highly automated vehicle.

In this architecture, the flow of information starts with the perception of the environment and one's own vehicle state by different sensors. Taking into consideration a priori knowledge, the sensor measurements are processed and fused in order to build a model of the environment. This model will be denoted as *environment representation* in this thesis. The representation is then used to infer relations, maneuvers and meanings of a traffic situation, depending on the requirements of the remaining components. Based on the result of the perception, the knowledge of the vehicle's functions and a predefined goal, the desired behavior of the vehicle has to be generated. First, a decision-making module chooses a reasonable maneuver from the set of currently possible maneuvers. Then, this maneuver serves as an input for the behavior planning, which calculates a trajectory including the desired positions and velocities. Finally, the available actuators are controlled to influence the lateral and longitudinal motion of the vehicle in order to achieve the desired trajectory.

The contributions of this thesis can help to overcome the major technical limitations concerning the resource requirements and reliability of existing highly automated vehicle approaches. As also illustrated in figure 1.4, the scope of this thesis is limited to the development of new approaches in the field of environment perception, which will be introduced in the next section.

1.2. Environment Perception

The perception of the vehicle's environment is one of the key challenges in the development of future ADASs and highly automated vehicles. In order to outline the contributions of this paper in this research area, the following section will explain the tasks of environment perception and define the taxonomy and architecture which will be used throughout the thesis. Based on the previously identified general conditions, the requirements of environment representations will be formulated.

1.2.1. Environment Perception Architecture

Environment perception has been examined extensively in different research areas, including robotics, artificial intelligence, computer vision, as well as control and automation engineering. Concerning the development of highly automated vehicles, the ultimate goal of this component is to provide all information about the environment that is necessary to generate a robust automated behavior of the vehicle. According to Hofmann [66], the task of perceiving the environment includes:

- The extraction of features from sensor data
- The development of models concerning the appearance of detected features in different contexts
- The development of models concerning the temporal behavior of detected features

In the research environment, this thesis originates from, the different environment perception components are subdivided into *primary* and *secondary environment perception*. This classification will also be used in this thesis.

The focus of the primary environment perception lies on the deduction of an internal model of the physical world from the raw measurements of all integrated sensors. The resulting model is based on a set of representations of different environment features with different levels of abstraction. Thus, there are for example low level descriptions which serve as intermediate representations for secondary environment perception and high level abstractions that already include models about e.g. temporal behavior. The application area and surrounding physical world strongly influence which aspects of the environment have to be recognized. All aspects the primary environment representation is concerned with will be denoted as *physical environment features* in this thesis. Taken as a whole, all features have to form an approximated but consistent internal image of the vehicle's physical environment on a low level of abstraction.

In any driver assistance system, environment perception has to deal with a complex and highly dynamic environment. In state-of-the-art highly automated vehicle architectures several different features of the environment are recognized, including:

- Free spaces and obstacles around the car (occupancy information), e.g. [16, 173, 178]

- All traffic participants (e.g. pedestrians, vehicles, bicyclists) including models of their appearance, type and temporal behavior, e.g. [78, 98], [SWBH12]
- Road infrastructure including road markings [66], road surfaces [93], traffic signs [106] and lights [57]

In the literature, also the vehicle's state and position is sometimes considered a part of the internal environment model [35]. Furthermore, the global position estimation and the retrieval of digital map information can be interpreted as a subtask of the primary environment perception.

To obtain this information, a wide range of different sensors can be applied. They vary significantly in their physical measurement principles. Today's most popular automotive sensors can be categorized into ultrasonic, radar, lidar, mono cameras, stereo cameras and time-of-flight cameras. Concerning details about the measurement principles of these sensors, the interested reader is referred to section B in [185], which provides an exhaustive overview. For the estimation of the vehicle's state and position, there exist vehicle dynamics sensors and Global Navigation Satellite System (GNSS) receivers. Besides that, digital maps with information about road topologies can also be interpreted as a *virtual sensor*. As of today, there does not exist a single sensor that is able to satisfy the requirements of driver assistance systems with higher degrees of automation. To overcome this issue, the common strategy is to combine several sensors with complementary measurement principles and mounting positions. This implies the need to fuse sensor data. For that purpose, there exist several architectures which differ in their topology, synchronization and the applied level of abstraction, a detailed study can e.g. be found in [35].

In order to satisfy the requirements of a driver assistance function, representations of higher abstraction levels have to be inferred during the secondary environment representation (also known as situation interpretation, analysis or assessment [133]). In this step, meanings can be assigned to represented objects by establishing relationships between different entities. The resulting situation representation provides information about *abstract environment features*, which cannot be measured directly in the physical world. The most popular examples in this category concern the further processing of occupancy information. Well-known publications in this context deal with the localization as a part of the popular SLAM problem [165, 182, 141], the extraction of road boundaries from occupancy states [181, 71, 30, 16], a compact description of free spaces for path planning [16, 144, 145] or the analysis of free spaces concerning evasion maneuvers [68, 135]. Detected road markings can be used to infer lanes in the vehicle's environment, which is essential information for highly automated vehicles, for example shown in [180, 55]. Finally, there also exist works which try to assign roles and maneuvers to dynamic objects, for example by linking them with topological road information [84, 69] or by learning interrelations [127]. For this purpose, abstract environment features can also be further related on higher levels of abstraction.

Overall, the perception system architecture illustrated in figure 1.5 will be assumed as the basis for this thesis. In this approach, the primary environment perception consists of

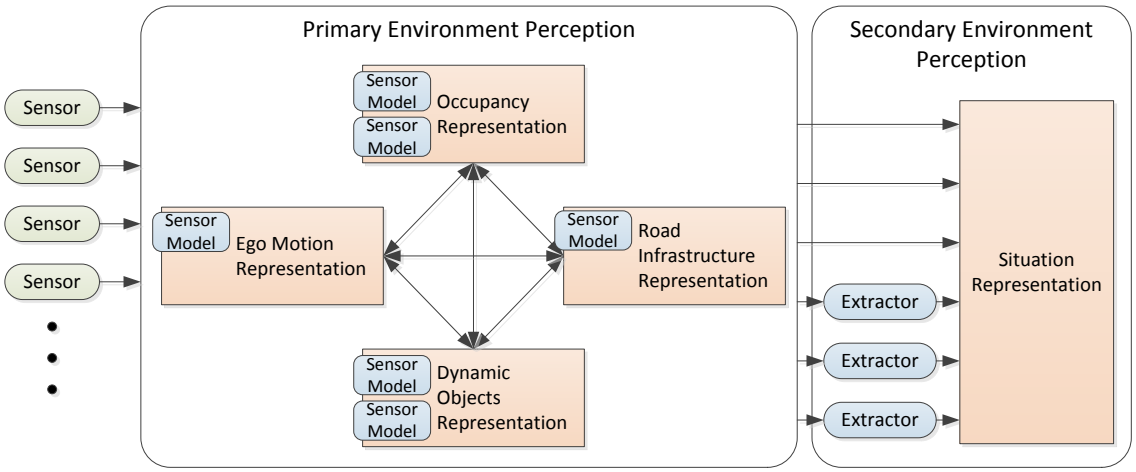


Figure 1.5.: Environment perception architecture assumed in this thesis, adapted from [16, 135, 133].

representations of four different environment features: occupancy states, dynamic objects, the own vehicle motion and the road infrastructure. Optionally, these modules can exchange information, for example the estimated ego motion. Each representation requires the implementation of dedicated *sensor models* which extract features from raw sensor measurements and update the data structures under consideration of the sensor characteristics. The combination of all represented environment features yields a consistent internal model of the environment, which can be used for several different driver assistance functions [135]. In order to deduce the relevant data from the representations, *extractors* are applied. During the secondary perception, the represented features and extraction results are combined to infer a high-level situation representation. The internal representation of environment features is one of the key issues in both primary and secondary environment representations.

1.2.2. Requirements on Environment Representations

The most popular approaches to represent environment features originate from the robotics and computer vision research community. Compared to these applications, environment representations for driver assistance systems are faced with advanced requirements. They result from the complex environment, the safety critical control task and the limited computing capabilities of automotive control units. They are listed in the following:

- **Accuracy**: An environment model has to be able to represent knowledge from high resolution environment measurements sufficiently accurate for driver assistance systems in varying situations. This especially addresses the ability to precisely describe unstructured environment features that do not conform with generic models.

- **Ability to represent uncertainties**: The representation has to be able to deal with inherent uncertainties. These uncertainties arise from the highly unpredictable behavior of a dynamic environment as well as the limited perception of sensors and the resulting measurement noise. Furthermore, all environment representations use abstractions and approximations in internal models and algorithms. Consequently, a feasible representation should be able to reliably provide the level of uncertainty and a measure of quality.
- **Enabling of accumulation**: The representation should be capable of reducing the uncertainties about the state of the environment by the consideration of several noisy measurements. This implies the ability to continuously and associably describe environment features over time.
- **Scalability**: In this context, scalability refers to the ability of an environment representation to adapt the level of abstraction and consequently the computational effort to the requirements of a driver assistance system. Moreover, scalability can also describe the capability to flexibly and uniformly deal with different sensors and input data.
- **Computational effort and memory consumption**: Considering the still limited computing capabilities and communication interfaces of today's automotive control units an environment representation has to be compact and easy to compute. Only this way it is possible to simultaneously represent the different features of the environment in an embedded device.
- **Interpretability**: One of the contributions of this thesis is the identification of the interpretation effort as a quality characteristic of environment representations. Within the scope of this paper, interpretability is defined as the effort that results from identifying and extracting relevant data from a representation. An ideal environment representation should provide sufficient accuracy but also offer high interpretability so that the other components of the system don't have to cope with irrelevant details. The interpretability of a representation heavily impacts the computational requirements of the overall environment perception and has hardly been considered in previous works.

Besides these requirements, a generic architecture of an environment representation is desirable. This especially addresses the ability to deal with several environment features and sensors as well as reusability in different driver assistance systems.

1.3. State of the Art

Having identified the structure and the requirements of the environment perception for highly automated vehicles, this section will categorize and briefly summarize state-of-

the-art environment representations. The identification of the limitations of existing approaches will lead to the formulation of the research problem in the following section.

1.3.1. Overview of Environment Representations

In [16], Bouzouraa provides a general classification of environment representations for driver assistance systems into *occupancy maps* and *model-based object descriptions*. Similar categorizations can be found in other publications. The first category is also denoted as *metric* representation [191, 145] or reduced to *grid-based* representations [68], whereas the second category is sometimes described as *parametric* [145] or *feature-based* [3, 119] representation. According to Bouzouraa, occupancy maps are characterized by:

- A dense and complete partition of a part of the physical world into space elements
- A neighborhood relationship between space elements
- An unambiguous assignment of attributes to space elements
- A pose between the map and the vehicle, which allows for the interpretation of the represented information

The best known of this type are occupancy grids, which were first mentioned by Moravec and Elfes in 1985 [109]. In this case, the space around a robot is partitioned into equally spaced grid cells, which provide the probability of the existence of obstacles in this space.

On the other hand, model-based object representations are composed of a possibly variable number of objects to which states and attributes can be attached [16]. This requires the prior definition of an object model. These models are usually compact and abstract descriptions of environment features, which are based on assumptions and simplifications of the real world. The most popular approach within this category is the application of bounding box models to describe moving objects in traffic scenes (e.g. [78], [SWBH12]). Both examples are illustratively compared in figure 1.6.

From another point of view, also occupancy maps use model assumptions if we consider their space elements as objects. The inherent assumption is that the surrounding world can be modeled by abstract and discretized space elements, in the case of occupancy grids by a number of equidistant grid cells. However, the model assumptions in occupancy maps are significantly less restrictive. Another key difference between both approaches is the handling of sensor measurements that do not match the predefined model assumptions. Model-based object representations typically decline measurements if they do not match the model, for example, if predefined shapes cannot be extracted from a laser scan. By contrast, standard occupancy mapping approaches try to express the measured environment as best as their model allows, for example when combining inconsistent measurements within a grid cell. This is also a consequence of the most obvious difference, the density and neighborhood relationship of the incorporated objects. Model-based object approaches can use an arbitrary number of objects to represent the environment, while

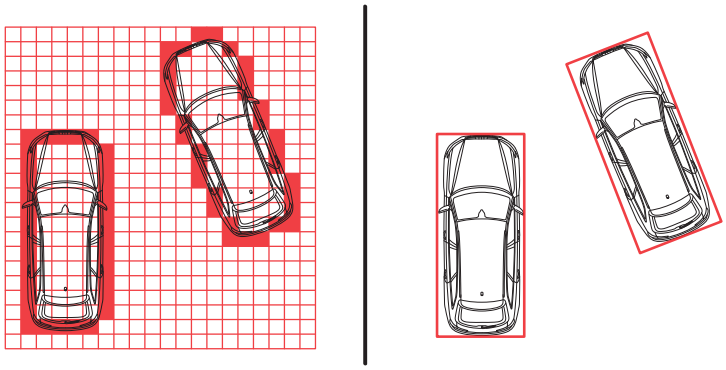


Figure 1.6.: Illustrative comparison of occupancy grid (left) and bounding box representation (right) of a traffic scene. Red paintings indicate the representation.

occupancy maps require a fixed dense partition of the world into objects with neighborhood relationship. In short, the main differences between map- and model-based object representations are characterized by:

- The density and neighborhood relationship of objects
- The handling of measurements that do not match the model
- The level of abstraction and model assumptions

The fundamental classification of environment representations in both categories will also be used in this thesis. The first category will be referred to as *map-based environment representations*, the term *occupancy map* will only be used for the representation of occupancy information, which is the knowledge about free space and obstacles. The term *occupancy grid* only refers to occupancy maps which use two-dimensional Cartesian grids to create space elements. Similar to [16], the second category of representations will be denoted as *model-based object representations*.

Concerning their positioning, map-based environment representations can be further classified into two categories. For grid-based representations, Scheunert et al. [140] describe these categories as *global static* grids and *moving windowed global* accumulation grids. In the first category, the pose of the map in a global coordinate system is fixed. In the second category, the map moves along with the vehicle. In this work, the categories will be denoted as *global* and *ego local* maps. Global maps originate from mobile robot applications, where the area in which the robot moves is limited and previously known. For the application in driver assistance systems, especially ego local maps with a guaranteed preview area are convenient, because a persistent mapping of the complete application area is unfeasible.

Due to these characteristics, map- and model-based object representations are typically applied for different tasks within the primary and secondary environment perception. Map-based approaches are particularly suitable for location dependent descriptions of

semi- or unstructured environment features. Consequently, their main application area is the representation of occupancy states including the highly unstructured free space around the vehicle. On the other hand, model-based object representations show their strengths whenever certain aspects of the environment conform well with a model. They are mostly used to describe dynamic objects, road markings, the topological road network or inferred abstract environment features.

Overall, there exist numerous map and model-based object representations for occupancy states in the environment of automated robots and vehicles. A detailed discussion of related publications regarding the previously identified requirements can be found in section 3.1. The presented techniques differ significantly in their level of abstraction and consequently also in their computational requirements. Concerning map-based representations, the most popular approaches rely on grid- or tree-based data structures for two- or three-dimensional space. The general advantage of using trees instead of grids results from the adaptable resolution and memory demand, which on the downside decreases computational efficiency and interpretability. To overcome the high memory consumption of equally spaced grid structures, approaches that extract or compress the represented information have also been proposed, e.g. [144, 145, 61, 62]. They will be analyzed in detail in section 3.1.

Concerning the description of moving objects, model-based object representations have become a de facto standard. The underlying spatial model is typically given by simplistic two-dimensional bounding box rectangles in case of laser and radar measurements processing (e.g. [184, 78], [SWBH12]) or by advanced three-dimensional geometric shapes for visual object tracking, e.g. [66, 112]. The estimation of the object dynamics is based on different models of the longitudinal and lateral motion of a vehicle, which can either be processed separately or together, e.g. by using the single-track-model [105]. During the secondary environment perception, maneuvers and roles of dynamic vehicles can also be inferred, which further improves the prediction of future behavior. There also exist map-based approaches to describe moving objects. So called *object-local* occupancy maps (e.g. [42, 140, 5]) aim to provide a more detailed description of the shape of a dynamic obstacle.

1.3.2. Limitations of Existing Approaches

The general knowledge about surrounding free and occupied areas at a certain point in time is a crucial piece of information for every highly automated vehicle. Model-based object approaches can be used to provide compact and easy-to-interpret representations of occupancy information, but have major shortcomings due to restrictive model assumptions, especially in unstructured environments. Map-based representations, however, use less limiting model assumptions, but create a higher computational effort and memory demand for processing and interpreting the data structure. One of the main disadvantages of grid-based approaches is given by their high and inflexible memory consumption.

Tree-based applications solve this problem at the expense of further increasing the interpretation effort and computation times. Proposed techniques which are based on a subsequent extraction or compression of occupancy maps do not address the problem of simplifying the accumulation process itself. To sum up, there does not exist a representation that is able to accumulate occupancy measurements and fulfills the requirements concerning scalability, accuracy, memory demand and computational effort. Especially the interpretation effort of the resulting data structures has not been considered sufficiently in the evaluation of existing approaches so far. As a consequence, state-of-the-art occupancy maps are not adapted to the specific requirements of dedicated driver assistance systems.

In the previously introduced state-of-the-art approaches, the representation of other traffic participants is limited to the estimation of spatio-temporal models of single objects. These models can be used to describe the current shape of the objects as well as to predict their future behavior. Still, all these approaches limit themselves to the description of the current state of a single object in the physical world. In doing so, they neglect the abstract environment information that is given by the *collective behavior* of traffic participants. By contrast, a human driver adapts his or her behavior to what he or she considers common standard behavior, for example in dense traffic scenarios around an accident site, as illustrated in figure 1.7. By representing the *collective motion of dynamic objects*, an additional input parameter for the behavior generation in driver assistance systems can be inferred. To the author's knowledge, a location-dependent representation of object dynamics has not been considered in the literature so far, a discussion of related work will be shown in section 4.1. If this information was additionally provided, taking into account the requirements concerning simplicity and interpretation, the overall reliability of highly automated vehicles could be further improved.

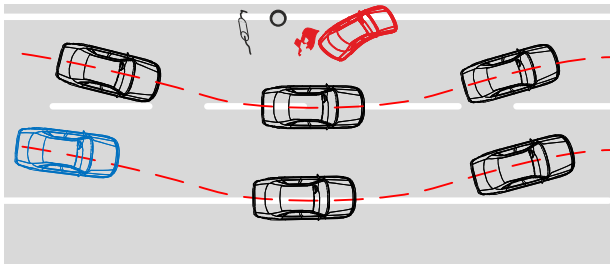


Figure 1.7.: Example for collective motion of traffic participants around an accident scene, own vehicle painted blue.

In the literature, map-based environment representations are usually considered as independent and stand-alone sources of information. Although there exist approaches which additionally incorporate information from model-based object representations (e.g. [3, 18]), map-based environment representations typically consist of a single map instance which is either aligned globally or ego locally. There does not exist an in-depth concept to combine map-based representations with different positions or even of different contents, for example maps that include lane markings [70], road curbs [83] or mapped camera im-

ages [89]. In doing so, the potential to mutually benefit from the simultaneous estimation of different environment features is neglected.

1.4. Research Problem Formulation

The identification and representation of all relevant information in the vehicle's environment is a crucial task in realizing reliable, highly automated vehicles. Especially unstructured and location-dependent environment information is best represented in map-based environment representations, because of their marginal model assumptions and their inherent location dependency.

In contrast to the previously described state of the art, map-based environment representations have to fulfill the requirements of automotive electronic control units with limited memory and computing capabilities. This results in the need of a *more compact, more efficient and easier to interpret map-based environment representation.* In spite of these requirements, a feasible representation of the vehicle's environment has to be able to *deal with the inherent uncertainties of sensor measurements.* This includes an adequate representation of existing uncertainties as well as the development of mechanisms to *accumulate different sensor measurements* while taking into account the laws of statistics.

A representation of *occupancy information* is an important source of information for a reliable and secure mode of operation of highly automated vehicles. The potential to reduce computational requirements by applying a simplified map-based representation for this purpose has to be systematically analyzed. In doing so, occupancy maps have to *consistently describe the static and dynamic* environment of a vehicle, which is an important requirement for an application in any driver assistance system. In order to evaluate the results of new and state-of-the-art occupancy maps, quantifiable methods to compare the contents and interpretability of different representations have to be developed.

To further improve the overall reliability of automated vehicles, the information about the *collective motion behavior of dynamic obstacles* has to be systematically investigated. This includes the definition of the aspects of collective motion that can be used in driver assistance systems and an analysis of sensor technologies concerning their abilities to detect this information. The objective is to obtain a map-based environment representation that is able to accumulate measurements of moving objects and fulfills the requirements concerning computational efficiency and accuracy. The quality of the represented information and the impact of influencing factors has to be quantified.

Current approaches lack in providing an adequate concept of how to *combine map-based environment representations* with different data structures, positions and contents. In order to allow the representations to benefit from each other, *concepts and mechanisms to transfer information* between maps have to be developed. In doing so, the *spatio-temporal*

consistency of all the represented information has to be ensured. Again, the resulting improvements concerning scalability, interpretability and accuracy have to be evaluated and compared to current approaches.

1.5. Contributions

In this thesis, the structure of the environment perception and the requirements of environment representations in highly automated vehicles have been determined. Based on a discussion of the state of the art, the limitations and scientific gaps in existing approaches have been identified. The general goal of this thesis is to optimize map-based environment representations and to represent additional environment features in map-based descriptions. The main contributions to the formulated research problem are illustrated in figure 1.8 and briefly summarized in the following:

- The introduction of an interval-based map framework as a generic and simplified map-based environment representation
- The representation of occupancy information in the interval-based map
- The representation of object motion in the interval-based map in order to obtain information about collective motion behavior
- The combination of grid- and interval-based environment representations
  - with different contents to improve the estimation of environment features
  - with different positions and spatial extents to enhance the scalability of the environment representation
- The development of evaluation methods to quantify the quality of the newly developed representations

All described concepts are implemented prototypically and integrated into an existing environment perception architecture. The improvement of the developed approaches over state-of-the-art technologies is evaluated by several different examinations. In order to quantify the computational requirements, the resulting computation times and memory requirements of the implemented modules are analyzed on standard PC hardware. The quality of existing and new environment representations is compared by the application of the evaluation methods. Finally, also the impacts of the new components on the performance of prototypical driver assistance systems are analyzed qualitatively. Taken together, these evaluations show a substantial improvement of the resulting environment perception compared to previous approaches.

1.6. Overview and Structure

The structure of this thesis is derived from the major contributions identified in the previous section and illustrated in figure 1.8. In this chapter, the general topics of vehicle

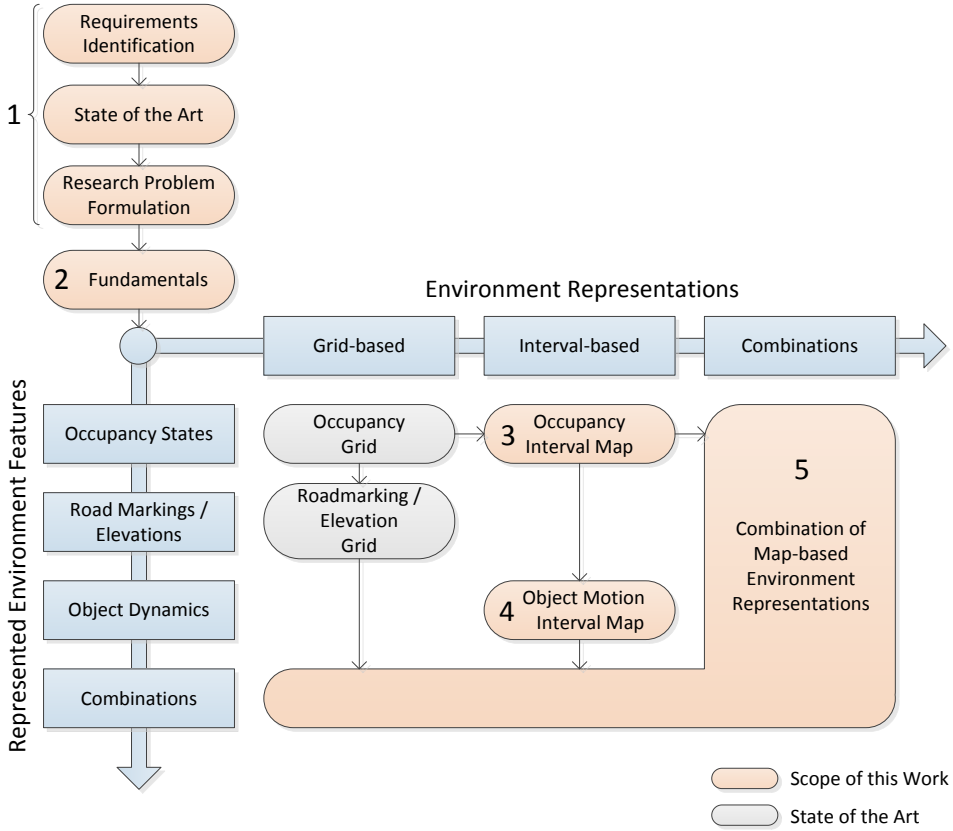


Figure 1.8.: Major contributions and structure of this thesis.

automation and environment perception have been introduced. Based on the identified requirements and a review of state-of-the-art approaches, the research problem has been formulated. The major contributions of this work have been pointed out and will be discussed in chapters 3 to 5. All three chapters follow a common structure that is illustrated in figure 1.9.

Before that, fundamental principles, which are necessary for the development of the new approaches in the subsequent chapters, are briefly summarized in chapter 2. The coordinate systems and coordinate transformations throughout this thesis follow a common and consistent formulation, which is introduced in this section. Besides that, the key principles of the collaborating environment perception modules are presented.

Chapter 3 starts with a detailed discussion of related works concerning the representation of occupancy information for robotic and intelligent vehicle applications. Then, the generic interval-based representation and its application to describe occupancy information are presented. After that, the required laser sensor model and corresponding update algorithms are derived. Another important topic is the analysis and implementation of ego motion compensation mechanisms for ego local interval maps. Finally, the occupancy map is also combined with model-based object tracking in order to correctly deal with dynamic objects. To evaluate the results of the interval-based occupancy representation and to compare both concepts, an extensive evaluation concept is proposed. For that purpose, mechanisms to infer ideal reference maps and to quantify differences between interval- and grid-based representations are presented.

The application of the interval-based representation to map object dynamics is the major topic of chapter 4. After a short discussion of related research concerning the detection of common dynamic object behavior, relevant parameters of traffic convoys for driver assistance systems are identified and defined. Then, a generic system architecture for the detection and description of traffic convoys is presented. The core of this approach forms a location dependent description of object dynamics, for which two different approaches are proposed. In order to develop suitable sensor models, a detailed analysis of laser and radar sensor measurements in typical occlusion scenarios is performed. The evaluation of the obtained results is based on the generation of ground truth data structures on different levels of abstraction and the development of metrics in order to quantify deviations.

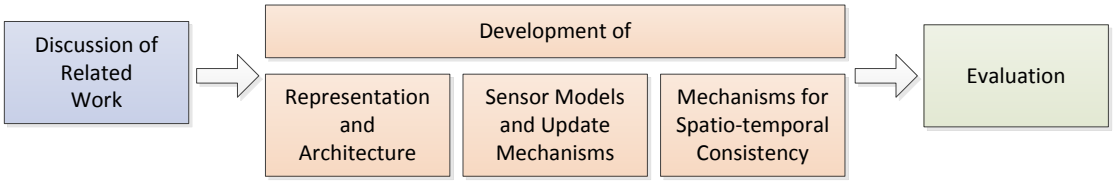


Figure 1.9.: Internal structure of chapters 3, 4 and 5.

Chapter 5 discusses different applications of grid maps and previous works on combinations of map-based environment representations. In order to enable a consistent representation of different environment features in map-based data structures, a layer concept for both grid- and interval-based representations is proposed. The resulting advantages for *cooperative sensor models* and the consequences on the previously developed ego motion compensation mechanisms are discussed. After that, both representations are enhanced to allow arbitrary poses between the map and the vehicle. By using this concept, a specified part of space can be covered by several different maps. *Virtual sensor models* are implemented to exchange information between map- and interval-based representations. On this basis, a scalable concept of combining map-based environment representations is formulated. Similar to the methods used in chapter 3, reference maps and inferred metrics are used to quantify the advantages of this methodology.

Finally, chapter 6.1 concludes the thesis and discusses possible perspectives for future work. Concerning the basic principles of recursive state estimation, which forms a key aspect in handling uncertain information in the environment perception, the interested reader is referred to appendix A. The measurement principles of the applied radar and laser sensors are described in appendix B, the software tooling environment is introduced in appendix C and additional experimental results can be found in appendix D.

2. Fundamentals

The following chapter will briefly introduce basic notations and principles, which will be required for the derivations in this thesis. All coordinate specifications and transformations will be described by using consistent notations and coordinate systems. The underlying homogeneous coordinates principle and coordinate system definitions will be outlined in the next two sections. After that, key principles of other participating environment perception subsystems will be explained.

2.1. Coordinate Notations

Homogeneous coordinates are a concept of projective geometry and typically known from computer vision applications [14]. By using this formulation, affine transformations between coordinate systems can be formulated by a single matrix multiplication. As a consequence, concatenated rotations and translations between several reference systems can be described consistently and easily comprehensible.

In general, a point $\boldsymbol{p}$ in a three-dimensional Cartesian coordinate system is transformed into a homogeneous coordinate $\tilde{\boldsymbol{p}}$ by adding a constant fourth element:

$$\boldsymbol{p} = [x, y, z]^T \rightarrow \tilde{\boldsymbol{p}} = [x, y, z, 1]^T \tag{2.1}$$

The translation and rotation between two coordinate systems A and B can then be described by a homogeneous 4x4 matrix, which will be denoted as *pose*. The pose incorporates a translation about a vector $[x, y, z]^T$ as well as a rotation around a yaw angle ψ, a pitch angle θ and a roll angle γ. The detailed composition of the required pose matrix is described in appendix A.1. Given a pose ${}^{B}\mathcal{P}_{A}$, we can transform any point $\boldsymbol{p}$ from the coordinate system A to coordinate system B by the matrix multiplication

$${}^{B}\tilde{\boldsymbol{p}} = {}^{B}\mathcal{P}_{A} \cdot {}^{A}\tilde{\boldsymbol{p}} \tag{2.2}$$

with ${}^{A}\tilde{\boldsymbol{p}}$ respectively ${}^{B}\tilde{\boldsymbol{p}}$ describing homogeneous coordinates in the coordinate system A respectively B. A back transformation can be accomplished by inverting the transformation matrix:

$${}^{A}\tilde{\boldsymbol{p}} = {}^{A}\mathcal{P}_{B} \cdot {}^{B}\tilde{\boldsymbol{p}} = \left({}^{B}\mathcal{P}_{A}\right)^{-1} {}^{B}\tilde{\boldsymbol{p}} \tag{2.3}$$

Furthermore, a concatenation of several transformations can be simplified by multiplying the transformation respectively back transformation matrices:

$$ {}^{C}\tilde{\boldsymbol{p}} = {}^{C}\mathcal{P}_{A} \quad \cdot {}^{A}\tilde{\boldsymbol{p}} = \left({}^{C}\mathcal{P}_{B} \cdot {}^{B}\mathcal{P}_{A}\right) \quad \cdot {}^{A}\tilde{\boldsymbol{p}} \tag{2.4} $$

$$ {}^{A}\tilde{\boldsymbol{p}} = \left({}^{C}\mathcal{P}_{A}\right)^{-1} \cdot {}^{C}\tilde{\boldsymbol{p}} = \left({}^{B}\mathcal{P}_{A}\right)^{-1} \cdot \left({}^{C}\mathcal{P}_{B}\right)^{-1} \cdot {}^{C}\tilde{\boldsymbol{p}} \tag{2.5} $$

Similarly, the homogeneous coordinate principle can also be applied to two-dimensional coordinates, which results in the following transformation:

$$ \boldsymbol{p} = [x, y]^{T} \rightarrow \tilde{\boldsymbol{p}} = [x, y, 1]^{T} \tag{2.6} $$

In that case, the pose ${}^{B}\mathcal{P}_{A}$ simplifies to a 3x3 matrix and is restricted to a two-dimensional translation and a single rotation. Again, the composition of the two-dimensional pose matrix can be found in appendix A.1.

2.2. Coordinate Systems

Throughout this thesis, several different coordinate systems will be used to process environment features and representations. They are summarized in the following list and illustrated in figure 2.1:

- *Sensor*: Each sensor uses a separate coordinate system to describe its measurements. Due to physical principles of laser, radar and video sensors, the resulting raw measurements are typically provided in polar coordinate systems. In that case, the origin of the coordinate system lies at the position of the sensor.
- *Construction*: The construction coordinate system is fixed on the vehicle body and therefore independent of the vehicle's dynamic behavior. The orientation of this coordinate system is defined in the norm ISO 4130. It will be used to describe the mounting position of different sensors in a vehicle by the pose ${}^{Construction}\mathcal{P}_{Sensor}$.
- *USK*: By contrast, the Environment Sensor Coordinate System (German: *Umfeldsensorkoordinatensystem*) (USK) [1] additionally incorporates the position and state of the vehicle in the environment. The axis directions are defined according to the norm DIN 70000, the origin of this system lies at the projection of the rear axis' center on the ground. As illustrated in figure 2.1, the orientation of the x-axis depends on the current *local supporting surface*. Consequently, also the current spring deflections and resulting pitch and roll angles of the vehicle are included in the pose ${}^{USK}\mathcal{P}_{Construction}$.
- *Map*: All map-based environment representations derived in this thesis will be defined by using a separated map coordinate system. The location of the maps and their reference system will be related to the USK. The pose ${}^{Map}\mathcal{P}_{USK}$ will also be applied to infer the position of the estimated map contents relatively to the vehicle.

[1] In order to avoid confusions with ESC, the German abbreviation will be used in this thesis

- *Global*: The world coordinate system will be used as a reference system for all previously described coordinate systems. Instead of a formally defined geodetic coordinate system as e.g. WGS 84 [47], an abstract, earth-fixed system on a tangential plane will be assumed. The aim of this simplified concept is to uniformly describe transformations of any other coordinate system, e.g. changes of global map poses ${}^{Global}\mathcal{P}_{Map}$.

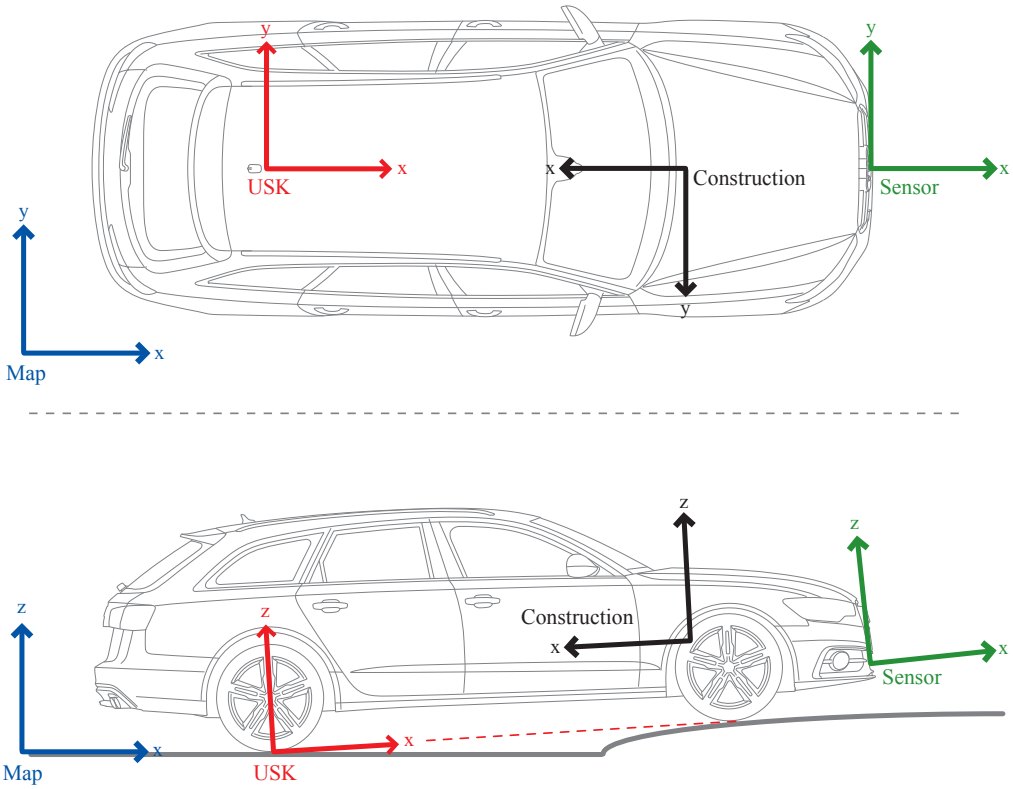


Figure 2.1.: Illustration of coordinate system definitions in side and top view.

2.3. Environment Perception Subsystems

As already stated in the introduction, all software modules, which were implemented in the scope of this work, were embedded into an existing environment perception architecture, see figure 1.5. The implementation of the modules was realized in C++ by using the ADTF, a software, which was created to simplify prototypical software development for driver assistance system. For details about ADTF, the interested reader is referred to appendix C and [172]. In the remainder of this chapter, the already existing subsystems of the environment perception architecture and especially their interfaces to the map-based environment representations will be explained.

2.3.1. Ego Motion Estimation

The detailed knowledge about the movement of one's own vehicle is a crucial source of information for interpreting sensor measurements and establishing environment representations. For this purpose, the ego motion estimation module, which has been introduced in [8], was applied in this work. In summary, the ego motion estimation builds on a fusion of the measurements from a low-cost automotive 6-D Inertial Measurement Unit (IMU) sensor cluster, the wheel speed sensors and the steering angle sensor. In order to quantify the pose of the IMU to the road surface, also the measurements of the chassis lift sensors are considered. As all incorporated sensors are part of the standard equipment of current upper class vehicles, no additional hardware requirements result from using this module.

To determine the provided state variables, the ego motion estimation uses a discrete Kalman Filter and a simple tire height estimation model. In this way, the pose of the USK in the global coordinate system can be inferred:

$$^{Global}\mathcal{P}_{USK} \tag{2.7}$$

The global coordinate system is reset during the initialization of the system. Furthermore, also the pitch and roll angle of the vehicle body to the current local supporting surface are provided. These angles are especially important to infer sensor alignments to the road surface.

Based on interpolations of the localization history, the ego motion module is also capable of providing the transformation of the USK coordinate system between two specified timestamps, e.g. two map update time steps $k-1$ and k:

$$^{(USK_k)}\mathcal{P}_{(USK_{k-1})} \tag{2.8}$$

This pose quantifies the position of the last cycle's USK in the current USK, which is of particular importance to the compensation of ego-local environment representation between two updates.

The inaccuracies of the ego motion estimation and their influence on the quality of the obtained environment representations will be discussed in the evaluation sections of chapters 3, 4 and 5.

2.3.2. Model-based Object Tracking Module

The model-based object tracking module from the perception architecture 1.5 has been described in [SWBH12]. The provided moving objects hypotheses will be utilized in several chapters of this thesis. Especially the close interaction between this module and the newly developed occupancy map will be discussed in detail in section 3.4.3. The provided object interfaces will be explained in the following. For the internally applied detection, association and estimation algorithms, the interested reader is referred to [SWBH12].

All detected moving objects are represented by using the "best knowledge model" [SWBH12], which allows to fuse object hypotheses from multiple sensors with different point of views. The goal of this approach is to represent the knowledge about a detected object without making unnecessary assumptions about its dimensions. Therefore, each sensor represents

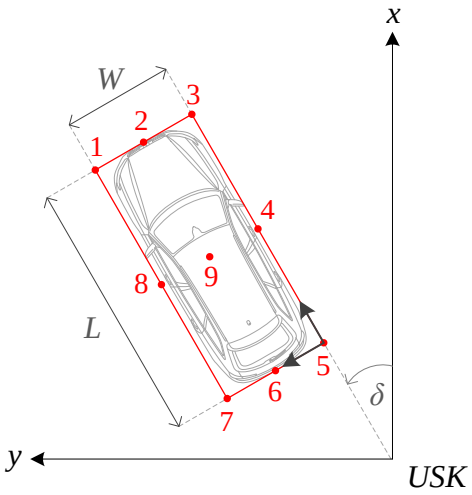


Figure 2.2.: Moving object representation with RPs. All possible RPs are indicated by red dots, the local coordinate system resulting from RP 5 is highlighted.

the dimensions of an object in a coordinate system which is considered most appropriate. For this purpose, one of nine possible coordinate system origins, called RP, can be chosen, as illustrated in figure 2.2. Each moving object is then described by a bounding box model, which refers to the chosen RP. The entire object state vector consists of:

$$\boldsymbol{o} = [x, y, v_x, v_y, a_x, a_y, \delta, L, W]^T \tag{2.9}$$

with v and a denoting the velocity and acceleration. The remaining state variables are illustrated in figure 2.2. All provided state variables, including the motion parameters, are quantified in relation to one's own vehicle. With the exception of the bounding box length and width, the object tracking module also provides the estimation variances of all inferred state variables.

2.3.3. Reference System

For the evaluation of newly created environment representations, a reference sensor system, which was introduced in [159, 8], will be applied. This system offers the opportunity to simultaneously record ground truth environment data and real sensor measurements in dedicated test scenarios. A comparison of these data can be used to quantify the quality of sensor measurements and resulting environment representations. In doing so, also the algorithms' ability to deal with characteristic measurement errors can be analyzed. This

is an important advantage over simulation-based evaluation techniques, which typically lack in exactly modeling sensor characteristics.

In general, this approach is based on Differential GPS (DGPS) measurements of the own vehicle, specially equipped other vehicles and surrounding static objects. In order to overcome the limitations of DGPS concerning the availability and measurement rate, the vehicles are additionally equipped with a highly accurate IMU. A combination of both techniques results in localization errors within a few centimeters [159]. As these measurements are still subject to measurement noise, the notions *reference* or *ideal* environment data will be used instead of *ground truth* environment data in this thesis.

Figure 2.3 gives an impression of the structure and data flow of the reference system. During the recording process, the position estimates of several moving objects can be instantly transmitted to the ego vehicle to simplify the monitoring and further processing. Together with the previously parametrized vehicle dimensions, the global position

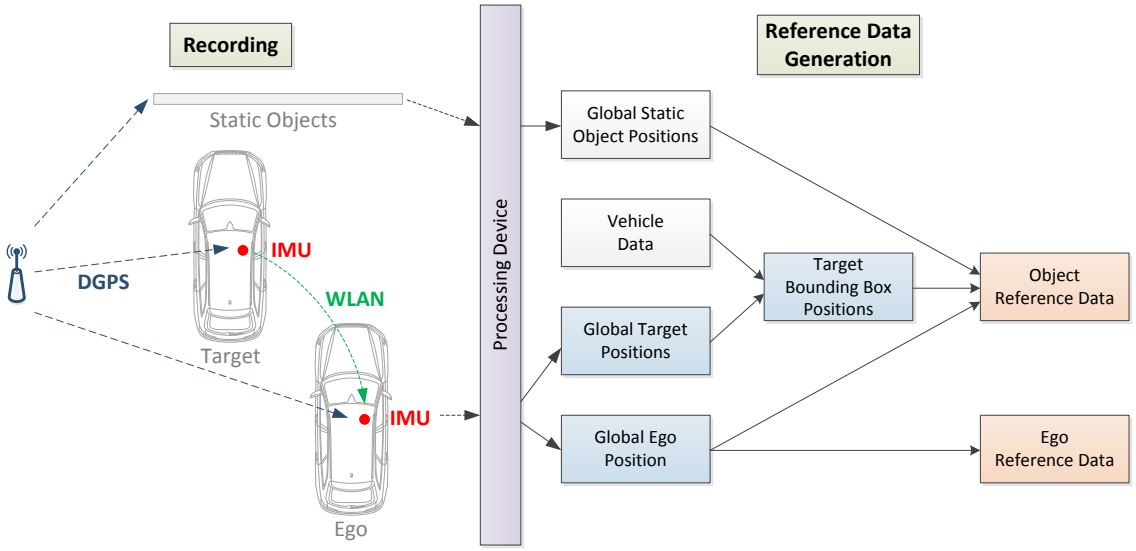


Figure 2.3.: Illustration of recording and processing of reference sensor data.

estimates of static and dynamic objects can be used to infer an accurate bounding box representation of the environment in the USK coordinate system. Besides that, the reference system also provides an accurate estimate of the ego vehicle motion.

When using this approach, it must be remembered that the reference description only contains previously measured static objects and specially equipped moving objects. In

order to reduce the hardware and measurement effort, this technique has to be restricted to less complex scenarios with structured environments and a small number of surrounding vehicles.

3. Representation of Occupancy Information

The development of a simplified environment representation of occupancy information for ADAS applications is subject of the first main chapter of this thesis. According to the structure proposed in the introduction, this chapter will discuss related works on this topic, develop a novel representation, suitable sensor models as well as additionally required compensation mechanisms, and finally evaluate the obtained results against the state of the art.

3.1. Related Works

The first section of this chapter discusses related works on environment representations of occupancy information. State-of-the-art approaches and proposed optimizations will be categorized and discussed concerning the previously identified requirements on environment representations.

3.1.1. Categorization and Discussion

Overall, numerous map- and model-based descriptions of occupancy states for robotic and ADAS applications have been proposed. Figure 3.1 gives a rough categorization of published approaches based on their level of abstraction respectively model assumptions and the considered dimensionality. According to the characteristics identified in the introduction, map-based approaches can be found in the lower part of this figure, whereas model-based object representations are located in the upper part. In between, there also exist combined methods that use principles from both major categories. Additionally, figure 3.1 lists point clouds representations (e.g. [25]), which will not be considered in detail due to their tremendous memory consumption and interpretation effort. The discussion will start with two- and three-dimensional occupancy grids and then focus on tree- and model-based object representations.

A) 2D Grid Maps

The approach to map the surroundings of an autonomous mobile robot was presented for the first time by Moravec and Elfes in 1985 [109]. Their basic idea was to systematically combine sonar range readings in a two-dimensional map, whose content improves by incorporating measurements from multiple points of view. The well-known term *occupancy grid*

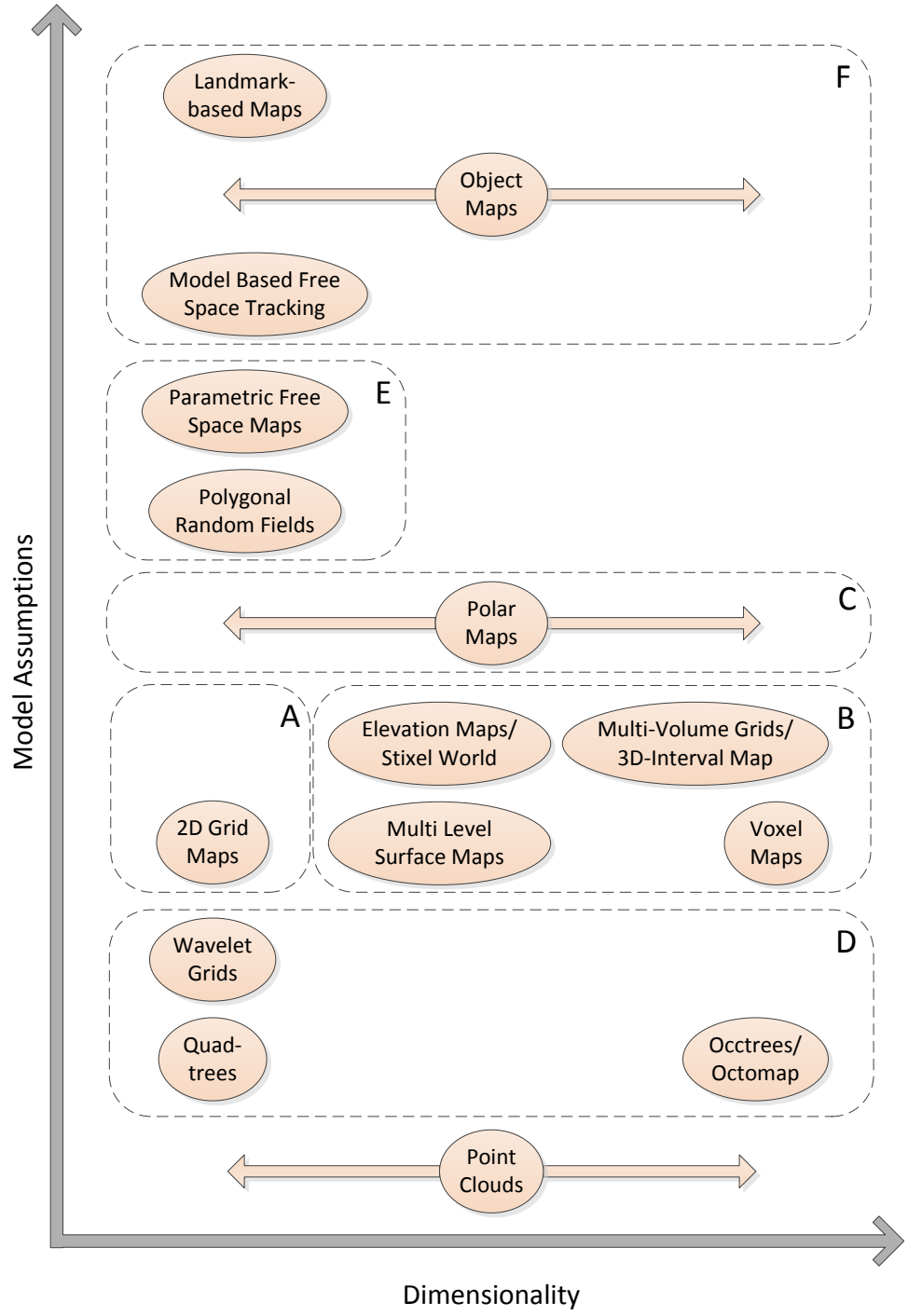


Figure 3.1.: Categorization of published representations for occupancy information.

was first used by Elfes in [46], where also a Bayesian estimation of occupancy probabilities was presented. Occupancy grid maps have become a de facto standard for representing the occupancy states in the environment, both in the robotics and intelligent vehicles community. The goal is to estimate each grid cell's occupancy probability between 1 for *occupied* and 0 for *free*, which signalizes that a vehicle can pass the cell. In the simplest case, the algorithms are based on the assumption of a pure static environment and perfectly known poses of one's own vehicle. The most common approach is to decompose the problem of estimating the posterior probability of a map $\boldsymbol{M}$

$$p\left(\boldsymbol{M} \mid \boldsymbol{z}_{1:t}\right) = p\left(o_{1,1}, \cdots, o_{L,C} | \boldsymbol{z}_{1:t}\right) \tag{3.1}$$

into the separated estimation of the cells' posterior occupancy probabilities [165]

$$p\left(\boldsymbol{M} \mid \boldsymbol{z}_{1:t}\right) = \prod_{l,c=1}^{L,C} p\left(o_{l,c} \mid \boldsymbol{z}_{1:t}\right) \tag{3.2}$$

with $\boldsymbol{z}_{1:t}$ being the sensor measurements between the first and current timestamp and $o_{l,c}$ denoting the occupancy state of the cell in line l and column c. In doing so, a statistical independence between the grid cells is assumed. This assumption prevents the modeling of dependencies among neighboring cells, but significantly reduces the computational effort. The estimation of a single cell's static binary state can be solved efficiently by using the inverse sensor model approach in the log odds formulation, see equation (A.68) in appendix A.2.6. In this way, the update of the posterior occupancy probability simplifies to a single addition. To overcome the limitations that result from the independence assumption, Thrun proposed to learn occupancy grids with forward sensor models, see equation (A.61) [163]. This approach requires the application of an expectation maximization algorithm to maximize the overall likelihood of a map given a set of sensor measurements. Although experimental results indicate substantial improvements, this algorithm is not able to work incremental and is not real-time capable.

An alternative grid map estimation scheme is given by the application of the theory of evidence, developed by Dempster [31] and Shafer [148]. The basic principle of the Dempster-Shafer Theory (DST) is to assign *belief masses* to the power set of possible states. The corresponding formulas are derived in appendix A.3, an application of the DST for occupancy grid mapping can be found in [42, 110, 87]. In contrast to the Bayesian estimation of posterior occupancy probabilities, the DST allows for a distinction between conflicting and missing knowledge in the map. On the downside, the memory consumption of the grid map increases by a factor of 3 and the map update is significantly more complex.

A rather unknown formulation of estimating occupancy states in grid maps are *Fuzzy Maps*, introduced by Oriolo et. al [117]. This approach is based on two fuzzy sets $\mathcal{F}$ and $\mathcal{O}$ and their membership functions $\mu_{\mathcal{F}}(x)$ and $\mu_{\mathcal{O}}(x)$, which describe free and occupied areas. In this context, the task of mapping corresponds to computing the membership functions for each cell of the fuzzy grid map. This methodology allows to represent conflicting

information and dependencies between neighboring cells, but also increases the complexity of the data structure respectively map update.

All grid-based representations provide the possibility to realize efficient ego-local maps. Therefore, the motion of the vehicle between two map updates can be subdivided into a translational and a rotational component. The translation of the map can be compensated by using a two-dimensional circular buffer and subcell values [16], as illustrated in figure 3.2. A rotational change can be used to update the pose between the vehicle and map ${}^{USK}\mathcal{P}_{Map}$. In this way, complex and inaccurate interpolations can be avoided, as Weiss discusses in [182].

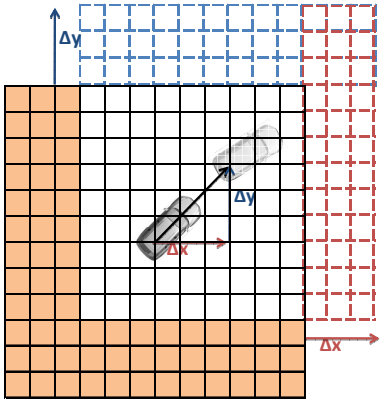


Figure 3.2.: Efficient ego motion compensation in grid maps, from [16].

The accuracy of grid maps mainly depends on the cell discretization. The cell size also impacts the computational effort of the mapping algorithms, as it influences the number of cells that have to be processed during the map update. Especially when using the log odds formulation and the independence assumption, large occupancy grids can be calculated in real-time [16], which is one of the main reasons for their popularity.

One of the major drawbacks of grid-based environment representations is the high memory consumption and the lack of scalability. A grid map is not able to provide information in different accuracies, as the cell sizes are fixed. Consequently, neither specific requirements of driver assistance functions, nor characteristics of the environment can be exploited. The high amount of cells also influences the interpretability of the data structure. To infer the required information, numerous cells have to be processed. When using rotated ego-local maps, also the alignment of vehicle and cells differs.

B) 2.5D and 3D Representations

In numerous publications, the grid-based mapping concept has been enhanced to represent three-dimensional environment information. They can be classified into 2.5D maps and full 3D representations.

2.5D maps are restricted to the description of height information instead of a complete three-dimensional representation of the environment. The most simple approach is given by *elevation maps*, which estimate a single height value of the surrounding surface in each grid cell, e.g. to extract occupancy information from 3D data [113], classify road surfaces [116] or detect road curbs [82]. Himmelsbach et al. use a similar elevation grid as intermediate representation to infer height information from 3D laser point clouds [65, 73], which is not used for an accumulation process.

Besides these applications of elevation maps in intermediate representations, there also exist publications that use maps for a temporal filtering of detected height values. Hong et al. accumulate height values for outdoor applications from preprocessed laser and stereo data by using a weighted average filter and measurement confidence values[72]. A more advanced approach is presented by Bouzouraa in [16], where up to three height hypotheses are estimated in a cell by using a one-dimensional Kalman Filter (KF). The resulting estimation variances quantify the reliability of the hypotheses.

Concerning their computational effort, scalability and discretization errors, elevation maps can be evaluated similarly as 2D grid-based environment representations. Although these data structures are often used as an intermediate representation, the capability of elevation maps to accumulate uncertain measurements has been demonstrated. Supposing a driver assistance function that requires information about free areas, an additional interpretation step has to take place. On the other hand, the detailed information about height profiles can improve the behavior of a highly automated vehicle, for example by detecting road curbs.

Nevertheless, elevation maps are not able to describe occupied volumes within a cell. To overcome this issue, Thrun et al. proposed the generation of 3D maps using polygonal models in a grid-based data structure [164]. In the *multi-level surface map* approach of Triebel et al., a cell can contain multiple surface patches, represented by the mean and variance of the measured height and an optional depth [167]. A similar method is demonstrated by Bouzouraa in [16]. The *3D Interval Map* represents several rectangular bars in a grid cell. The update of existing bars is based on KFs for the borders and an additional reliability measure that is derived from the number of measured laser points. In [126], Pfeiffer also extends the stixel world to allow several stixels along an image column. Still, the methods mentioned so far are not able to represent three-dimensional free space. Dryanovski et al. solve this issue in their interesting work about *multi-volume occupancy grids* [41]. In each cell of this map, the measured occupied and free space is represented in two separated lists of positive and negative volumes. In order to extract probabilistic occupancy information for a certain point in space, the occupancy densities of the existing positive and negative volumes have to be combined.

More detailed full 3D representations are given by *Voxel*[1] *Maps*, which transfer the concept of creating equidistant cells to the third dimension. First implementations for robotic indoor environments have been demonstrated by Moravec [108], similar papers deal with

[1] Volumetric Pixels

robotic outdoor environments [51, 121] or construction sites [160]. Another familiar application area is to warp textures from camera images onto voxel maps [100]. In [18], Bouzouraa also introduced an ego-local voxel map for an application in driver assistance systems. According to the presented results, a $140 \times 140 \times 6$m map can be processed online.

In conclusion, voxel maps show similar advantages and disadvantages as two-dimensional grid-based environment representations. Of course, the memory demand further increases by considering additional dimensions. Moreover, also extractors have to deal with a substantially increased amount of cells. Multi-level surface maps and multi volume occupancy grids mitigate this problem at the expense of more complex update operations.

C) Polar Maps

Another category of map-based environment representation uses polar instead of the Cartesian grids. These data structures are especially suitable for preprocessing radar, laser and video sensor measurements, which are typically provided in polar coordinates due to their measurement principles. Kang et al. used a probabilistic volume polar grid map to represent detected free spaces and obstacles from stereo vision [79]. According to them, a polar grid can help to simplify the measurement analysis and free space computation.

Similar ideas are used in the *Stixel World* [7], which aims to infer a compact and flexible representation of three-dimensional traffic situations in stereo camera disparity images. It is based on the model assumption that the free space around the vehicle is limited by a set of vertical rectangular sticks (*stixels*). In [125], Pfeiffer et al. also extended the approach to track stixels that belong to dynamic objects over several frames.

To develop their strengths, polar maps have to be used ego locally and aligned with the applied sensor. On the downside, ego local polar maps are not able to accumulate information, as the transfer of information between two transformed polar grids would require complex interpolation calculations. Concerning the remaining requirements, polar maps can be evaluated similarly as grid-based representations. Supposing a suitable alignment, the interpretation effort decreases as the traversability of the data structure simplifies [79].

D) Tree-based Representations

To overcome the inflexible and high memory consumption of occupancy grids, tree-based data structures for occupancy representations have been examined extensively. In this context, the root node of a tree represents the whole area that has to be covered by a map. The tree's leaf nodes describe the occupancy state of a partition of the overall space. Each inner node of the data structure is recursively subdivided into a set of child nodes. Depending on the number of child nodes, existing approaches can be classified by the N^d formulation proposed in [43], which states that the volume of an inner node is split up into N intervals in d dimensions. Common two-dimensional representations use 2^2-trees,

which are also denoted as *quadtrees* [90], e.g. illustrated in figure 3.3. Three-dimensional approaches are typically realized by *octtrees*, which correspond to 2^3-trees, e.g. [122, 189].

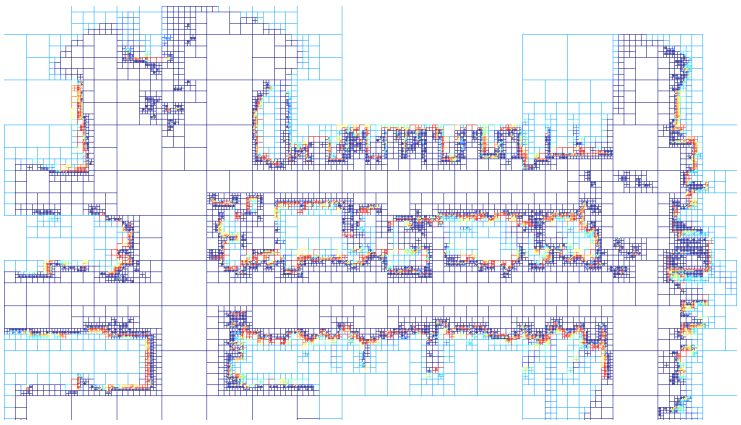


Figure 3.3.: Leaf nodes of quadtree map representing occupancy information, from [44].

A sophisticated approach for 2D and 3D robotics applications was presented by Einhorn et al. in [43, 44]. Depending on the measurements, the presented mapping approach adapts the depth of the tree online. Therefore, a histogram of *hits* and *misses* is calculated in each tree node. If this histogram deviates considerably from the expected characteristic of a free or occupied cell, the node will be split up. According to the presented results, the number of cells is reduced by a factor of 7 compared to fixed resolution mapping. Although a performance gain of the factor 2-3 is pointed out, absolute computation times are not reported. Schmid et al. demonstrated the application of similar tree-based representations for ADASs [143, 142, 141]. In [141], Schmid proposes different strategies to control the depth of the underlying octtree data structure: For an application in pre-crash safety systems, a stepwise level of detail is formulated, depending on the distance from the own vehicle. For the detection of landmarks, the resolution is increased in dedicated areas. Otherwise, a dynamic level of detail is proposed, which is based on the DST evidences of free and occupied areas. For a 120×120 m 2D map with a minimal cell size of 0.1 m, an average computing time of 10 ms per cycle is reported, a 3D map with equal dimensions and a minimal cell size of 0.2 m can be processed within 10 ms. Both variants allocate in average between 1 and 1.5 MB of memory.

A rather unknown approach in this category are *Wavelet Occupancy Grids*, proposed by Yguel et al. in [191, 192]. The basic idea of this technique is to replace the occupancy probabilities of a grid map by a set of Haar wavelet functions. Due to the hierarchical structure of the Haar wavelet basis, the resulting map can be interpreted as a quadtree in 2D, respectively an octtree in 3D. Yguel et al. point out that this representation saves about 90% of memory compared to an occupancy grid, while the computing time is "sufficient" for a real-time processing in 2D and 3D [192].

To sum up, the application of tree-based representations shows great advantages concerning the requirements on scalability and memory consumption. Both 2D and especially 3D data structures show a significant reduction of the memory consumption. As Schmid and Einhorn show in their work, the accuracy and the required resources can be adapted to the characteristics of the surroundings and to specific requirements of driver assistance systems. On the downside, tree-based maps are confronted with a further increased computational and interpretation effort. Although Schmid and Einhorn introduce advanced node addressing schemes, major parts of the tree structures have to be traversed for updating and extracting information. This task is usually more complex and less cache-efficient than rasterization algorithms for grid-based data structures.

E) Combined Map and Model-based Representations

According to the definitions of section 1.3.1, all representations discussed so far can unambiguously be classified as map-based environment representations. Between both major categories, there also exist approaches, which try to combine the advantages of both philosophies.

In [120], Paskin and Thrun developed a mapping algorithm with *polygonal random fields*, which is inspired by the polygonal coloring problem in 2D planes. A polygonal coloring is defined by a start anchor point and a set of edges, which represent the boundaries between black and white regions. To obtain a map with edges between occupancy states, the likelihood of the map given a set of range measurements is optimized by using a sampling algorithm. Paskin et al. claim that the resulting vector representation avoids discretization errors and can represent simple environments more compactly. On the other hand, their approach is not able to process sensor measurements online and implies a huge interpretation effort.

Andert et al. developed a combined grid- and feature-based occupancy map for large outdoor environments [3]. This approach combines local occupancy grids with a global feature-based map. Between both representations, bounding box shapes can be exchanged. By contrast, Pandey et al.'s *feature-based occupancy grid* is based on a single occupancy grid with enhanced map update procedure [119]. Before the update takes place, range measurements are associated with predefined features in order to reduce the impact of the grid cell independence assumption.

In [144, 145], Schreier et al. present their interesting work on *parametric free space maps*. Their goal is to develop a compact and function-independent representation of free space information. Therefore, they obtain an intermediate occupancy grid, from which the detected free space is extracted by a combination of several familiar image processing techniques. The outer boundary of the extracted contour is represented by a temporally filtered B-Spline, inner boundaries are additionally described by geometric primitives.

In conclusion, Andert's and Pandey's contributions do not change the underlying occupancy grid but improve the applied sensor models and the extraction process [3, 119]. Concerning the identified requirements, both approaches can be evaluated as occupancy

grids. Similarly, Schreier uses an intermediate occupancy grid to estimate the contours of the free space around the vehicle. Although this approach results in a compact environment representation, the authors admit that the computational effort of the mapping procedure is increased, as it includes the creation of an intermediate occupancy grid. To the author's knowledge, a direct estimation of spline-based representations without using intermediate grids has not been considered so far, although there exist promising computer vision approaches e.g. advanced active contour models [32].

F) Model-based object representations

Especially for indoor robotic applications, also purely model-based object representations have been proposed. One class within this category is the *landmark-based* mapping approach, which limits to the description of significant features in the environment [161, 162, 4]. These maps provide a highly compact description but only aim at providing input data for SLAM applications instead of creating a complete representation of the surroundings. Besides that, there exist several approaches which use geometric primitives to provide a complete 2D or 3D image of the surroundings, which can be summarized by the term *object maps* [162]. So called *Line maps* are especially suitable to describe structured indoor environments [21, 171, 94]. On the downside, they usually require an offline processing after a set of measurements has been collected from a predefined area.

Basic model-based object representations have difficulties in describing semi- or unstructured features in outdoor environment, e.g. the free space in an urban scenario. They either use limiting model assumptions or have to cope with a huge number of object hypotheses, which eliminates the main advantages of this approach. Nevertheless, model-based object representations have been applied to describe occupancy states for driver assistance systems. In [96], Lundquist et al. introduce an object-based representation of road boundaries. In [17], Bouzouraa also presents a model-based approach for free space tracking. The free space in front of the vehicle is described by a varying number of parallel rectangles, whose lateral borders are estimated by a 1D KF. In order to compensate the ego-aligned rectangles, the impact of the vehicle's rotational movement is approximated. This approach demonstrates, how model-based principles can be enhanced to describe unstructured environment features, and serves as an origin for the development of the 2DIM in the following sections.

3.1.2. Summary

Table 3.1 sums up a qualitative evaluation including the advantages and disadvantages of the discussed representations. In summary, there is no approach which simultaneously fulfills the requirements identified in chapter 1.2.2.

	Accur.	Unc.	Accum.	Scal.	Int.	Mem.	Com.
Point clouds	++	−−	−	−	−−	−−	−−
2D Grid Maps	+	+	++	−	−	−	+
Elevation Maps	+	o	+	−	−	−	+
Multi-Volume Grids	+	o	+	−−	−	−	o
Voxel Maps	+	+	++	−−	−−	−−	+
Polar Maps	+	+	−−	−−	+	−	+
Quadtrees/Octtrees	++	+	+	++	−−	+	−
PRF Mapping	+	−	−−	−−	−−	o	−−
PFS Maps	o	−	−−	−	o	+	−
Object Maps	−−	−	−	+	+	++	+
Model-Based FST	−	−	o	+	+	+	++

Table 3.1.: Evaluation of representations for occupancy information.
(Accur. = Accuracy, Unc. = Uncertainties, Accum. = Accumulation, Scal. = Scalability, Int. = Interpretability, Mem. = Memory consumption, Com. = Computation times)
(PRF = Polygonal Random Fields, PFS = Parametric Free Space, FST = Free Space Tracker)

3.1.3. Occupancy Grid Optimizations

In order to eliminate the high resource requirements of occupancy grids, further enhancements, which do not focus on the representation itself, have been proposed. One possibility to improve the performance of computationally complex algorithms is the application of massively parallel GPU hardware. Yguel et al. presented an approach, in which occupancy grid mapping is formulated as texture mapping problem, which also helps to eliminate ray casting *moire effects* [190]. According to the experimental results, the application of GPU hardware allows to fuse up to 13 high resolution laser sensors in real time. Similar work is presented by Homm et al. in [71]. The reported computation times for updating a 512×512 cells occupancy grid with radar and laser measurements are in the range of 5-6 ms using a GeForce 285GTX GPU.

A different approach is shown by Grewe et al. in [61, 62]. They propose a generic architecture, in which several sensors calculate intermediate occupancy grids which need to be combined in a global fusion grid. In order to reduce the computational effort and bandwidth requirements for data transmission, Grewe et al. propose a lossless compression of occupancy grids by incremental difference data structures and run length encodings.

Both approaches try to reduce the impacts of grid mapping algorithms instead of improving the underlying data structure itself. The application of GPU hardware offers a great

potential to reduce the overall computation times. Nevertheless, this technique requires an in-depth extraction concept for transferring data from and to the GPU, which is one of the major bottlenecks in these architectures. Besides that, reasonably priced and energy-efficient GPU hardware for safety critical calculations in automotive control units is not yet in sight. Grewe's approach to compress and exchange entire occupancy grids seems inappropriate considering well-established methods that infer high-level descriptions from map contents, e.g. free spaces and road boundaries [182, 141, 181, 71, 30, 16, 144, 145, 68, 135].

3.2. Development of a Simplified Representation for Occupancy Information

In order to overcome the discussed shortcomings of existing approaches, a newly developed, two-dimensional interval-based representation will be proposed in the following sections. This novel approach forms the key contribution of this thesis and will also be reused in chapters 4 and 5. The following section will introduce the basic principles of the generic environment representation, after that it will be shown, how this data structure can be used to represent occupancy information.

3.2.1. Interval-based Environment Representation

In order to approach the goal of a simplified representation with low interpretation effort, the requirements of automated vehicle behavior generation and control have to be understood. In general, the vehicle control is subdivided into a lateral component, which is influenced by manipulating the steering, and the longitudinal control, which is realized by accessing throttle and brake signals. First of all, this implies that the lateral movement can be controlled considerably more accurate than the longitudinal one. Second, most existing behavior planning approaches determine the desired longitudinal motion of the vehicle by evaluating moving target objects, detected road markings or speed limits. Map-based representations of free spaces and obstacles are typically used to optimize the lateral positions of possible trajectories, e.g. to avoid obstacles in safe distances. If we restrict to longitudinal traffic scenarios, vehicle automation hence requires a higher resolution of occupancy information in lateral direction, whereas inaccuracies and simplifications in longitudinal direction can be accepted to a certain extent.

Consequently, the key idea of the following simplified representation is to subdivide the 2D space around the vehicle into intervals, which are aligned orthogonal to the driving direction. In contrast to grid-based environment representations, this approach discretizes the surrounding space only in longitudinal direction, whereas the lateral positions of environment features can be represented by continuous values. In this way, this approach simultaneously improves the lateral accuracy and simplifies the interpretation of map con-

tents on higher levels of abstraction. The underlying geometrical structure is derived from a skeleton of 2D nodes $\boldsymbol{p}$ in the USK coordinate system [2]:

$$\boldsymbol{S} = \left[\boldsymbol{p}^{(0)}, \cdots, \boldsymbol{p}^{(n)}, \cdots, \boldsymbol{p}^{(N+1)}\right] \tag{3.3}$$

The positions of the nodes can origin from different sources. If there exists information about the course of the road (e.g. from digital map information or the road infrastructure detection module, see perception architecture 1.5), maps can be "attached" to a *reference path*, e.g. the center line of the currently used traffic lane. Otherwise, a straight continuation is assumed, which results in equidistant nodes along the x-axis of the current USK. Both variants are illustrated in figure 3.4 and will be referred to as *curved* and *straight* 2DIM. The idea of the straight 2DIM has already been presented in the patent application [67] and in [WBH12, WBH13].

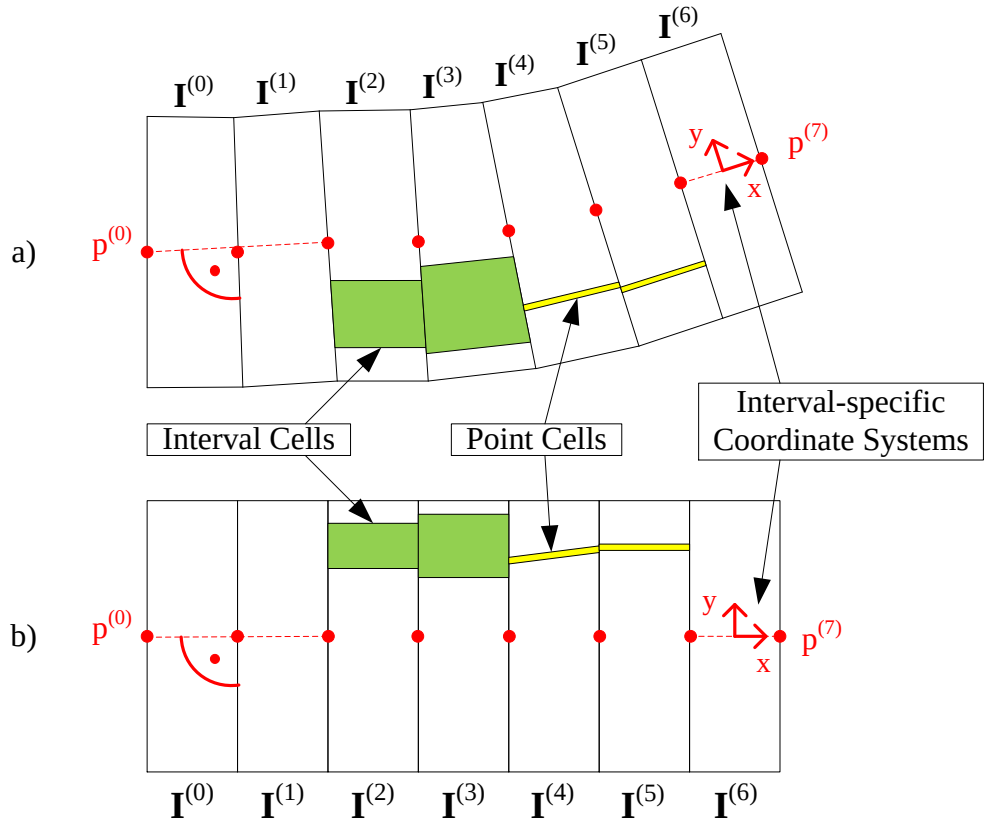


Figure 3.4.: Geometrical structure of a) curved and b) straight 2DIM.

[2] Time index k will be left out for clarity

The nodes of the polygonal line define the center position of the borders between the intervals. The alignment of the borders $\boldsymbol{b}$ is given by the normalized orthogonal vector to the central difference gradient at node n, as also illustrated in the lower part of figure 3.4:

$$\boldsymbol{b}^{(n)} = \frac{\boldsymbol{c}^{(n)}}{\left\|\boldsymbol{c}^{(n)}\right\|} \qquad \text{with} \quad \boldsymbol{c}^{(n)} = \begin{bmatrix} y^{(n-1)} - y^{(n+1)} \\ x^{(n+1)} - x^{(n-1)} \end{bmatrix} \tag{3.4}$$

The intervals will be denoted by the index of the lower border interval. The complete map at time step k is given by a tuple of N intervals, which provide a neighborhood relationship among each other:

$$\boldsymbol{M}_k = \left(\boldsymbol{I}_k^{(0)}, \cdots, \boldsymbol{I}_k^{(n)}, \cdots, \boldsymbol{I}_k^{(N)}\right) \tag{3.5}$$

In this structure, an interval can store several generic information containers of different types. A container that describes a punctual environment feature will be called *point cell*, while *interval cells* specify information with dimension, as also illustrated in figure 3.4. The position and shape of these containers will be described by using *interval-specific coordinate systems*, whose orientation is defined by the vector between two consecutive reference path nodes. In this way, the lateral position of the containers is represented orthogonal to the reference path, which implies an improved interpretability of the resulting map. The transformation from the interval coordinate system $I^{(n)}$ to the USK results as[3]:

$${}^{USK}\mathcal{P}_{I^{(n)}} = \mathcal{T}\left(\frac{x^{(n)} + x^{(n+1)}}{2}, \frac{y^{(n)} + y^{(n+1)}}{2}\right) \cdot \mathcal{R}_Z\left(\psi^{(n)}\right) \tag{3.6}$$

$$\text{with } \psi^{(n)} = \text{atan2}\left(y^{(n+1)} - y^{(n)}, x^{(n+1)} - x^{(n)}\right) \tag{3.7}$$

In case of the straight 2DIM, the interval borders are given by equidistant parallels to the vehicle's y-axis. Consequently, the transformation task simplifies to a translation by the longitudinal center of the interval:

$${}^{USK}\mathcal{P}_{I^{(n)}} = \mathcal{T}\left(\frac{x^{(n)} + x^{(n+1)}}{2}, 0\right) \tag{3.8}$$

3.2.2. Interval-based Representation of Occupancy Information

Within the scope of this work, two different concepts of representing occupancy information in the interval-based framework have been developed. Whereas the *cell-based approach* will be described and evaluated in-depth in the remaining sections of this chapter, the alternative approach to represent polygonal chains will only be briefly outlined as a suggestion for future works in the conclusion of this thesis, see section 6.1. The aim of

[3]The function atan2 denotes the two-argument inverse tangent function, which correctly distinguishes all four quadrants.

the cell-based approach is to model surrounding free and occupied areas by rectangular interval cells with variable widths. The semantics of the interval cells should be similar to conventional occupancy grid cells [165]: an *occupied cell* indicates the existence of an obstacle in the enclosed area, whereas a *free cell* can be passed by a vehicle without a collision. Certainly, these requirements allow a large number of possible cell specifications and estimation schemes.

The first design decision deals with the relation between free and occupied cells. The cells of an interval $\boldsymbol{I}^{(n)}$ can either be managed in a single list or in separated lists for each occupancy type, as for example used in the related problem of 3D multi-volume mapping [41]. One obvious drawback of keeping separated lists is the occurrence of conflicting information and blank areas, which requires a dedicated post-processing step to infer occupancy probabilities [41]. Even if two cells of different types do not conflict and exactly border on each other, the coinciding lateral positions of the borders have to be stored twice. To avoid these problems, the presented approach is based on a dense list of cells, which only contain a single position. In order to allow efficient insertions and deletions during the update procedure, the cells are organized in a doubly linked list [85].

The second design decision concerns the representation of uncertainties. In contrast to grid cells which only have to describe the uncertainty about the occupancy state, interval cells additionally have to quantify the confidence of the estimated border positions. For the sake of simplicity and an easy interpretation, a 1D Gaussian probability distribution, which is aligned orthogonally to the border, will be assumed. Thereby, a border between two cells can be reused for slightly different occupancy measurements and is able to represent a smooth transition between occupancy states. In order to enable an accumulation process, the cells also have to provide a measure of reliability. By using the posterior occupancy probability formulation, cells of different occupancy types can be treated equally and change their states over time. In this case, common occupancy grid state estimation derivations (A.57) can be reused.

These considerations lead to the interval structure illustrated in figure 3.5. Formally, an interval $\boldsymbol{I}^{(n)}$ is defined by a tuple of M interval cells, which together completely cover the map width:

$$\boldsymbol{I}^{(n)} = \left(\boldsymbol{OC}^{(n,0)}, \cdots, \boldsymbol{OC}^{(n,m)}, \cdots, \boldsymbol{OC}^{(n,M)}\right) \tag{3.9}$$

The state $\boldsymbol{x}^{(n,m)}$ of an occupancy cell $\boldsymbol{OC}^{(n,m)}$ is twofold. The first state variable describes the binary occupancy state, the second variable represents the continuous lateral end position in the interval coordinate system $I^{(n)}$:

$$\boldsymbol{OC}^{(n,m)} = \left(o^{(n,m)}, y^{(n,m)}\right) \tag{3.10}$$

Due to the dense sequence of cells, the representation of a single lateral position per cell is sufficient. As the first and last cell of an interval indicate the outer boundaries of the map, their lateral positions $y^{(n,0)}$ and $y^{(n,M)}$ form a special case and do not provide variances.

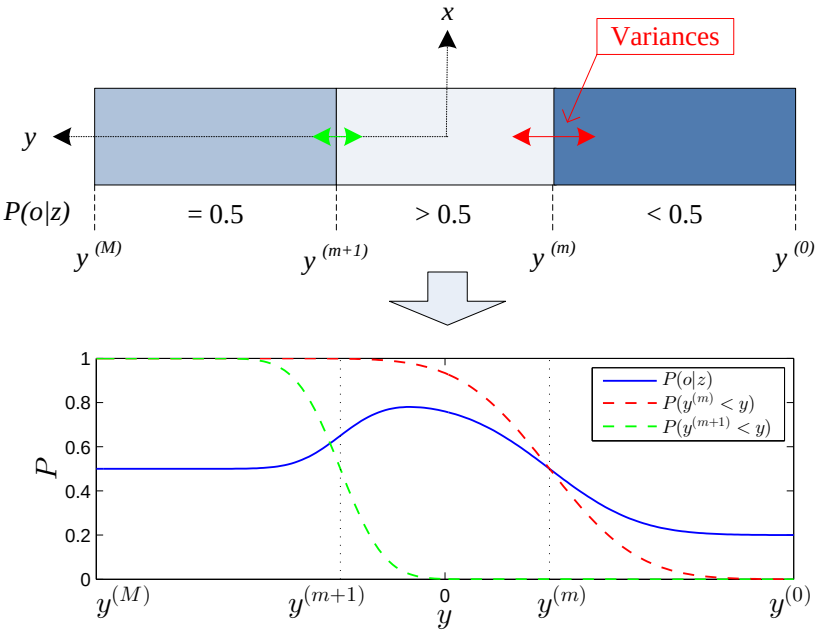


Figure 3.5.: Representation of occupancy cells in an interval and inferred continuous probability profile.

The represented knowledge about free and occupied areas can be used in two ways. On the one hand, the border uncertainties can be neglected to directly infer distinct distances between possible future trajectories and unknown and occupied areas. On the other hand, a continuous posterior *probability profile* along the interval can also be calculated. Assuming that the cell border position distributions are independent, the occupancy probability at any position y in an interval $I^{(n)}$ results from the combination of the cells' occupancy probabilities and border distributions:

$$P(o|\boldsymbol{z})(y) = \sum_{m=0}^{M-1} P(o^{(m)}|\boldsymbol{z}) \cdot P(y^{(m)} \leq y) \cdot P(y^{(m+1)} > y) \tag{3.11}$$

$$= \sum_{m=0}^{M-1} P(o^{(m)}|\boldsymbol{z}) \cdot \Phi\left(y; \hat{y}^{(m)}, \mathrm{Var}\left(\hat{y}^{(m)}\right)\right) \cdot \left(1 - \Phi\left(y; \hat{y}^{(m+1)}, \mathrm{Var}\left(\hat{y}^{(m+1)}\right)\right)\right) \tag{3.12}$$

with $\Phi(x; \mu, \sigma^2)$ denoting the cumulative distribution function (cdf) of the normal distribution with mean μ and variance σ^2. An exemplary resulting probability profile is illustrated in the lower part of figure 3.5. This interpretation will be especially useful for the transformation of knowledge between different map-based environment representations in chapter 5. Please note that the probability profile does neither represent a cdf nor a probability density function (pdf), as the occupancy probabilities along an interval do not integrate to 1.

3.3. Development of Sensor Models and Update Algorithms

An overview of the required subtasks that are necessary to recursively estimate the proposed interval-based occupancy map can be found in figure 3.6. The following sections will deal with the development of sensor models and update algorithms, which correspond to the blue steps in figure 3.6. The aim of these components is to realize the estimation of the map contents under consideration of the requirements concerning the computational efficiency and the regard of the measurements' uncertainties. After a discussion of possible forward and inverse sensor model implementations in this context, the required algorithms will be derived for the introduced occupancy representation. Due to the ability to measure free spaces and object contours, the presented approach will focus on the application of the laser sensor described in appendix B.3, but can also be adapted to support similar range measurement sensors.

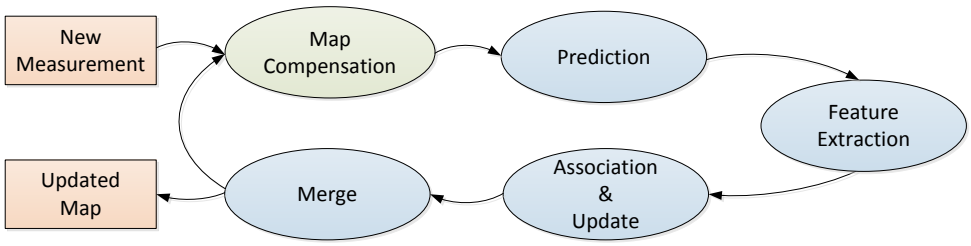


Figure 3.6.: Recursive update cycle of interval-based occupancy map.

3.3.1. Forward and Inverse Sensor Model Approaches

In general, two different probability distributions can be used to incorporate sensor measurements in recursive state estimations (see also section A.2): The forward sensor model probability $P(\boldsymbol{z}_k|\boldsymbol{x})$ and the inverse sensor model probability $P(\boldsymbol{x}|\boldsymbol{z}_k)$. Especially when estimating the occupancy state of a single space element, the space of possible states is significantly smaller than the space of possible measurements. Hence, it is usually more convenient to derive a distribution over both states given the current measurement, than determining the distribution over all possible measurements showing one of both states, which explains the wide dissemination of inverse sensor model occupancy mapping approaches [165]. The most popular forward sensor model approach extends the state space to the combined occupancy state of several cells and tries to find a map which yields the maximum sensor model probability [163]. In this general taxonomy, also the KF can be classified as a forward sensor model-based approach for continuous state space. In this case, the state with the maximum forward sensor model probability is derived by formulating an expected measurement and modifying the current state estimate according to the deviation between expected and real measurement [75].

Both principles can be transferred to the problem of estimating occupancy cells with variable sizes from range measurements. The resulting procedures are visualized in figure 3.7. The inverse sensor model approach, illustrated in the left part of the figure, tries to

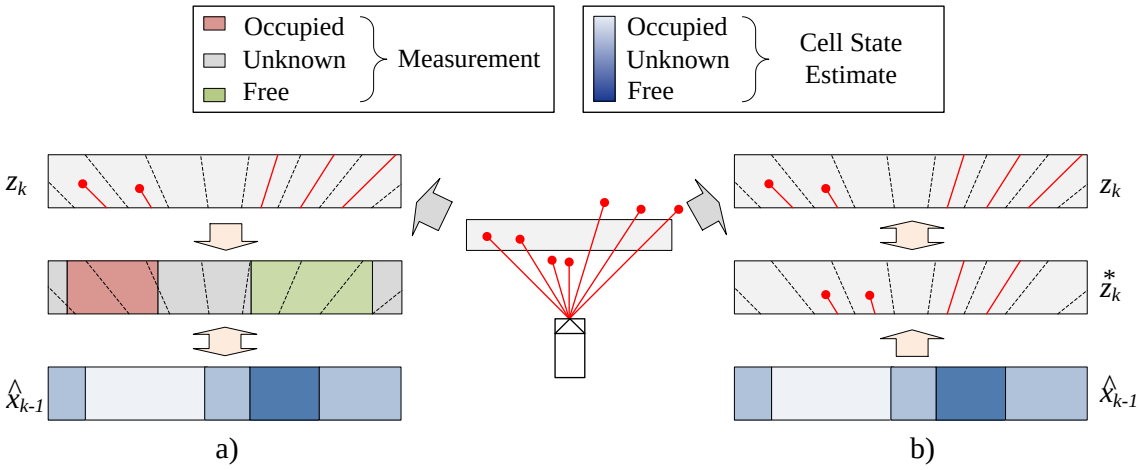


Figure 3.7.: a) Inverse and b) forward sensor model approach for occupancy cell estimation within an interval. Red lines and dots indicate laser beams and measurements.

estimate the cell states based on the measurement results within the interval. However, the widespread inverse sensor model formulation (see A.2.6) is limited to the estimation of static binary states, hence, this approach has to be restricted to the estimation of single occupancy states. Thus, an independence between the cells of an interval has to be assumed, which is one of the main criticisms of common grid mapping algorithms [163]. The right side of figure 3.7 shows a forward sensor model strategy, which is inspired by the KF. In this case, the last state estimate is used to derive the expected range measurements within an interval. The difference between the expected and real measurements can be used to optimize the cells' occupancy states and dimensions. On the one hand, this strategy would allow for estimating combined occupancy states without independence assumptions. On the other hand, the modification of occupancy states by quantifying deviations in the high-dimensional space of possible range measurements is a highly complex task. Furthermore, this method requires the development of a dedicated detection strategy, since e.g. newly measured obstacles cannot be associated with expected measurements. Due to these considerations and the previously introduced cell states, the developed update mechanism is twofold: The cells' borders are estimated by using a 1D Extended Kalman Filter (EKF), whereas the binary occupancy state of the cells is estimated by using the inverse sensor model approach for binary states.

3.3.2. Sensor Model and Feature Extraction

Both estimation principles require a preprocessing step to infer rectangular (straight 2DIM) or trapezoidal (curved 2DIM) occupancy cells from the raw laser range measurements obtained by the sensor described in appendix B.3. Therefore, the 3D multi-layer laser scan first has to be simplified to a 2D measurement structure. Assuming that all measurements within a channel cover the same horizontal angle range, several vertically differing measurements can be reduced to the shortest available distance. By the restriction to the obstacles that directly limit the measured free space, the computational effort can be significantly reduced without losing the most relevant information for trajectory planning. Overall, the measurement vector $\boldsymbol{z}_k$ is reduced to a number of C range and angle measurements:

$$\boldsymbol{z}_k = \left[z_k^{(0)}, \cdots, z_k^{(c)}, \cdots, z_k^{(C)} \right] \tag{3.13}$$

$$\text{with } z_k^{(c)} = \left[r_k^{(c)}, \vartheta_k^{(c)}, \varphi_k^{(c)} \right] \tag{3.14}$$

with r representing the radial distance, ϑ denoting the vertical angle and φ being the horizontal angle. All three values are subject to measurement errors: Inaccuracies in the provided distances result from distortions in the physical measurement process and discretizations during the signal processing. Uncertainties in the specification of a measurement's angle originate from the divergence of the laser beam, the reflection behavior of the obstacles and potential blooming effects. Besides that, also the yaw and pitch angle of the sensor's mounting position can only be determined to a certain degree of accuracy. In order to enable a constant transformation pose ${}^{USK}\mathcal{P}_{Sensor}$, these inaccuracies can also be expressed by the measurement angles.

For the following derivations, independent zero mean Gaussian noise terms for each individual measurement quantity will be assumed. The combination of the distributions' variances leads to the following measurement error covariance:

$$\boldsymbol{R}_k = \begin{bmatrix} \mathrm{Var}(r) & 0 & 0 \\ 0 & \mathrm{Var}(\vartheta) & 0 \\ 0 & 0 & \mathrm{Var}(\varphi) \end{bmatrix} \tag{3.15}$$

When restricting to measurements which result from objects that are aligned orthogonal to the laser beams, this error covariance suitably reflects the actually occurring deviations. Due to the non-consideration of ground measurements for the representation of occupancy information, the negative impacts of this simplification are limited.

For the deduction of the occupancy cells, the developed sensor model considers the horizontal field of view α of a measurement channel, which is mainly caused by the laser beam divergence, see figure 3.8. If we assume that all laser reflections were caused by obstacles that are aligned perpendicular to the ground plane, the projection of the measurements'

vertical extent results in a minor uncertainty range. As the vertical extent of a measurement also depends on several hardly identifiable factors, it will not be considered explicitly in the sensor model. The occurring inaccuracies can be compensated by $\mathrm{Var}(\vartheta)$.

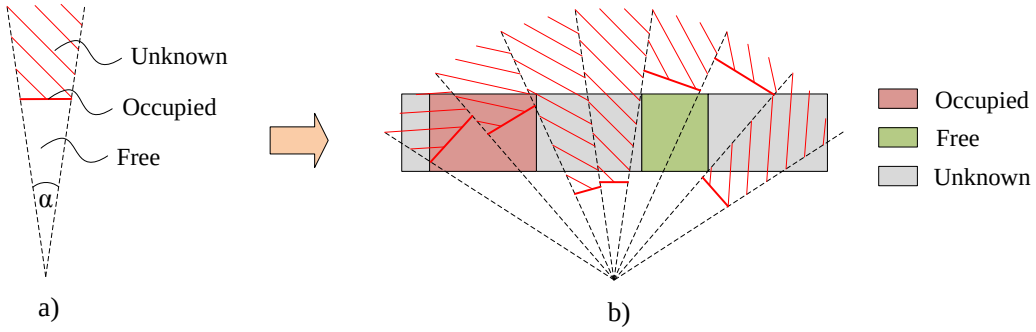


Figure 3.8.: Occupancy sensor model and feature extraction.

In each channel, the projected area in front of the measured distance r is considered free space, whereas the area behind the measurement has an unknown occupancy state. If the received echo is neither classified as ground nor clutter measurement, a reflecting obstacle is assumed transversely to the measurement cone in distance r. After a conversion of the polar measurements to the Cartesian USK, the inferred boundaries can be distributed along the intervals $\boldsymbol{I}^{(n)}$. The occupancy cells within each interval are extracted according to the following rules, which are also depicted in figure 3.8 b):

- The position and lateral extent of laser beam reflections directly determine the borders of *occupied cells*. Neighboring and overlapping obstacle measurements can be merged to enable a consistent data structure and reduce the further processing effort.

- Depending on the configuration of the 2DIM, *free cells* describe rectangular or trapezoidal areas that have been passed by laser beams without reflections. To obtain this information, all laser beams which completely pass an interval are grouped. The resulting regions are limited by the borders of neighboring occupied and unknown areas, as can be seen from figure 3.8. The procedure is a generalization of the model-based free space tracking algorithm proposed in [16].

- In order to obtain a dense data structure, the remaining areas are filled with unknown cells containing the a priori knowledge.

An exemplary extraction result is shown in figure 3.9. For each extracted cell, an inverse sensor model occupancy probability $P(o|\boldsymbol{z}_k)$ can be modeled. In the developed implementation, this probability is inferred by a combination of two familiar principles: On the one hand it is assumed, that the reliability of free space measurements decreases with increasing distance [165]. On the other hand, the ratio between the received and theoretically possible measurements is also considered, similar to the method developed in [16].

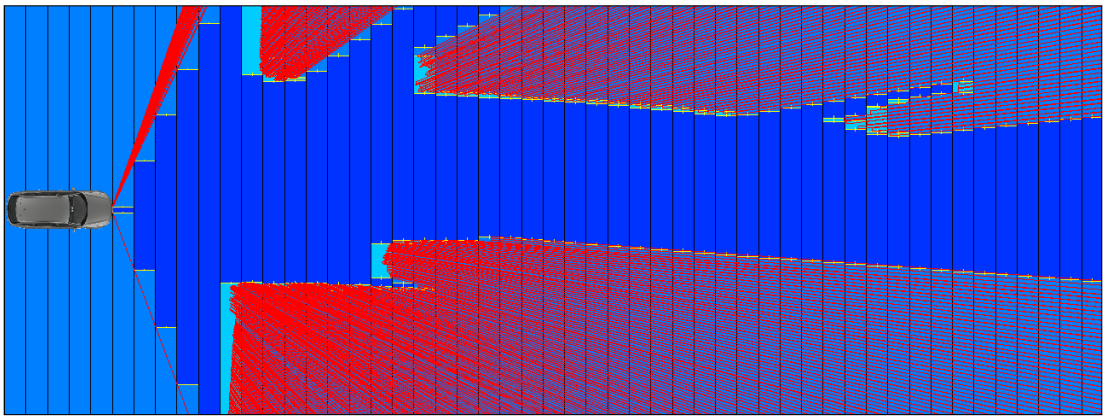


Figure 3.9.: Example of extracted occupancy cells from a laser scan.
For the purpose of improved clarity, only the areas behind the laser reflections are visualized by red trapezoids. Similar to the previous illustrations, the blue colors indicate the occupancy state of the extracted cells.

3.3.3. Update Algorithm

Having extracted cells from raw measurements, the update of existing cell information can be realized by the same procedure for both curved and straight 2DIM. In this algorithm, both introduced state estimation mechanisms have to be considered: The estimation of the cell borders by a 1D EKF and the estimation of the cells' binary occupancy state by the inverse sensor model approach. The entire procedure is summarized in the pseudo code in 3.10 and in figure 3.11. A currently measured cell is used to improve the border position of associable cells and to update the posterior occupancy probability of all overlapping cells. If a measured cell cannot be assigned to an existing one, a new cell has to be established. The association of the cells in the $BorderIsAssociable()$-routine is realized by a comparison of their lateral border positions and bordering occupancy probabilities. Furthermore, it has to be assured that the order of the cell borders cannot be changed by an update.

A major challenge in the recursive estimation of the borders ($UpdateBorderEstimate()$) is the proper treatment of measurement uncertainties. As the cell boundaries result from individual range measurements, whose noise is modeled independently, the following derivation is restricted to the estimation of a single occupancy cell border. Without loss of generality, we further assume that the m-th cell border is determined by the left border of the p-th range measurement, as also visualized in the left part of figure 3.11. In this case, the state and measurement variables are defined as:

$$\boldsymbol{x}_k = \left[y_k^{(m)} \right] \tag{3.16}$$

$$\boldsymbol{z}_k = \left[r_k^{(p)}, \vartheta_k^{(p)}, \varphi_k^{(p)} \right]^T \tag{3.17}$$

```
1:  procedure AssociateAndUpdate(I^(n), z)
2:      C^(Map) ← GetFirstCell(I^(n))
3:      for all C^(Meas) extracted from z do
4:          if BorderIsAssociable(I^(n), C^(Meas)) then
5:              C^(Assoc) ← AssociatedCell
6:              UpdateBorderEstimate(C^(Assoc), C^(Meas))
7:          else
8:              C^(Assoc) ← CreateCell(I^(n), C^(Meas))
9:          end if
10:         while C^(Map) <= C^(Assoc) do
11:             UpdateOccupancyEstimate(C^(Map), C^(Meas))
12:             C^(Map) ← GetNextCell(C^(Map))
13:         end while
14:     end for
15: end procedure
```

Figure 3.10.: Pseudo code of interval update algorithm.

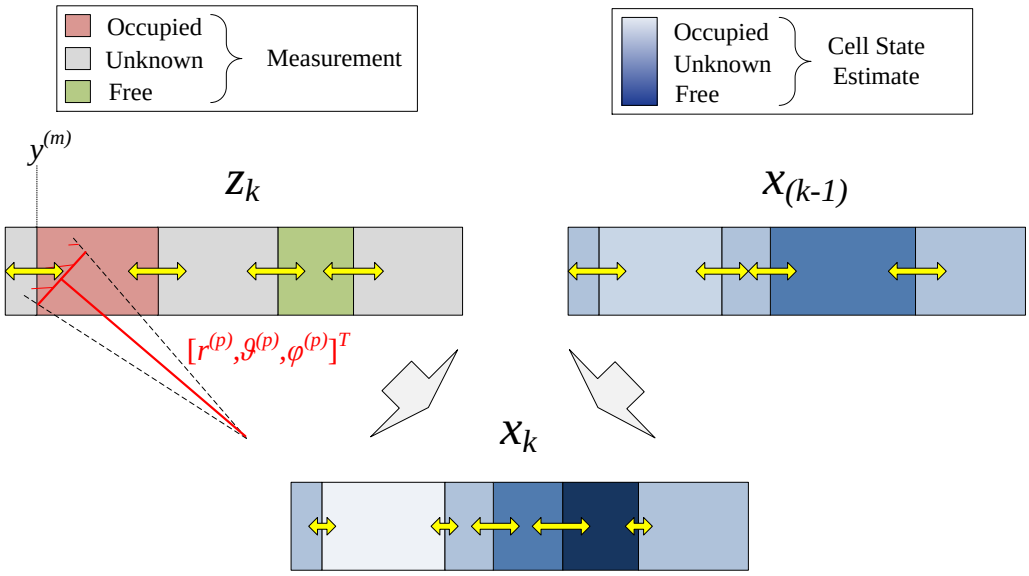


Figure 3.11.: Visualization of interval update algorithm.

Given this relation between state and measurement, it is not possible to formulate a function of the form $\boldsymbol{z} = f(\boldsymbol{x})$, as the position of a lateral border $y^{(m)}$ can result from several range and angle combinations. In such cases, an EKF can be adapted to incorporate a nonlinear *implicit measurement equation* of the form [6, 152, 112]:

$$\boldsymbol{g}\left(\check{\boldsymbol{z}}_k, \boldsymbol{x}_k\right) = \boldsymbol{0} \tag{3.18}$$

A detailed derivation of the EKF using implicit measurement equations is introduced in section A.2.5. Based on the identified state and measurement variables, this equation results as:

$$\boldsymbol{g}\left(\check{\boldsymbol{z}}_k, \boldsymbol{x}_k\right) = y_k^{(m)} - \begin{bmatrix}0\\1\\0\\0\end{bmatrix}^T {}^{I^{(n)}}\mathcal{P}_{Sensor} \begin{bmatrix}\check{r}_k^{(p)} \cos(\check{\varphi}_k^{(p)} + \alpha/2)\\ \check{r}_k^{(p)} \sin(\check{\varphi}_k^{(p)} + \alpha/2)\\ \check{r}_k^{(p)} \sin(\check{\vartheta}_k^{(p)})\\ 1\end{bmatrix} = \boldsymbol{0} \tag{3.19}$$

with ${}^{I^{(n)}}\mathcal{P}_{Sensor}$ describing the homogeneous transformation matrix from the sensor to the interval coordinate system and α denoting the measurement's horizontal field of view, as introduced in figure 3.8. The formulation of the EKF requires the calculation of the partial derivatives of the implicit measurement function g. The partial derivative with respect to the state $\boldsymbol{x}$ simplifies to:

$$\boldsymbol{C}_{\boldsymbol{x}_k} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}(\boldsymbol{z}, \overset{*}{\boldsymbol{x}}_k) = 1 \tag{3.20}$$

The derivative with respect to the measurement $\boldsymbol{z}$ results in:

$$\boldsymbol{C}_{\boldsymbol{z}_k} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{z}}(\boldsymbol{z}_k, \overset{*}{\boldsymbol{x}}_k) \tag{3.21}$$

$$= -\begin{bmatrix}0\\1\\0\\0\end{bmatrix}^T {}^{I^{(n)}}\mathcal{P}_{Sensor} \begin{bmatrix}\cos(\varphi_{(k)}^{(p)} + \alpha/2) & 0 & -r_{(k)}^{(p)} \sin(\varphi_{(k)}^{(p)} + \alpha/2)\\ \sin(\varphi_{(k)}^{(p)} + \alpha/2) & 0 & r_{(k)}^{(p)} \cos(\varphi_{(k)}^{(p)} + \alpha/2)\\ \sin(\vartheta_{(k)}^{(p)}) & r^{(p)} \cos(\vartheta_{(k)}^{(p)}) & 0\\ 0 & 0 & 0\end{bmatrix} \tag{3.22}$$

If we further assume the state to be static, the state transition function $\boldsymbol{f}$ becomes the identity function and the prediction simplifies to an increase of the estimated variance:

$$\overset{*}{y}_k^{(m)} = \hat{y}_{k-1}^{(m)} \tag{3.23}$$

$$\mathrm{Var}(\overset{*}{y}_k^{(m)}) = \mathrm{Var}(\hat{y}_{k-1}^{(m)}) + Q_{k-1} \tag{3.24}$$

The enhancements of this prediction step in the context of ego motion compensation and moving objects mapping will be discussed in the following sections. Due to the previous derivations, the correction step of the EKF results in:

$$\hat{y}_k^{(m)} = \overset{*}{y}_k^{(m)} + K_k \left(-\boldsymbol{g} \left(\left[r_k^{(p)}, \vartheta_k^{(p)}, \varphi_k^{(p)} \right]^T, y_k^{(m)} \right) \right) \tag{3.25}$$

$$\mathrm{Var}\left(\hat{y}_k^{(m)}\right) = (1 - K_k) \cdot \mathrm{Var}\left(\overset{*}{y}_k^{(m)}\right) \tag{3.26}$$

where the Kalman gain is given by:

$$K_k = \frac{\mathrm{Var}\left(\overset{*}{y}_k^{(m)}\right)}{\mathrm{Var}\left(\overset{*}{y}_k^{(m)}\right) + \boldsymbol{C}_{\boldsymbol{z}_k} \boldsymbol{R}_k \boldsymbol{C}_{\boldsymbol{z}_k}^T} \tag{3.27}$$

Hence, the derived equations correspond to a 1D EKF, in which the measurement uncertainties are transformed into the state space by the means of the Jacobian $\boldsymbol{C}_{\boldsymbol{z}_k}$. Detailed results of the transformed uncertainty $\boldsymbol{C}_{\boldsymbol{z}} \boldsymbol{R}_k \boldsymbol{C}_{\boldsymbol{z}}^T$ in combination with varying poses ${}^{I^{(n)}}\mathcal{P}_{Sensor}$ are figured in appendix A.4.1. An example of transformed border uncertainties can be seen in figure 3.12.

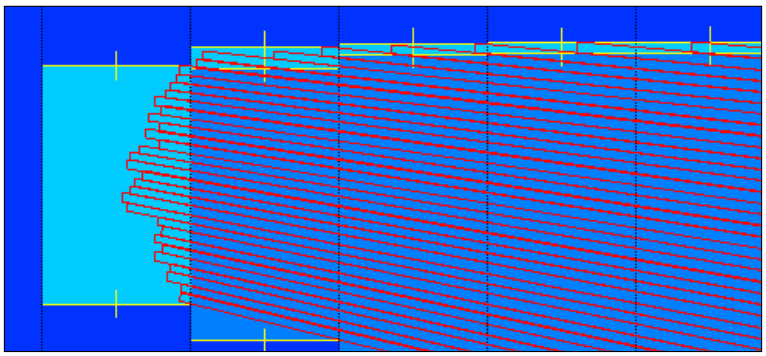


Figure 3.12.: Detail view of extracted occupancy cells and transformed measurement errors.
The red rectangles visualize the areas behind the laser reflections. The border uncertainties are depicted by horizontal yellow bars.

For the update of the cells' binary occupancy state by an inverse sensor model probability $P(o|\boldsymbol{z}_k)$, two different equations are derived in section A.2.6. The standard approach is to convert the probabilities to logarithmic odds ratios, which simplifies the update operation to a single addition. On the downside, the represented posterior occupancy probabilities have to be recovered for the evaluation of the map. This is especially derogatory, as the implementation includes a merging step, which requires a recurring comparison of regular probability values. In this context, it turned out that the effort of repeated conversions

between probabilities and log odds ratios is higher than the benefit of the simplified update equation. Under the assumption of a uniformly distributed a priori occupancy probability ($P(o) = 0.5$), the update in the $UpdateOccupancyEstimate()$-routine is thus realized by the following equation:

$$p\left(o|\boldsymbol{z}_{1:k}\right) = \frac{p\left(o|\boldsymbol{z}_{k}\right) \cdot p\left(o|\boldsymbol{z}_{1:k-1}\right)}{1 - p\left(o|\boldsymbol{z}_{k}\right) - p\left(o|\boldsymbol{z}_{1:k-1}\right) + 2 \cdot p\left(o|\boldsymbol{z}_{k}\right) \cdot p\left(o|\boldsymbol{z}_{1:k-1}\right)} \tag{3.28}$$

The computational effort of the update algorithm strongly depends on the overall number of represented cells. Figure 3.11 already indicates that the quantity of cells can be increased during the update process. On the one hand, the association criteria can be used to control the resulting cell fragmentation. On the other hand, the already introduced merging step takes place, which combines neighboring cells with similar probability values as well. Hereby, it is important to consider the age of the cells for the merging decision, in order to allow the emergence of new cells with conflicting information. The age and probability differences that lead to a combination of neighboring cells can be used to conveniently regulate the memory requirements of the resulting data structure.

3.4. Development of Mechanisms for Spatio-Temporal Consistency

For the application of the newly developed representation in highly automated vehicles, further developments are required. The 2DIM is particularly suitable for extraction purposes if the intervals are aligned orthogonal to the future driving direction. This either requires the repeated alignment of the map to the vehicle's orientation or the attachment of the map to a changing reference path estimation. Furthermore, the assumption of a purely static environment cannot be accepted in typical ADAS scenarios. Under all described circumstances, the spatio-temporal consistency of the representation has to be maintained: In case of a movement of the vehicle or a changing traffic lane hypothesis between two map updates, the previously represented information has to be transferred into the newly aligned data structure in order to enable accumulation mechanisms. Similarly, occupancy information belonging to moving obstacles has to be considered and transformed correctly.

3.4.1. Straight 2DIM Compensation

The ego motion compensation is a mandatory task in each ego local environment representation that tries to incorporate knowledge from previous measurements. Therefore, the knowledge between two transformed representations has to be transferred both accurately and efficiently. A special challenge results from comparatively small vehicle movements between two map updates, which individually have a negligible impact but need to be accumulated over time. For occupancy grid maps, there exist convenient compensation

mechanisms, which avoid interpolations by using two-dimensional circular buffers with subcell values and introducing an additional yaw angle between map and vehicle [182, 16]. To the author's knowledge, only grid-based ego local occupancy maps have been proposed so far. The key challenge in the development of non-grid-based representations lies in the development of suitable ego motion compensation algorithms. In case of the 2DIM, a special requirement results from the fact that an additional yaw angle between vehicle and map has to be avoided, as it would eliminate the main advantages of the interval alignment orthogonal to the driving direction.

Formally defined, the ego motion compensation corresponds to the transformation of the represented features between two transformed map coordinate systems. If we express the vehicle motion by the transformation of the USK between the time indices $k-1$ and k (cf. equation (2.8) in section 2.3.1) and assume a time-dependent pose between map and USK, the ego compensation of a homogeneous map coordinate ${}^{Map}\tilde{\boldsymbol{p}}$ is defined as:

$$ {}^{(Map_k)}\tilde{\boldsymbol{p}} = {}^{(Map_k)}\mathcal{P}_{(USK_k)} \cdot {}^{(USK_k)}\mathcal{P}_{(USK_{k-1})} \cdot {}^{(USK_{k-1})}\mathcal{P}_{(Map_{k-1})} \cdot {}^{(Map_{k-1})}\tilde{\boldsymbol{p}} \tag{3.29} $$

In this equation, a coordinate in the last cycle's map coordinate system is first transformed into the last USK_{k-1}, then into the current USK_k and from there into the current map coordinate system.

If we restrict to the compensation of a point in the interval coordinate system $I^{(n)}$ of a straight 2DIM, the pose between map and USK simplifies to a translation by $x^{(n)}$ (cf. (3.8)). In that case, the coordinates of the point in the new interval coordinate system result as:

$$ \left(I_k^{(n)}\right)\boldsymbol{p} = \underbrace{\begin{bmatrix} -x_k^{(n)} + \Delta x \\ \Delta y \end{bmatrix}}_{\text{Translation}} + \underbrace{\begin{bmatrix} \left(\left(I_{k-1}^{(n)}\right)x + x_{k-1}^{(n)}\right)\cos(\Delta\psi) - \left(\left(I_{k-1}^{(n)}\right)y\right)\sin(\Delta\psi) \\ \left(\left(I_{k-1}^{(n)}\right)x + x_{k-1}^{(n)}\right)\sin(\Delta\psi) + \left(\left(I_{k-1}^{(n)}\right)y\right)\cos(\Delta\psi) \end{bmatrix}}_{\text{Rotation}} \tag{3.30} $$

with Δx, Δy describing the translation of the vehicle on the road surface and $\Delta\psi$ denoting the change of the vehicle's orientation between time steps $k-1$ and k. Hence, the transformation can be subdivided into a rotation and subsequent translation.

Similar to occupancy grid algorithms, the longitudinal movement Δx can be compensated accurately without adapting the map cells. The key idea is to organize the intervals in a circular buffer. Whenever the longitudinal distance exceeds one interval length, the rearmost interval is reinitialized and logically moved to the beginning of the data structure. In case of movements of less than one interval length, the longitudinal shift can be accumulated by adapting the distance between the USK and the intervals $x_k^{(n)}$, similar to sub-pixel values in grid-based compensation algorithms [16]. By contrast, the lateral movement Δy has to be compensated by adapting the borders of all represented occupancy cells.

The rotation of the vehicle impacts both the x- and y-component of represented coordinates (3.30). While the y-component can also be compensated by adapting the border

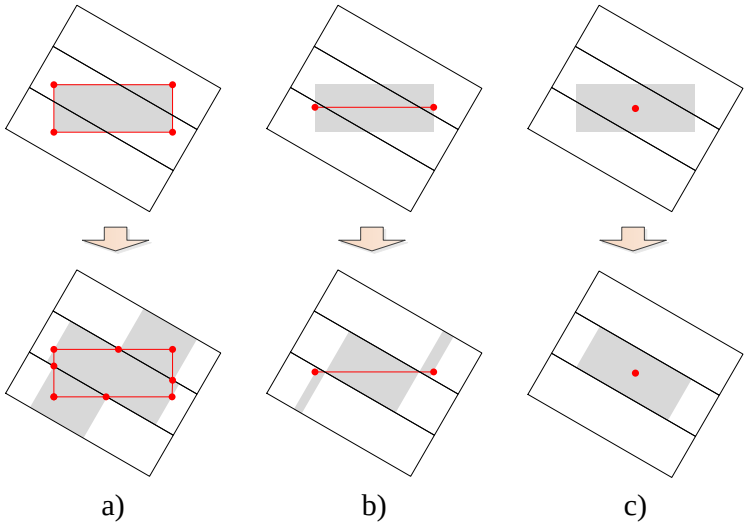


Figure 3.13.: Different strategies for the transfer of cell information across interval borders: a) Modeling of rectangular cell, b) restriction to the cells' lateral extent and c) reduction to a central point.

positions, a representation of longitudinal positions within an interval has not been considered so far. However, a complete disregard of longitudinal positions would result in substantial errors, as some cells could be located in wrong intervals after large rotational changes. To overcome this issue, one can think of several different strategies to transfer cell information between intervals, as illustrated in figure 3.13. The most accurate solution would be to subdivide the rectangular cells into several polygons according to the rotated interval borders (see figure 3.13 a)). On the downside, this strategy would require to maintain complex polygonal lines in order to correctly represent the cells' internal shapes. Due to the increased memory consumption and the necessity of complicated update routines, this approach conflicts with the identified computational requirements. Figure 3.13 b) illustrates a slightly simplified processing, in which only the lateral extent of a cell is considered. Still, this strategy would require to transfer and combine information between neighboring intervals in each system cycle once a center line has reached the border. The third approach (see figure 3.13 c)) only considers the position of a central point to decide when a cell has to be transferred to an adjacent interval. By this approximation, the impacts of vehicle rotations can be modeled without violating the computational requirements. The resulting inaccuracies strongly depend on the rotation magnitude and the lateral extent of the occupancy cells. Nevertheless, it is important to be aware of the impacts resulting from this simplification. Due to the missing modeling of the cells' longitudinal extent, the compensation may lead to gaps in represented obstacles, as figure 3.14 illustrates. The impacts of this approximation will also be evaluated in section 3.5.

The overall ego compensation algorithm is summarized in figure 3.15. At first, the cell's central point is rotated according to the derived equations. The offset between the point

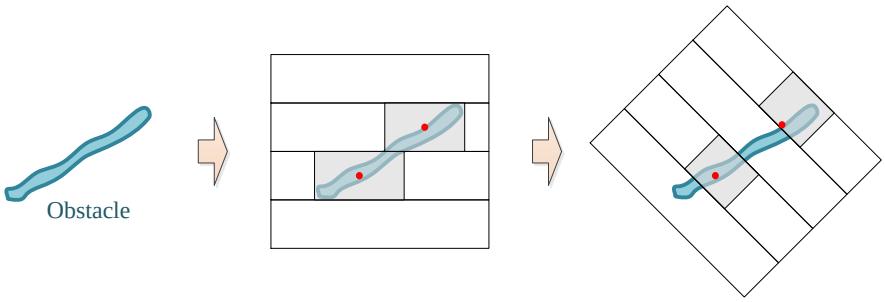


Figure 3.14.: Gaps due to approximation errors in central point compensation strategy.

and the x-axis of the interval coordinate system is evaluated to decide whether a cell has to be transferred. This subcell value will also be used for the rotation compensations in the following cycles. After that, the cell borders are corrected by considering the rotation and lateral movement of the vehicle. Finally, the pose between the map and the USK is adapted in order to compensate the longitudinal motion.

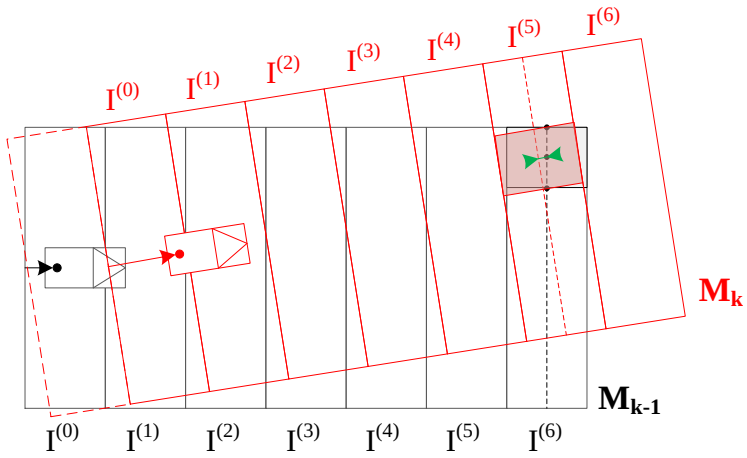


Figure 3.15.: Illustration of the straight 2DIM ego compensation algorithm. The structure of the maps and an occupancy cell at two consecutive time steps is shown in black respectively red. The offset between rotated central point and interval x-axis is colored green.

The introduced compensation approach requires the translation and rotation of estimated border positions. While the translation by a constant does not affect the variance of a random variable, the rotation of the state estimate also requires a modification of the estimated error variances. Given a two-dimensional Gaussian noise term with covariance $\boldsymbol{P}$ and a rotation matrix $\boldsymbol{\Phi}$, the error covariance could be corrected to $\boldsymbol{\Phi P \Phi}^T$. If the provided rotation angle estimation variances are additionally considered, $\boldsymbol{\Phi}$ could be replaced by the corresponding Jacobian. However, the introduced border state and variance

is one-dimensional and can therefore not be adapted exactly. In order to compensate the approximations and missing transformation, the estimated error variance is increased by the process noise Q_{k-1} during the prediction step, as already shown in equation (3.24).

3.4.2. Curved 2DIM Compensation

The previous considerations on occupancy cell transformations can be adapted to the curved 2DIM. In general, the estimation of the underlying skeleton line $\boldsymbol{S}$ in two consecutive time steps is independent from the motion of the vehicle. As the nodes $\boldsymbol{p}$ of the skeleton line are provided in the USK coordinate system, the formal definition is, however, similar to the previous case:

$$\left(I_k^{(n)}\right)\tilde{\boldsymbol{p}} = \left(I_k^{(n)}\right)\mathcal{P}_{(USK_k)} \cdot {}^{(USK_k)}\mathcal{P}_{(USK_{k-1})} \cdot {}^{(USK_{k-1})}\mathcal{P}_{\left(I_{k-1}^{(n)}\right)} \cdot \left(I_{k-1}^{(n)}\right)\tilde{\boldsymbol{p}} \tag{3.31}$$

In contrast to the straight 2DIM compensation, the transformation between the USK and an interval $I^{(n)}$ consists of both rotation and translation, as already defined in equation (3.6).

One major advantage of attaching maps to lane hypotheses is given by the fact that two consecutive estimations of a traffic lane typically only differ in certain parts. In large areas, the existing hypothesis about the course of a lane is usually confirmed or not updated due

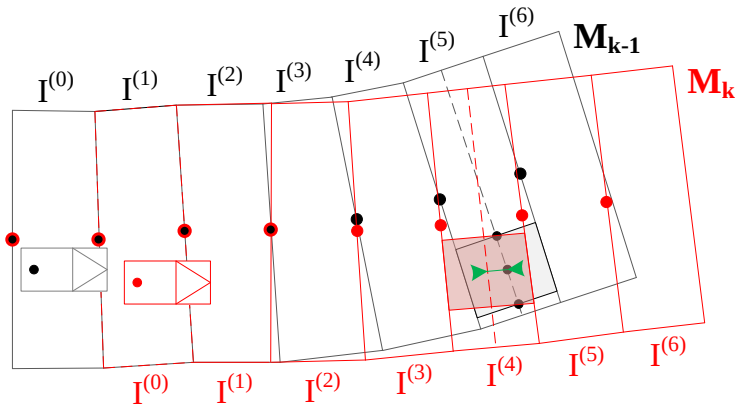


Figure 3.16.: Illustration of the curved 2DIM ego compensation algorithm. The maps respectively occupancy cells at two consecutive time steps are depicted in black and red. Bicolored map nodes indicate unchanged skeleton points. The offset between rotated central point and interval x-axis is colored green.

to missing measurements, e.g. behind one's own vehicle. In these areas, the skeleton line can be defined by globally fixed points, such that:

$$\boldsymbol{p}_k^{(n)} = {}^{(USK_k)}\mathcal{P}_{(USK_{k-1})} \cdot \boldsymbol{p}_{k-1}^{(n)} \tag{3.32}$$

The contents of intervals whose geometry is not changed between two map updates do not even have to be compensated. The remaining contents can be transferred by using similar approximations as in the straight 2DIM compensation algorithm.

The resulting procedure is illustrated in figure 3.16. As the nodes that define the two rearmost intervals have not been modified, the compensation of occupancy cells can be limited to the remaining intervals. Again, the offset value resulting from the rotation of a cell's central point indicates, whether a cell has to be transferred to an adjacent interval. Having identified the destination interval, the border positions are adapted according to the results of equation (3.31). The overall processing cycle remains the same as in the straight 2DIM case.

3.4.3. Consideration of Moving Objects

One major challenge in mapping occupancy states for driver assistance applications is the handling of dynamic objects. As already mentioned, early mapping approaches and especially robotic applications make the assumption of a purely static environment. A violation of this condition leads to characteristic trails of moving objects, making occupancy maps unusable in highly dynamic environments. The development of a profound concept of how to deal with moving objects is a crucial requirement for the application of the newly developed 2DIM for ADAS purposes and another major contribution of this thesis. In lit-

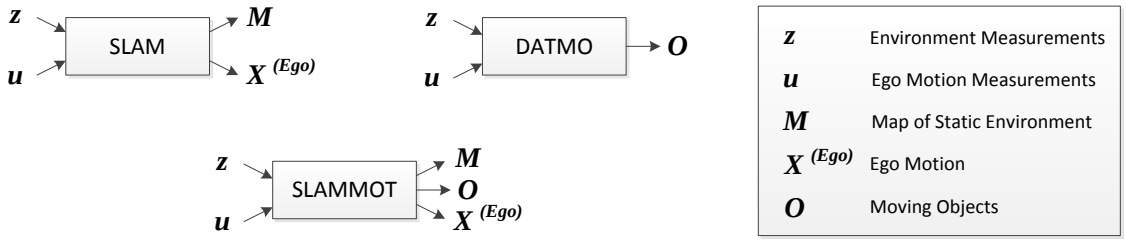


Figure 3.17.: Black box descriptions of SLAM and DATMO, adapted from [178].

erature, the detection and tracking of dynamic objects are often considered together with the SLAM problem, which includes a simultaneous estimation of the vehicle's position. For a comprehensive overview of published SLAM and DATMO approaches (also described as Simultaneous Localization, Mapping and Moving Object Tracking (SLAMMOT)), the interested reader is referred to [118, 178]. A black box description of the abstract problem formulations is illustrated in figure 3.17. Considering the environment perception architecture introduced in figure 1.5, the subtasks of the entire SLAMMOT problem are

distributed to several modules, which exchange information with each other. The development of localization and object tracking algorithms is beyond the scope of this thesis, the corresponding environment perception modules have been described in section 2.3. Instead, the focus lies on the development of a module for *mapping with known poses* [165] and its interaction with a model-based object tracking in order to detect and consistently describe moving objects. Given these conditions, related works on the SLAMMOT problem will be briefly described in the following.

Related Works on Occupancy Mapping in Dynamic Environments

Simplistic mapping approaches for low-dynamic indoor environments create local maps for clusters of possible configurations, e.g. open and closed doors [155]. A more sophisticated mechanism for outdoor applications was presented by Wolf et al. [188]. They maintain two occupancy grids to differentiate static and dynamic parts, new observations can be assigned to the dynamic map by finding inconsistencies between the content of the static map and new observations. However, the overall goal of Wolf et al.'s approach is to create a purely static map for localization tasks in dynamic scenarios.

Substantial contributions to the combined solution of the SLAMMOT problem in urban areas have been made by Wang et al. [177, 176, 178]. A simplified system architecture from

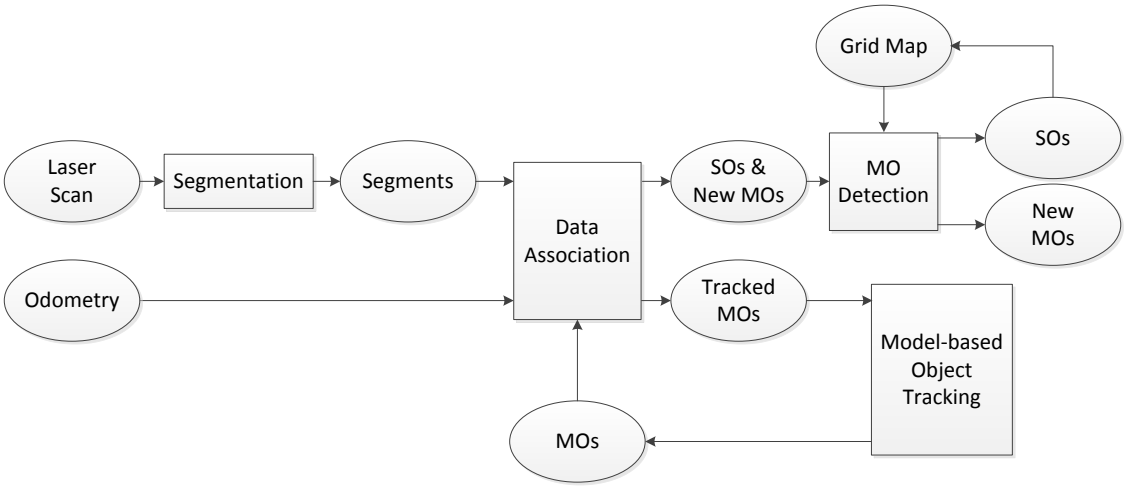


Figure 3.18.: High level architecture of Wang's SLAM with DATMO algorithm, adapted from [178] (SO = Static Object, MO = Moving Object).

their work is depicted in figure 3.18, the localization modules have been left out as they are not in the scope of this thesis. The general idea of Wang's approaches is to represent the static environment by occupancy grids and moving objects by *free-form* objects, a combination of feature- and grid-based description. As illustrated in figure 3.18, the laser data processing starts with a segmentation and the association of the resulting segments to

existing moving objects. From the remaining segments, new moving objects are detected by using a combined consistency- and object map-based detector. All segments that belong to moving objects are used for a MHT[4] and IMM[5] object tracking, the remaining segments are added to a local occupancy grid.

A similar approach is used by Vu et al. in several works on SLAM and DATMO in dynamic outdoor environments [174, 175, 173, 9]. Again, moving objects are represented by bounding box models, whereas an occupancy grid describes the static environment. The detection of moving objects is also based on finding inconsistencies between previously observed occupancy states and new measurements. In contrast to Wang's work, radar [175] and stereo vision data [9] are additionally incorporated for the detection and tracking of moving objects. Overall, the combination of grid- and model-based representations has become a de facto standard for solving the SLAMMOT problem, e.g. also in the 2007 DARPA challenge winning vehicle [49]. Nevertheless, the approaches discussed so far distribute the knowledge about the environment to several representations. They are not able to consistently represent the static and dynamic environment within an occupancy grid.

To overcome this disadvantage, Coue et al. proposed the application of *Bayesian Occupancy Filtering* for multi target tracking [28]. Therefore, they develop a four-dimensional occupancy grid, in which the state of each cell additionally incorporates detected velocities in order to enable a prediction of the map before the update takes place. A similar method was proposed by Richter et al. in [136]. In order to reduce the computational effort, they propose a *hierarchical prediction* of the map, depending on the represented occupancy states.

A related, but more efficient method is presented by Bouzouraa and Hofmann in [18, 16]. This approach uses a combination of model-based object tracking and occupancy grid, in which dedicated grid cells are associated to dynamic objects. In order to consistently describe moving objects in the occupancy grid, the dynamic cells are predicted according to the estimated object motion before the map update. The presented results show that the algorithm is able to process laser and radar measurements for both 2D and 3D mapping applications in real time. Due to the efficient and consistent representation of moving objects, this approach forms the basis of the following 2DIM enhancements.

Interaction of Occupancy Map and Model-based Object Tracking

The main philosophy of the architecture introduced in figure 1.5 is to consider the environment perception system as a combination of experts with different tasks, strengths and weaknesses. All participating subsystems interact closely and support each other in case they require information which is easily provided by any other subsystem. The occupancy map is capable of representing detailed information about free, occupied and

[4]Multiple Hypothesis Tracking
[5]Interacting Multiple Model

unknown areas with unstructured boundaries. The model-based object tracking module utilizes shape and motion models to estimate state vectors of moving objects including alignment, velocity and acceleration. As already described in [16] and [SWBH12], both approaches can benefit from each other and hence improve the quality of the resulting environment representation. On the one hand, the occupancy map can help to classify the dynamic state of raw range measurements in order to reduce erroneous object hypotheses and associations in the tracking module. On the other hand, the estimated state vectors can be used to correctly represent moving objects in the occupancy map. Optionally, the accumulated object shapes in the occupancy map can also be used as feedback for the model-based object tracking. These general principles can be adapted straightforwardly to the interval-based occupancy map. Figure 3.19 shows the resulting interaction between both modules in a message sequence chart. The individual subtasks will be described in the course of this section.

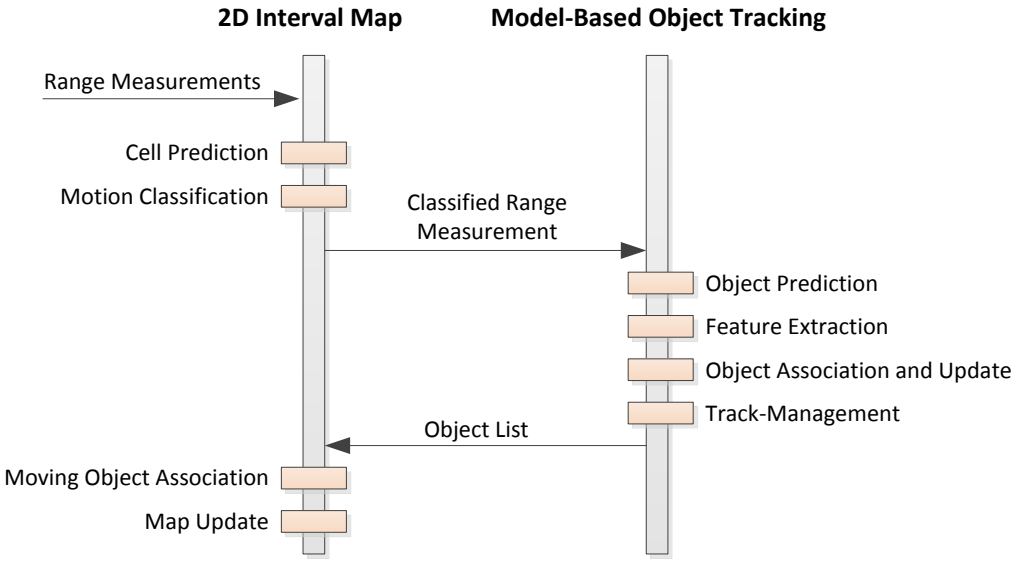


Figure 3.19.: Message sequence chart showing the interaction between 2DIM and model-based object tracking.

In order to allow for a special treatment of moving objects, the previously introduced occupancy cells have to be expanded by a binary dynamic state $d_k \in D = \{static, dynamic\}$ [16]. Formally, the estimation of the time-dependent occupancy probability modifies to:

$$p\left(o_k|\boldsymbol{z}_{1:k}\right) = \sum_{d_k \in D} p\left(o_k|\boldsymbol{z}_{1:k}, d_k\right) \cdot p\left(d_k|\boldsymbol{z}_{1:k}\right) \tag{3.33}$$

In case of a dynamic cell, the posterior occupancy probability depends on the occupancy probability at the previous position and the probability of having moved to the new position. For a detailed derivation of the involved probability distributions, the interested reader is referred to [16].

Prediction

All cells which are associated to the same moving object form a *group* of dynamic cells and have to be predicted according to the estimated state vector and error covariance. As introduced in section 2.3.2, the object tracking module provides the state vector of moving objects including velocities and accelerations in relation to one's own vehicle. However, the prediction of occupancy cells has to be performed by using global velocity and acceleration values relative to the ground. An accurate conversion can be realized in three steps:

- The motion of the detected moving objects has to be transformed from the rotating USK coordinate system into a stationary reference frame, taking into account the angular velocity of one's own vehicle, as derived in appendix A.5.
- By adding up the estimated ego velocity and acceleration, the absolute motion of dynamic obstacles can be inferred.
- In order to finally deduce the motion at a specific map cell, also the estimated yaw rate of the moving object has to be considered. Under consideration of the lever between the required position and the object's reference point, this transformation can also be realized by using the equations described in appendix A.5.

The resulting velocity ${}^{(abs)}\boldsymbol{v}$ and acceleration ${}^{(abs)}\boldsymbol{a}$ can be used to predict the longitudinal and lateral position of a dynamic cell in the map. In x-direction, the previously introduced central point can be moved to determine the containing interval. In y-direction, the cell borders have to be shifted according to:

$$\overset{*}{y}_{k+1} = \boldsymbol{f}\left(\begin{bmatrix}\hat{y}_k & \hat{\boldsymbol{o}}_k\end{bmatrix}^T\right) = \begin{bmatrix}1 & T & T^2/2\end{bmatrix}\begin{bmatrix}\hat{y}_k \\ {}^{(abs)}v_y \\ {}^{(abs)}a_y\end{bmatrix} \tag{3.34}$$

with $\hat{\boldsymbol{o}}_k$ denoting the estimated state vector of the associated object including position, alignment, velocities and accelerations. By linearizing $\boldsymbol{f}$ at the estimated border position and object state, the modification of the border variance can be approximated as follows:

$$\mathrm{Var}(\overset{*}{y}_{k+1}) = \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\hat{\boldsymbol{x}})\right)\mathrm{Cov}(\hat{\boldsymbol{x}})\left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\hat{\boldsymbol{x}})\right)^T \tag{3.35}$$

$$\text{with}\quad \hat{\boldsymbol{x}} = \begin{bmatrix}\hat{y}_k & \hat{\boldsymbol{o}}_k\end{bmatrix}^T \tag{3.36}$$

Assuming independence between the border position and estimated object states, the required covariance matrix results as:

$$\mathrm{Cov}(\hat{\boldsymbol{x}}) = \begin{bmatrix}Var(\hat{y}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \hat{\boldsymbol{P}}_k\end{bmatrix} \tag{3.37}$$

where $\hat{\boldsymbol{P}}_k$ denotes the estimation error covariance of the object tracking module. The resulting calculation rules are listed in appendix A.5.

In order to further consider the uncertainty about a dynamic cell's motion, the predicted occupancy values can be modified by an attenuation mechanism, which decreases the probability value towards the a priori value of 0.5. In the experiments, this method considerably improved the robustness of the cell prediction algorithm. All described calculations are integrated in the prediction step described in figure 3.6. After this processing, the map represents the expected environment at the current time step.

Motion Classification

The consistency-based assignment of dynamic states to raw range measurements plays an essential role in most combined SLAM and DATMO algorithms [188, 177, 176, 178, 174, 175, 173, 9]. Recently, Nuss et al. particularly examined the advantages of using DST occupancy maps for this classification task [114, 115]. In terms of the introduced perception architecture (figure 1.5), this task is performed by the map module as it provides the best information about occupancy states in the environment. The classification result is exchanged with the object tracking module in order to reduce the number of erroneous moving object hypotheses, a formal derivation of the improved object generation can be found in [SWBH12].

This common principle can be adapted to the 2DIM. Taking into account the lateral extent of the laser echos (see figure 3.8 a), the measured line segments can be compared to the predicted 2DIM:

- All measurements that hit a dynamic cell are marked as *dynamic*.
- The remaining measurements are classified according to the map's occupancy values. In case the measurement overlaps several cells with different probabilities, the maximum value is processed.
  - If the resulting probability is below a predefined threshold t_{free}, it is assumed that an obstacle has moved into a previously free space. Hence, the measurement is classified as *dynamic*.
  - If the value exceeds the threshold t_{occupied}, the measurement is categorized as *static*.
  - In all other cases, the measurement is classified as *unknown*.

The accuracy of consistency-based motion classification mainly depends on the sensor characteristics and the motion to be detected. The theoretical and practical limitations of this approach have been examined by Matthaei et al. in [101]. To overcome the well-known shortcomings, the motion classification can additionally incorporate radar measurements, which provide reliable data about the radial velocity of detected obstacles. For details

of this approach, the interested reader is referred to [SWBH12]. Figure 3.20 shows an example of a laser scan classified by using a 2DIM.

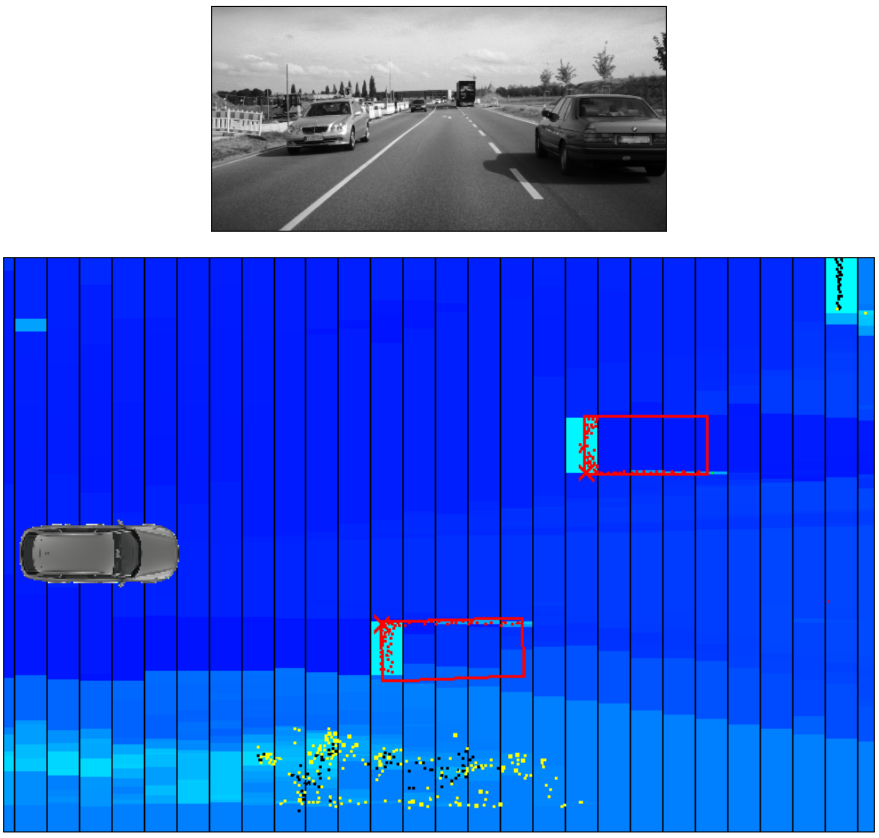


Figure 3.20.: 2DIM-based motion classification of raw range measurement and corresponding video screenshot
Red dots indicate dynamic measurements, black dots static measurements and yellow dots unknown measurements. The red boxes show the corresponding hypotheses from the model-based object tracking module.

Association and Update

Based on the classified range measurements, the object tracking module provides an updated list of moving objects as described in section 2.3.2. By a simple comparison to the specified bounding boxes, range measurements can be associated to moving objects in the 2DIM. If a measurement is assigned to a dynamic obstacle, the object's ID will be stored in the updated occupancy cell in order to enable a prediction in the subsequent system cycle. Cells with differing object IDs will not be merged during the update and merge steps introduced in figure 3.6.

Thus, moving objects can be consistently described by both model-based object tracker and map-based occupancy representation. The additional description of moving objects by dynamic occupancy cells allows to accumulate their shape without using restrictive model assumptions, as also illustrated in figure 3.20. Optionally, the accumulated shape can also be used to improve the model-based estimation of object states [16] or to build local object maps.

3.5. Evaluation

In order to analyze the characteristics of the newly developed representation and to quantify the differences to existing grid-based maps, this thesis further covers the development and application of several occupancy map evaluation approaches. Possible error sources of occupancy representations have been identified by Bouzouraa and Grewe et al. and can also be transferred to the interval-based mapping approach [16, 63]:

- Errors in the data processing: Both grid- and interval-based mapping approaches discretize the vehicle's environment and use approximations, e.g. simplified measurement transformations and independence assumptions.
- Sensor-specific errors: In addition to measurement noise, also inaccurately determined mounting positions, measurement timestamps or latencies can lead to errors in the resulting representation.
- Erroneous ego motion estimation: The quality of the ego motion estimation directly impacts the accuracy of the ego motion compensation. Occurring errors impair the temporal accumulation of occupancy states in the map, as the information from different time steps does not match. According to the simulation results presented in [16], grid mapping algorithms are especially sensitive to this kind of error.

Furthermore, the task of quantifying the quality of a Grid Map's content has also been examined in several publications for robotics and driver assistance applications. In [16], Bouzouraa develops GM quality measures with and without taking into account prerecorded ground truth knowledge. Grewe et al. classify existing evaluation techniques depending on whether they consider the map's suitability for dedicated applications [63]. Similar distinctions can be found in the robotics area, where a map's "usefulness" is often evaluated in addition to map- or cell-based quality metrics [27, 10].

This categorization will also be used in the further course of this section. First, a *map-based* evaluation approach will be developed, which aims at quantifying an application independent quality of interval- and grid-based maps. Then, the impacts of applying the newly developed approach will be tested in a pre-crash ADAS function. As the map-based evaluation approach has to be restricted to a small number of structured environment scenarios, the obtained mapping results are qualitatively compared in real-world scenarios,

too. Finally, the interpretability of 2DIM and GM will be quantified and the evaluation will be summarized.

3.5.1. Objectives

Based on the identified error sources and the general requirements on environment perception for highly automated vehicles (see section 1.2.2), the developed interval-based occupancy map will be compared to a state-of-the-art occupancy grid implementation concerning the following questions:

- Given ideal sensor data, which quality can the environment representation actually achieve? How do the proposed approximations and discretizations influence the quality of the map?
- How does the application of real sensor measurements affect the quality of the map?
- What is the impact of using erroneous ego motion data on the map's quality?
- What are the memory and computational requirements of the representation? How do the approximations and discretizations change the computational effort?
- How does the interpretability of the map influence the computational requirements of the entire system?
- What are the effects of the occupancy map on the behavior of specific ADASs?

3.5.2. Map-based Quality Evaluation

The basic idea of the map-based evaluation approach is to calculate a map's quality by quantifying the similarity to a ground truth representation. That way, the shape and position of all represented occupancy areas can be incorporated without limiting to a specific application. The particular challenge in this evaluation is to consider the characteristic distribution of probability values and the different geometries of the compared representations. The approach will be developed according to the structure illustrated in figure 3.21.

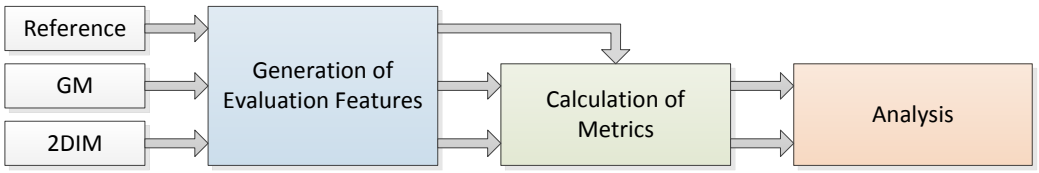


Figure 3.21.: Structure of map-based quality evaluation.

Reference Map

Whereas the ground truth for robotic mapping applications is typically given by architectural blue prints [27], outdoor applications require to precisely measure obstacles in a predefined testbed [16, 63]. The introduced reference sensor system (see section 2.3.3) allows to simultaneously record highly accurate measurements of surrounding moving and static obstacles as well as the corresponding real sensor measurements. In contrast to purely simulation-based evaluations, the provided ideal environment data can also be used to analyze the algorithms' ability to deal with characteristic sensor effects.

The resulting positions and shapes of static and dynamic obstacles are used to infer an ideal *reference map*, which should depict the best possible map describing the current environment. As also the reference sensor system is subject to measurement inaccuracies, the term reference map will be used in the following to emphasize that this map does not represent a perfect theoretical ground truth. In contrast to a direct comparison to the reference object shapes, this approach allows to obtain more reasonable comparison results, as the reference map e.g. considers the sensor characteristics and occlusion conditions. For this purpose, a high resolution occupancy grid is created according to the steps presented in [16]:

- Compensation of the map by using highly accurate ego motion estimation.
- Generation of ideal range measurements based on the provided reference object positions and shapes.
- Update of the map by optimized sensor models, which consider the improved accuracy of the reference measurements.

Generation of Evaluation Features

One of the key issues in comparing the resulting representations to the reference map is to establish comparability between different map formats. The occupancy cells of interval- and grid-based representations have considerably different shapes. Furthermore, the alignment and sizes of two compared representations may differ. In order to quantify the similarity of two maps, the contained occupancy values have to be transformed into comparable *evaluation features*. For this purpose, two different strategies have been developed in the scope of this work.

Figure 3.22 a) shows a *beam-based sampling* approach, which is derived from [16]. For details, the interested reader may also refer to [Sei13]. The key idea of this method is to sample the map's probability values in equally spaced radial distances to the center of the USK. Figure 3.22 b) illustrates the calculation of a separate *evaluation grid*. In each cell of this grid, the average of all overlapped occupancy probabilities is represented. For specific analyses of free and occupied areas, also the minimum and maximum probabilities could be inferred. In case of the GM, these values can be determined by using well-known

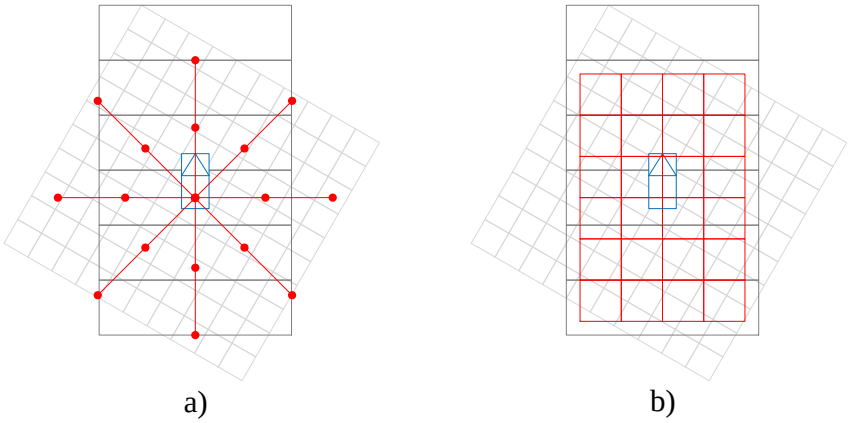


Figure 3.22.: a) Beam-based and b) grid-based evaluation feature generation.

scanline algorithms [20], in case of the 2DIM the required information can be gained by intersecting the evaluation grid cells with the interval cell borders. In contrast to the beam-based sampling, this approach is able to generate dense evaluation features of the compared maps.

Calculation of metrics

By using one of both approaches, a vector of comparable probability values can be calculated from both the reference map and a representation under evaluation. In the further course, they will be denoted as R_i and E_i. In order to assess the deviations of the extracted probabilities with regard to the overall quality of a representation, an appropriate metric has to be applied. Due to the similarity of occupancy grids to greyscale images, the application of image correlation coefficients has repeatedly been proposed for this purpose [10, 11, 27]. However, the presented results indicate that the standard algorithms can hardly be transferred to the task of benchmarking occupancy maps [63]. Besides that, several pixel-to-pixel comparison schemes for occupancy probabilities have been introduced, including:

- The occupied and free cells ratio [63, 26, 80]:

$$M_{occ} = (N_{occ}^{E \wedge R} / N_{occ}^{R}),\ M_{free} = (N_{free}^{E \wedge R} / N_{free}^{R})$$

  with N denoting the number of cells.

- *Spearman's rank correlation* coefficient [16]:

$$M_{\rho} = \frac{\sum_i (rk(R_i) - \overline{rk}(R))(rk(E_i) - \overline{rk}(E))}{\sqrt{\sum_i (rk(R_i) - \overline{rk}(R))^2 \sum_i (rk(E_i) - \overline{rk}(E))^2}}$$

  with $rk()$ denoting the rank of a probability value, for the calculation rule see [91].

- The sum of squared errors (in combination with *normalization maps*, e.g. [27]):

$$M_{SSE} = \sum_i (R_i - E_i)^2$$

- The *map score* [99]:

$$M_{Score} = \sum_i (1 + log_2(R_i E_i + \neg R_i \neg E_i))$$

Although the approach of counting the correctly determined occupied and free cells does consider the meaning of the occupancy probabilities, the exact probability differences do not influence the resulting metric value. Especially in case of the occupied cells, this value furthermore only is restricted to a small fraction of the entire map. The rank correlation coefficient, instead, considers the distribution of the values in both vectors, but also does not quantify the exact distances between the compared probabilities. As a consequence, a scaling error between two compared probability vectors would not be penalized by this coefficient. However, a map with more significant free and occupied probabilities should achieve a higher evaluation metric value.

For a better understanding of the remaining approaches, figure 3.23 illustrates the resulting map scores and squared errors when comparing two probabilities R_i and E_i. The left graph shows that the highest overall map scores are obtained by similar probability values. Additionally, the scores are improved with increasing distance from the probability 0.5. Consequently, this metric rewards both low probability deviations and significant free and occupied map areas. By contrast, the squared error distance in the right part of

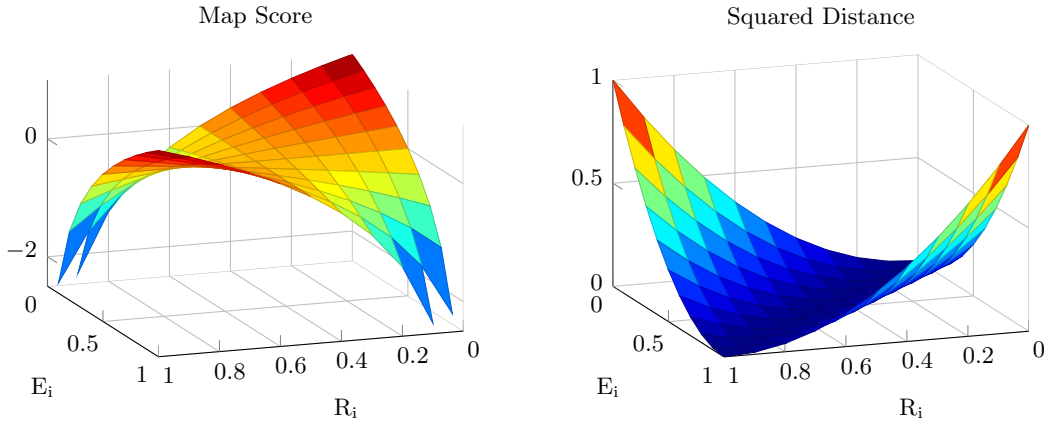


Figure 3.23.: Comparison of possible results of map score and squared error metric.

figure 3.23 does not incorporate the semantics of the compared values. Nevertheless, a stronger emphasis of free and occupied areas can be achieved by subsequently weighting the calculated distances by the deviation of the compared probabilities from 0.5. In the course of this section, this approach will be denoted as *weighted sum of squared errors*.

To be able to compare maps with different sizes, the following experiments will focus on the calculation of the averaged *map score* and the averaged *weighted sum of squared errors*,

which result from dividing the defined metrics by the number of compared cells. Please note that the resulting map score values can only be compared within a single scenario, as the maximum achievable map score depends on the distributions of the probability values in the corresponding reference map.

Experiment 1: Evaluation with Ideal Sensor Data

The goal of the first evaluation experiment is to examine the limitations of interval- and grid-based mapping approaches independent of the effects of real sensor measurements. For that purpose, different representations of occupancy information have been created by using ideal range measurements and compared to the reference map. A detailed description of the corresponding reference scenarios can be found in appendix D.1, the scenarios will be referred to as S1a to S1c.

In order to investigate the effects of the environment discretizations, maps with different cell size respectively interval height configurations have been evaluated by using the grid-based evaluation feature generation and the previously introduced metrics. For a better understanding of the resulting values, figure 3.24 illustrates the development of both metrics for several straight 2DIMs with different interval heights in scenario S1a. During the middle section of this scenario, the vehicle's surroundings only consist of the guardrail at the side of the road, which explains that the metric values indicate identical map qualities independent of the interval height configurations. Towards the end of the recording, the surrounding stationary vehicles reach the mapping area and generate better quality levels for 2DIMs with reduced interval heights.

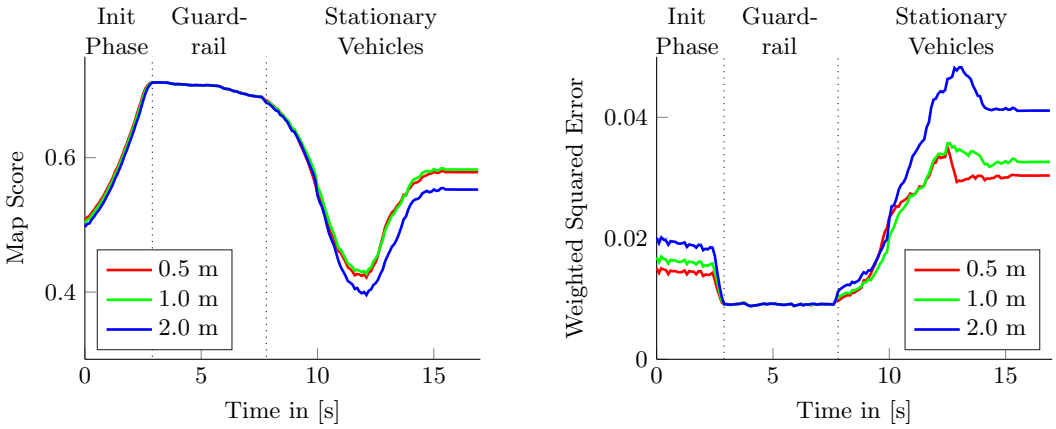


Figure 3.24.: Development of averaged map scores and averaged weighted squared errors for straight 2DIMs with different interval height configurations in scenario S1a.

The obtained interval-based representations have been compared to a GM, which was based on the approach proposed in [16]. Due to the applied ego compensation algorithm

and the required center position of the ego vehicle (see 3.2), the grid maps had to be set to a size of 140 × 140 m to guarantee a preview area of 70 m. Since the developed 2DIM ego compensation mechanism allows a flexible positioning of the ego vehicle and does not change the orientation of the map over time, the map size can be set as the required area of coverage. For the evaluations in this section, a required mapping area providing at least 70 m in front of the vehicle and 20 m behind the vehicle while 30 m width was assumed.

Table 3.2 summarizes the averaged quality metric values, computation times and memory consumptions of grid- and interval-based occupancy representations with different discretization sizes in scenario S1a. Since the provided 2DIM discretization sizes restrict to the longitudinal direction, a comparison of a GM and 2DIM with an equal discretization size of 0.5 m shows quality advantages for the interval-based representation. A further increase of the interval height to 1 m results in almost identical quality metric values, which are still above those of a GM with a cell size of 0.5m. Nevertheless, the best overall quality values can be achieved by applying high-resolution grid-based representations, which on the downside require a substantially increased computational effort. The provided memory sizes in table 3.2 refer to the maximum memory required for the internal representation of the map contents during the processing of the recorded scenario. Whereas a GM's memory demand is predefined by the number and size of the grid cells, the consumption of a 2DIM depends on the number of actually required interval cells. The listed memory sizes illustrate significant benefits from the application of interval-based representations.

Map	Discr. [m]	MS	WSE	CT [ms]	MR [MByte]
GM	0.1	0.61	0	3.1	9.8
	0.2	0.61	0.01	1.2	2.5
	0.5	0.56	0.03	0.4	0.6
Straight 2DIM	0.5	0.60	0.02	0.8	0.1
	1.0	0.60	0.02	0.6	0.05
	2.0	0.58	0.03	0.4	0.03

Table 3.2.: Comparison of ideal occupancy map evaluation metrics with different discretization sizes in scenario S1a.

(Discr. = Discretization size, cell sizes of GM / interval heights of 2DIM, MS = Averaged Map Score - Higher values indicate better map qualities, WSE = Averaged Weighted Squared Errors - Lower values indicate better map qualities, CT = Computation Time [6], MR = Memory Requirement)

[6] All computation times in this thesis were measured on a laptop with Intel Core i7-3840QM CPU with 16 GB RAM.

In further examinations, a GM cell size of 0.2 m and a 2DIM interval height of 1 m turned out as a reasonable compromise between the requirements of highly automated ADAS functions and the computational efforts. The remaining investigations will be based on these map configurations. Table 3.3 summarizes the measured metric values, computation times and memory consumptions for the scenarios S1a, S1b and S1c. In addition to the previous analysis, the table also lists the results of a curved 2DIM, whose reference path was based on a simple estimation of the future trajectory depending on the current ego yaw rate. As the vehicle is moving in a straight line in scenario S1a,

Scenario	Map	Discr. [m]	MS	WSE	CT [ms]	MR [MByte]
S1a	GM	0.2	0.61	0.01	1.2	2.5
	Straight 2DIM	1.0	0.60	0.02	0.6	0.05
	Curved 2DIM	1.0	0.60	0.02	1.6	0.06
S1b	GM	0.2	0.69	0.01	1.2	2.5
	Straight 2DIM	1.0	0.55	0.06	0.6	0.05
	Curved 2DIM	1.0	0.57	0.05	1.7	0.07
S1c	GM	0.2	0.66	0.01	1.4	2.5
	Straight 2DIM	1.0	0.65	0.02	0.6	0.07
	Curved 2DIM	1.0	0.65	0.02	1.6	0.07

Table 3.3.: Comparison of ideal occupancy map evaluation metrics in different scenarios.

straight and curved mapping approaches result in almost identical representations and quality evaluations. However, the complex coordinate transformations of the extensive laser range measurements into the interval coordinate system significantly increase the computational effort of the curved mapping approach. The additional description of the differing interval geometries only slightly increases the memory consumption.

Due to the curved approach of the stationary vehicles in scenario S1b, the ego compensation mechanisms have to deal with larger rotational changes of the ego vehicles than in the other scenarios. Consequently, the measured quality values indicate a larger distance between interval- and grid-based representations, which can be explained by the introduced approximations for the transfer of cell information across interval borders (see section 3.4.1). By the application of the curved 2DIM, the impacts of the approximation can be diminished, as the orientation of the intervals along the estimated future trajectory reduces the compensation effort. Figure 3.25 illustratively compares the resulting ideal straight and curved 2DIM in scenario S1b. The obtained results in scenario S1c are, despite of the included dynamic lane changing maneuver, very similar to those of scenario S1a. Here the measured map qualities of grid- and interval-based maps are almost identical as well, whereas the computation time of the straight 2DIM is significantly lower than those of the GM and curved 2DIM.

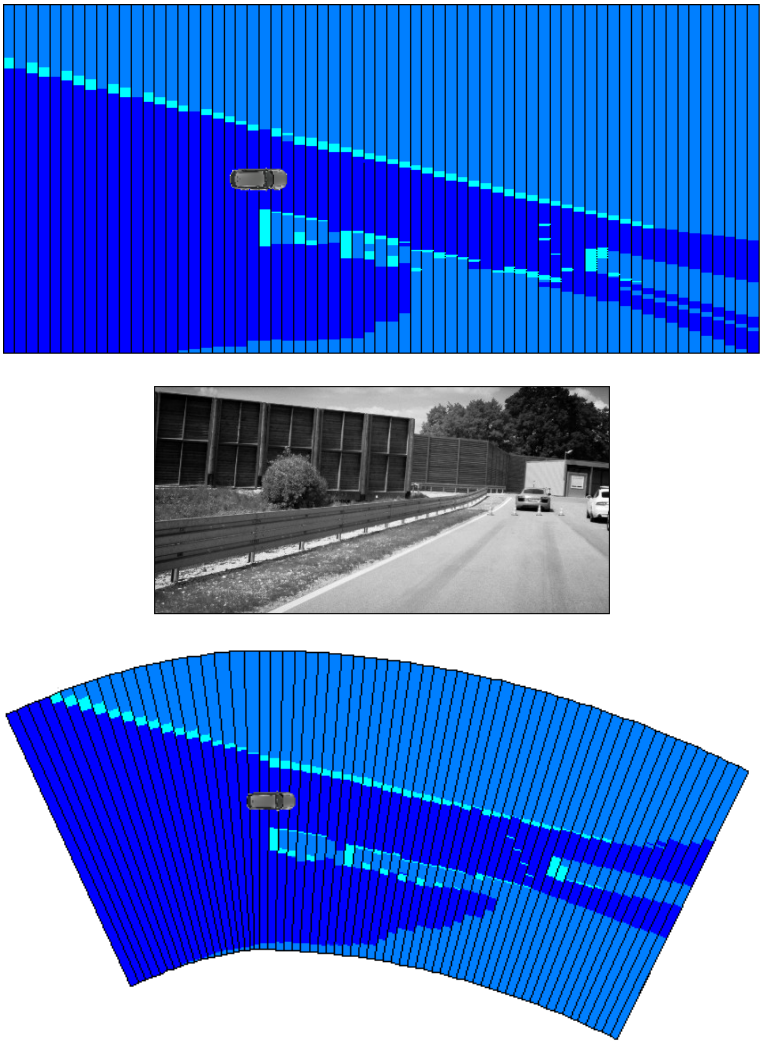
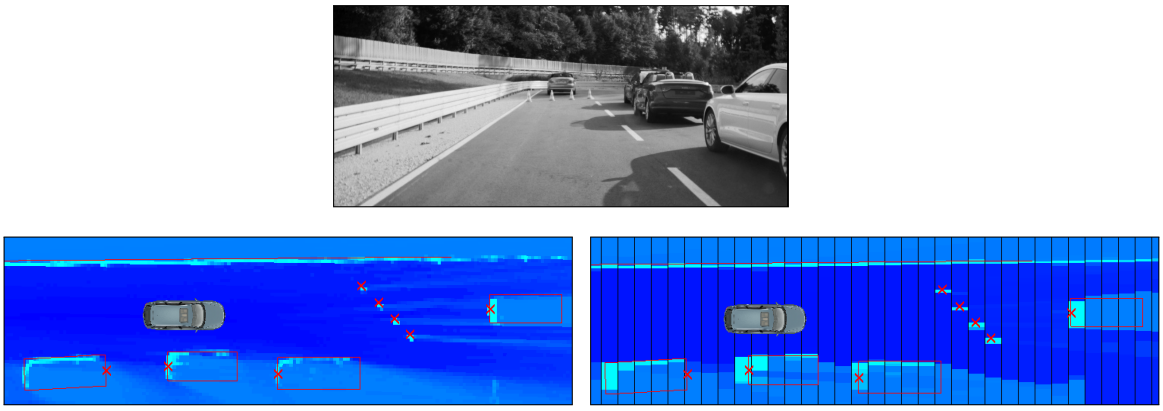


Figure 3.25.: Ideal straight and curved 2DIM during approach of stationary vehicles in scenario S2b.

Experiment 2: Evaluation with Real Sensor Measurements

The second experiment aims to compare grid- and interval-based representations of occupancy information obtained by using real laser range measurements and the previously introduced ego motion estimation (see section 2.3.1). By that, the ability of both approaches to deal with characteristic sensor effects can be quantified. The evaluation setup is identical to the previous experiment: A high resolution grid obtained by ideal range and ego motion measurements will be used as reference map. In order to compare representations with different geometries to the reference map, evaluation grids will be extracted. Their cells are finally compared by using the map score and weighted squared difference metric. The evaluation was performed in the reference scenarios S1a to S1c.

Figure 3.26 gives a first impression of the characteristic differences of the resulting grid- and interval-based representations in scenario S1a. Furthermore, the positions and dimensions of the reference objects are depicted by red lines and boxes. Once more, this comparison illustrates both the advantages and disadvantages of the interval-based mapping approach: One the one hand, the longitudinal distances to the stationary vehicles are, of course,



(a) (b)

Figure 3.26.: Resulting grid- and interval-based occupancy representations in scenario S1a. Red lines and boxes indicate reference objects.

better described in the grid-based representation. On the other hand, the lateral distances represented in the 2DIM are not subject to discretization errors and consequently more clearly accumulated and easier to determine.

Table 3.4 summarizes the averaged performance metrics and resource requirements of the maps when using real laser range measurements and standard ego motion estimation data. Overall, the measured quality values show similar distances between grid- and interval-based representations as in the previous experiment, the application of real sensor data does consequently not result in explicit advantages or disadvantages for one of both approaches. In scenario S1a, the interval-based map achieves an even slightly better per-

Scenario	Map	Discr. [m]	MS	WSE	CT [ms]	MR [MByte]
S1a	GM	0.2	0.38	0.17	1.3	2.5
	Straight 2DIM	1.0	0.39	0.17	1.0	0.12
	Curved 2DIM	1.0	0.39	0.17	2.3	0.14
S1b	GM	0.2	0.39	0.21	1.3	2.5
	Straight 2DIM	1.0	0.29	0.27	0.9	0.11
	Curved 2DIM	1.0	0.31	0.26	2.3	0.14
S1c	GM	0.2	0.39	0.18	1.3	2.5
	Straight 2DIM	1.0	0.38	0.20	1.1	0.13
	Curved 2DIM	1.0	0.38	0.20	2.4	0.14

Table 3.4.: Comparison of occupancy map evaluation metrics in different scenarios.

formance evaluation than the grid-based. The greatest difference between both techniques can still be determined in scenario S1b.

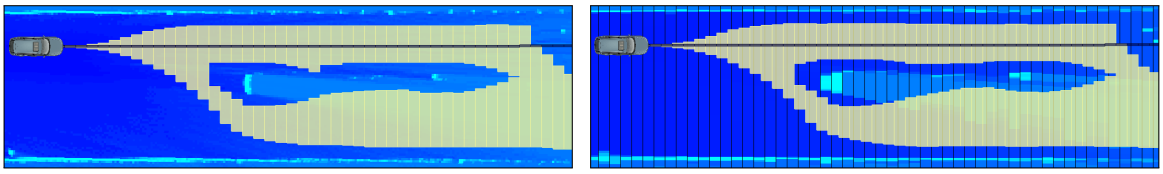
Concerning the computation times, the application of real sensor measurements reduces the gap between interval- and grid-based mapping approach. This can be explained by the fragmented and incomplete detection of the environment by the laser sensor in contrast to ideal range measurements. Due to these characteristics, the number of cells per interval is increased, which raises the processing effort of the developed update mechanisms. Furthermore, also the memory consumption of the 2DIMs is substantially increased. Nevertheless, the measured quantities are still around 100 KByte, which is about 20 times smaller than the static memory requirement of the compared grid map.

The same experiment has also been performed by using the ego motion data obtained by the reference system instead of the introduced ego motion estimation module. However, the modified setup did not significantly change the measured map quality metrics. In all three reference scenarios, the characteristic ego motion estimation errors only had a small influence on the mapping results, especially in comparison to the inaccuracies caused by the laser measurements and the approximations in the developed algorithms.

3.5.3. Application-dependent Quality Evaluation

To further investigate the impacts of applying grid- and interval-based environment representations, the behavior of a collision avoidance system presented in [135] has been analyzed. Based on extracted occupancy states, this function evaluates different evasion trajectories and derives a potential crash distance. The calculated trajectory spaces in scenario S1a are compared in figure 3.27. Potential crash distances can so be estimated, which are also illustrated in figure 3.28. A value of 0 indicates that any crash can be avoided.

Besides small deviations, the illustrated distances and the inferred function trigger are similar in both setups. In all tested recordings of this scenario, the collision avoidance system showed a similar behavior, consequently, there are no disadvantages for this ADAS function when using the interval-based map.



(a) (b)

Figure 3.27.: Space of possible evasion trajectories derived from a) grid- and b) interval-based occupancy representation.

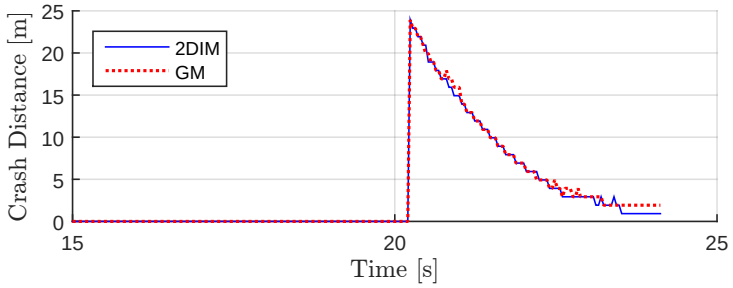


Figure 3.28.: Crash distance estimations derived from grid- and interval-based occupancy representation.

3.5.4. Mapping Results in Real World Scenarios

As already stated before, the previously described experiments cannot provide a complete evaluation of the developed algorithms, as the application of the reference system always has to be restricted to structured scenarios with limited complexities. Therefore, the application of interval-based occupancy maps has also been tested in complex real-world highway and urban scenarios. Although this investigations do not allow to quantify an absolute quality of a map by a comparison to a reference map, the deviations between occupancy GMs and 2DIMs can still be quantified by using the previously introduced extraction mechanisms and metric calculations.

As expected, the largest deviations between grid- and interval-based representations have been observed when taking bends, due to the approximations in the developed ego compensation algorithms. As already observed in the reference scenario S1b, the impacts of these inaccuracies can be reduced by applying the curved 2DIM in combination with a sound prediction of the vehicle's future trajectory. An example, in which the curved

mapping approach improves the obtained results, can be seen in figure 3.29. In this case, the estimation of the future trajectory perfectly matches the actual course of the road, which improves both the map quality and the required computation times. When driving straight ahead, which is especially the case in the extensively investigated dense traffic scenarios on highways, only minor differences between the straight 2DIM and state-of-the-art GMs could be observed.

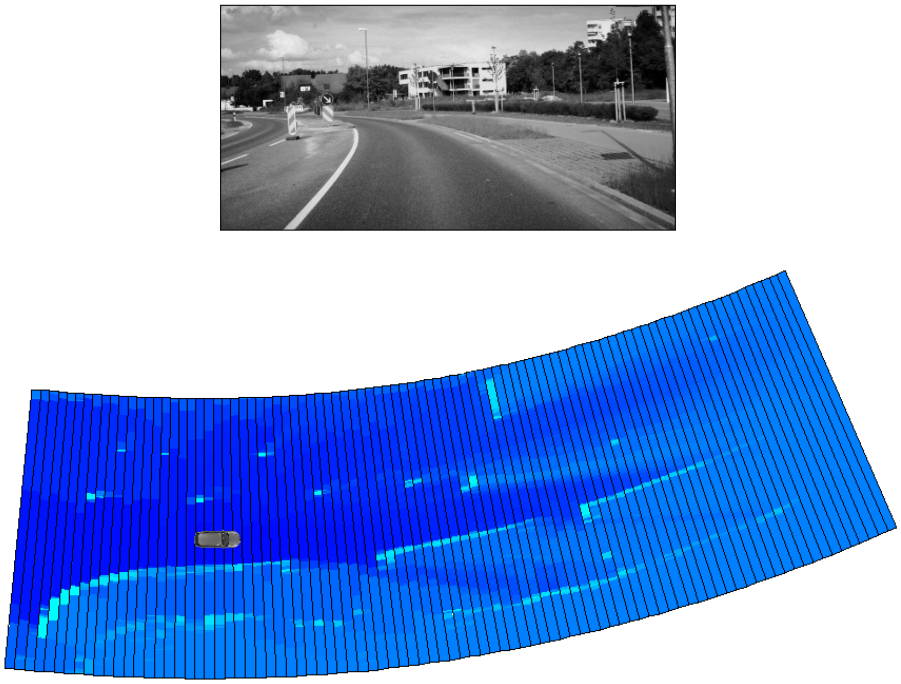


Figure 3.29.: Representation of occupancy information in curved 2DIM.

Due to the mainly unstructured surroundings, the gap between the computation times of interval- and grid-based approach further decreases in complex urban scenarios and highway scenarios. In a five minute long urban reference file, the average computation time of both GM and straight 2DIM was about 1.4 ms, whereas the creation of the curved 2DIM took about 3 ms. The memory consumption of the interval-based maps was about 200 kByte in each tested scenarios, which is still about ten times smaller than the static 2.5 MByte of the compared GM. A key factor in representing the surrounding occupancy states in complex dynamic scenarios is the correct consideration of moving objects. The introduced concept of combining the 2DIM with a model-based object-tracking allows to obtain occupancy representations, in which both static and dynamic objects are consistently represented. By this functional enhancements, the average computation times of both 2DIM and GM are raised by less than 0.1 ms, which is mainly caused by the dynamic classification of the laser range measurements and the prediction of the dynamic cells. All

previously provided memory quantities already include the additional amount of memory that is necessary to assign occupancy cells to moving objects.

3.5.5. Interpretability

Although the measured computation times of GM and straight 2DIM in complex scenarios are approximately at the same level, processing interval-based occupancy representations further offers great advantages. According to the previously introduced architecture (see figure 1.4), the map-based description of occupancy states only serves as an intermediate representation for the extraction of information with higher levels of abstractions during the secondary environment perception, e.g. road boundary polylines [16] or compact descriptions of free spaces [135, 145]. During this process, the previously defined interpretability of the provided data structure has a major impact on the resource requirements of the entire environment representation. Only if the relevant information of the representation can easily be identified, a further step towards an efficient environment perception in highly automated vehicles can be made.

For most extraction purposes during the secondary environment perception, e.g. the previously mentioned road boundary extractor, it is especially advantageous to extract the estimated occupancy probabilities along the past and future trajectory of the ego vehicle. Also collision avoidance functions as [135] benefit from a representation of the surrounding occupancy states in the current ego vehicle coordinate system, as several different possible evasion trajectories can be evaluated with reasonable effort. In case of the 2DIM, this extraction can be realized by a simple iteration over the represented cells within a map interval. Whenever the required occupancy states need to be determined

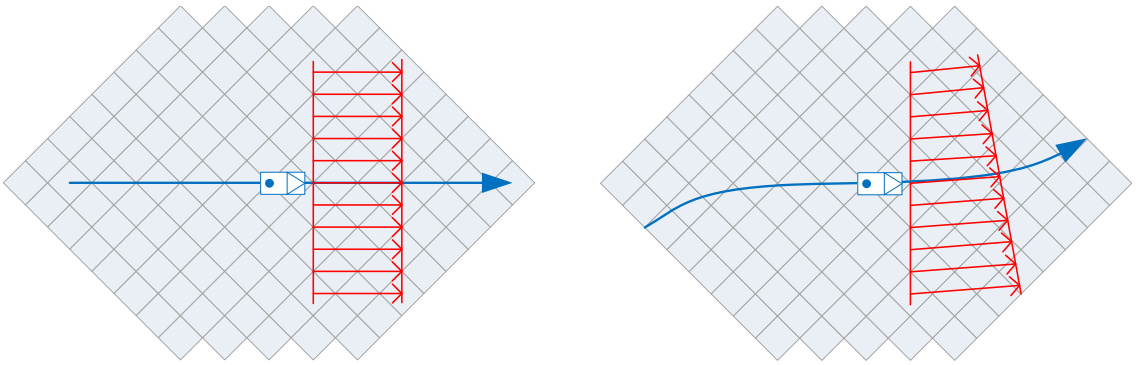


Figure 3.30.: Rasterization of a GM for a) straight and b) curved extraction.

within the current ego coordinate system, the straight 2DIM can be applied, in case of a required extraction along a predicted trajectory, the curved 2DIM can be used. When using a GM instead, the direction-specific extraction of occupancy states requires to rasterize

the data structure by several Bresenham lines, as e.g. described in [135, 16] and also illustrated in figure 3.30. According to [135, 16], a gapless rasterization of all represented occupancy values in the required preview area has to be based on lines along and across the required direction.

In the following analysis, the computation times of an implemented extractor, which determines surrounding obstacles by a simple probability threshold comparison, have been used as a measure to quantify and compare the interpretability of grid- and interval-based representations. According to the previously derived rasterization principles, the occupancy states of GM have been extracted both along a straight line and a predicted trajectory and compared to the extraction from a straight respectively curved 2DIM. In all cases, the extraction covered an area from 20 m behind to 70 m in front of the vehicle with 30 m width. Figure 3.31 illustrates and compares the measured computation times for the mapping and extraction process. Whereas the iteration over the interval-based

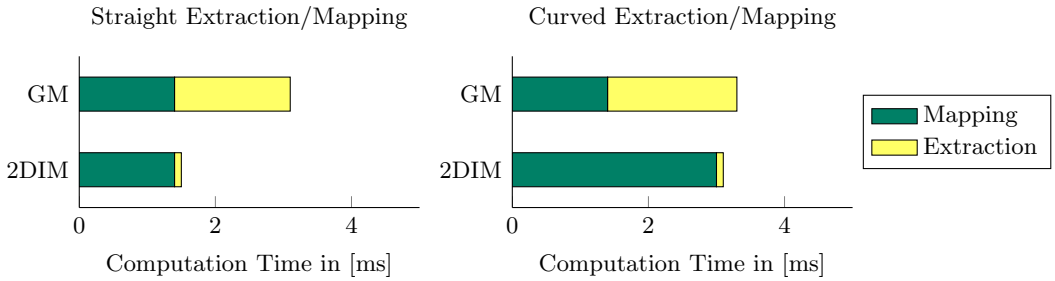


Figure 3.31.: Comparison of mapping and extraction computation times of GM and 2DIM.

occupancy cells took less than 0.1 ms, the rasterization of the obtained grid maps took 1.7 ms respectively 1.9 ms and thus longer than the computation time of the actual mapping process. This leads to a significant performance advantage of the straight 2DIM over the GM when comparing the overall computation times. Also the curved 2DIM benefits from the simplified interpretation of the represented occupancy states. By the application of this map, the main computational effort is transferred from the extraction to the creation of the representation, which finally results in a reduction of the entire computation times.

3.5.6. Summary

Concerning the previously mentioned goals of the occupancy map evaluation, the following conclusions can be made:

- At the same discretization size, ideal interval-based maps provided better quality metric values than comparable grid-based maps. Overall, the representation of continuous values in lateral direction allows to operate 2DIMs at higher discretization sizes than GMs. In the curved reference scenarios, the quality differences between GM and straight 2DIM are increased, which can be explained by the approximations

in the developed ego compensation algorithms. The impacts of these inaccuracies can be diminished by applying the curved mapping approach.

- The application of real sensor measurements did not significantly change the distances between the measured quality metric values of GMs and 2DIMs. Consequently, both approaches are equally able to deal with characteristic sensor effects.
- In comparison to the inaccuracies in the laser range measurements and the data processing, the ego motion estimation errors did only have a minor impact on the mapping results in the tested reference scenarios. Concerning the effects of these errors, no characteristic difference between the interval- and grid-based mapping approach could be observed.
- In the reference scenarios, the computation times of a straight 2DIM were below those of a compared GM, whereas the creation of a curved 2DIM took about 30 % longer. In unstructured urban scenarios the computation times of straight 2DIM and GM were approximately at the same level, the computation times of the curved 2DIM were considerably higher. The memory consumption of all interval-based maps were at least ten times lower than the compared grid-based map.
- A tested collision avoidance function provided equal trigger times for grid- and interval-based maps in all tested scenarios.
- By the improved interpretability of interval-based maps, the complexity of the secondary environment perception can be significantly reduced. If we additionally consider the extraction process, the computation times of a curved 2DIM outperform those of a comparable GM and its extractor. On the downside, the application of a 2DIM requires the commitment to a predicted trajectory. Whenever several substantially different extraction directions have to be considered, e.g. at junctions or in parking scenarios, a more general and direction-independent grid-based representation offers advantages.

4. Representation of Common Object Motion Behavior

One of the main contributions of this thesis is the development of a novel approach to represent the information about collective motion in the vehicle's environment in an early stage of the perception process. By using this information, the overall reliability of highly automated vehicles can be further improved, as the interpretation of other environment features simplifies and additional input parameters for the lateral and longitudinal control can be generated.

4.1. Related Works

As already stated in the introduction, existing environment perception approaches mainly aim to estimate the current state of single moving objects in a traffic scene. A detailed discussion of model-based object representations is not in the scope of this thesis. Besides section 2.3.2, [178, 78, 184, 124] may serve as an introduction to the subject of multi-sensor object detection and tracking. For the derivations in this chapter, the object hypotheses from the previously introduced tracking module (see section 2.3.2) will be used as input data. The interpretation and deduction of relationships between moving objects usually take place on higher levels of abstractions during the secondary environment perception. Apart from a few exceptions, the collective behavior of several traffic participants has not been considered for this purpose.

In military applications, the identification of moving vehicles' common behavior has been examined with different goals. In this case, the focus typically lies on processing point source objects without spatial extension from airborne GMTI [1] radar sensors. In order to reduce the computational load of processing extensive ambiguous data, an important task is to detect and track clusters of moving objects. Waxman et al. give a comprehensive overview over published approaches in this field [179]. In this context, Pollard et al. propose the detection of convoys, which is according to their definition "a group of vehicles traveling together for mutual support and protection" [128, 129]. For that purpose, they analyze the vehicle positions and velocities in a dynamic Bayesian network. Similar works deal with an improvement of convoy tracking by using road-maps [48], the detection of convoy merging sites [139] or an optimized recognition of convoy-splitting scenarios [23].

[1] Ground Moving Target Indicators

As both the requirements and the available input data differ significantly from driver assistance applications, the described concepts can hardly be transferred.

Besides that, there also exist few patent applications which describe the deduction of simplistic collective object behavior for ADAS purposes. Patents [123] and [64] propose an advanced target object selection for ACC systems. Therefore, both approaches model traffic lane hypotheses, which are adapted according to the detected moving objects, for example by analyzing measured lateral velocities. The resulting lane hypothesis can then be used to determine a suitable target object for the longitudinal control in ACC systems. However, these simple approaches are restricted to the adaptation of an initial lane hypothesis by detected objects and do not consider the entire trajectories of dynamic objects.

One possible application of road traffic *convoy tracks* has already been described by Reichel et al. in [134, 133], whereas the deduction of this information has been left open. In the presented approach, the parallelism of detected traffic convoys has been used as an input feature for classifying convoy merging maneuvers. According to the presented results, this input feature considerably improves the classification result, the developed algorithm leads to a classification error of 9.08%.

4.2. Development of an Architecture and Representation of Common Object Motion Behavior

Having identified the scientific gap and possible application scenarios of representing collective motion information for ADAS purposes, the following sections deal with the question of which information can be inferred from surrounding traffic participant behavior and how this information can be adequately represented.

4.2.1. Definition of Represented Information

An example of a traffic situation in which an automated vehicle should adapt to the collective behavior of the surrounding traffic participants has been introduced in figure 1.7, section 1.3.2. Due to an accident, the vehicles begin to form traffic convoys whose courses deviate from the traffic lanes defined by the road markings. In this situation, a human driver adapts to the behavior of the surrounding vehicles. In doing so, he or she preferably does not rely on a single leading vehicle, as it may deviate from the standard behavior, e.g. by performing a lane change maneuver.

In order to utilize this abstract information for ADAS purposes, feasible quantities of collective vehicle behavior have to be identified first. Within the scope of this thesis, the following definitions have been developed:

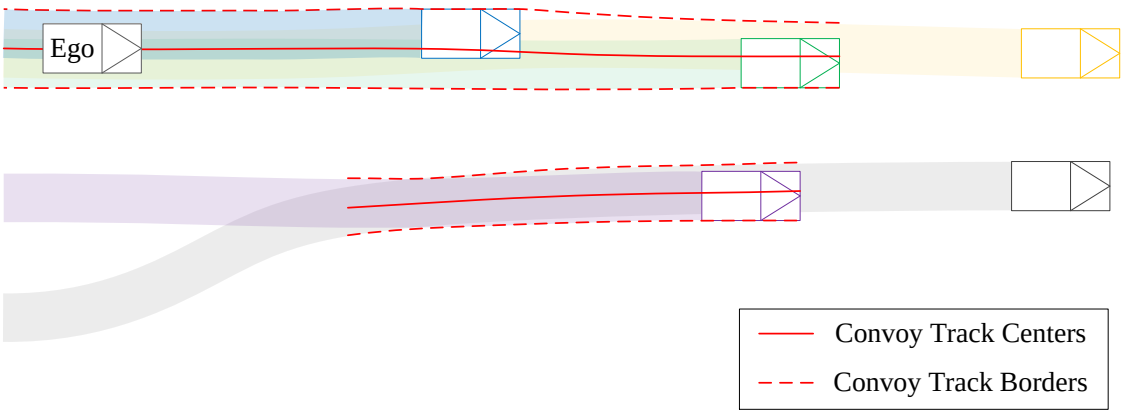


Figure 4.1.: Top view illustration of extracted convoy track centers and borders in a dense traffic highway scenario.

- All collective behavior quantities refer to traffic *convoys*, which have to be established by at least two vehicles. In this way, the influence of single vehicles, which possibly do not conform to the standard behavior, is eliminated. Vehicles are assigned to a common convoy if and only if their trajectories overlap and their motion directions as well as their velocities are similar. These constraints allow to model multiple convoys at the same location, e.g. intersecting convoys at junctions.
- The *convoy track center* represents the center line of the average motion within a convoy. The center line has to incorporate both the covered positions and the driven directions and velocities respectively.
- The *convoy track borders* represent the borders of the complete traversed area of a convoy. Hence, the right border results from the rightmost convoy participant, whereas the left border is defined by the leftmost vehicle.

Figure 4.1 illustrates an exemplary dense traffic scenario with several vehicles and their past trajectories. According to the defined criteria, two convoys can be identified, the corresponding convoy track centers and borders are depicted by red lines. Due to the constraints concerning the motion direction and the minimum number of vehicles, the black depicted trajectory is not incorporated into a convoy track until the lane change maneuver of the corresponding vehicle is finished. Apart from the illustrated case, the developed concepts can also be applied to opposing and cross traffic scenarios. However, the high relative velocities of opposing vehicles and the limited detection ranges of state-of-the-art onboard sensors make it difficult to detect multiple vehicles at coincident positions.

The inferred convoy information can be used for various ADAS applications. First of all, the center line of one's own convoy can serve as an additional and independent input parameter for lateral vehicle control. For that purpose, also the detected borders of one's own and neighboring convoys can help to deduce corridors, within which the vehi-

cle has to remain. Besides spatial information, average velocity profiles can be extracted from detected convoys in order to improve longitudinal control or recommend beneficial traffic lanes. Looking further ahead, convoy information can also be used to realize overtaking assistance systems and safety systems that detect non-standard vehicle behavior [HBW14d, HBW14b, HBW14c, HBW14a].

Concerning the introduced classification into primary and secondary environment perception, the deduction of convoy information can be interpreted as a task of the secondary perception: The represented knowledge about the physical environment feature of moving objects is used to infer an abstract environment feature about their collective motion. However, this deduction does not require a high-level interpretation of the represented environment knowledge, but is restricted to a mutual assignment of similar object trajectories. The resulting information can also help to simplify other tasks within the secondary environment perception, e.g. the interpretation of detected road markings.

4.2.2. Interval-based Representation of Common Motion Information

The key idea of the developed convoy track approach is to map the detected moving object motions in an interval-based representation. In accordance with the introduced architecture 1.5, the represented knowledge can then be used to extract continuous convoy information. A high-level architecture of the entire approach is illustrated in figure 4.2. As input data for the mapping process, model-based object hypotheses derived from radar and laser measurements will be used.

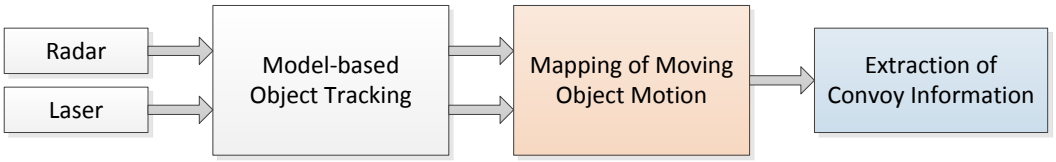


Figure 4.2.: Convoy track detection architecture.

The location-dependent description of detected object motions in a map-based environment representation offers several advantages over pure model-based convoy detection approaches. Although most trajectories of individual vehicles could be represented accurately with generic models, their application would require complex association and fusion algorithms to deduce common motion information. By contrast, the utilization of a map-based representation allows for the accumulation of multiple detected object motions without restricting to models about vehicle trajectories and convoy shapes. Due to the semi-structured character of the required information, the previously introduced straight and curved 2DIM are especially suitable for this application. Therefore, the map needs to be adapted to contain cells which represent detected moving object motion at specific locations in the vehicle's environment. The contribution of the represented *motion cells* to continuous traffic convoys is then derived during the extraction process.

The available input data provide heterogeneous information about the longitudinal and lateral extent of moving objects. The radar-based object hypotheses mostly consist of a single reflection point and do not include reliable information about object contours. Laser-based object measurements on the other hand contain contour information, which may however be incomplete due to occlusions or object locations in the border area of the sensor's field of view. For that reason, all object bounding boxes are provided by using the best knowledge model (see section 2.3.2), which allows to choose a suitable coordinate system for describing object dimensions. Similarly, the heterogeneous information about moving object motion has to be represented uniformly within the 2DIM.

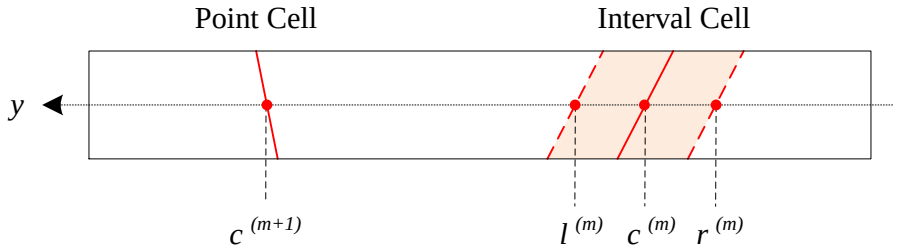


Figure 4.3.: Representation of different motion cells within an interval.

Therefore, a motion cell contains three optional lateral positions which can be estimated according to their observability: the left and right border positions l respectively r and the center position c. Hence, the detected motion can be represented by point or interval cells, depending on the available information. As also illustrated in figure 4.3, all attributes refer to the longitudinal center of the interval, the entire motion cell is formally defined as:

$$\boldsymbol{MC}^{(n,m)} = \left(c^{(n,m)}, l^{(n,m)}, r^{(n,m)}, \phi^{(n,m)}, v^{(n,m)}\right) \tag{4.1}$$

with ϕ denoting the moving direction and v being the velocity magnitude. By incorporating the information of moving directions and velocities, the association between motion cells as well as the convoy extraction process significantly simplify. Furthermore, the additional consideration of cell directions can help to decrease the impacts of the interval discretization. As all input data are provided by bounding boxes, the lateral boundaries of the trajectories can be represented without further approximations. Remaining inaccuracies appear at the longitudinal boundaries of the represented trajectories, which can be regarded as negligible if the trajectories are assumed to be continuable.

In contrast to the occupancy cells derived in the previous chapter, the motion cells do not incorporate a dedicated existence probability of the represented environment feature. In this case, the resulting estimation variances will be used as a measure of reliability. This approach has already been used in several approaches in [16] and turned out to be sufficient for this application.

Similar to the previous application, an interval of a straight or curved 2DIM contains a variable number of motion cells:

$$\boldsymbol{I}^{(n)} = \left(\boldsymbol{MC}^{(n,0)}, \cdots, \boldsymbol{MC}^{(n,m)}, \cdots, \boldsymbol{MC}^{(n,M)}\right) \tag{4.2}$$

However, the cells within an interval are not required to be organized in a dense list in this case. This allows to model gaps and overlapping areas between neighboring motion cells, e.g. to represent motions with different directions at the same position. Concerning the identified characteristics of map- and model-based representations (see section 1.3.1), the interval data structure can, strictly speaking, not be classified as a map-based representation, as the described space elements are not dense and lack in providing a neighborhood relation in lateral direction. On the other hand, the description of the motion cells fulfills the requirements of a low level of abstraction and especially does not make use of restrictive model assumptions.

4.3. Development of Sensor Models and Update Algorithms

Within the scope of this thesis, a scalable concept for detecting convoy track information from laser and radar sensors has been developed. For details concerning the physical measurement principles of the available sensors, the interested reader is referred to appendices B.2 and B.3. The applied double Third Generation Long Range Radar (LRR3) system is able to measure visually occluded moving objects, which is an essential prerequisite for a reliable detection of collective vehicle motion. On the other hand, the laser sensor provides detailed information about object contours and hence spatial extensions of trajectories and resulting traffic convoys. The following section begins with an analysis of the provided measurements of both sensors. After that, the sensor fusion concept, the sensor models and the update algorithms will be derived.

4.3.1. Sensor Data Analysis

An analysis of the provided measurements for the detection of convoy tracks requires the generation of reference objects in characteristic dense traffic scenarios. For that purpose, two different strategies have been applied in the experiments for this thesis, both are illustrated in figure 4.4. The upper part of the figure shows the application of the introduced reference system (see section 2.3.3). All target vehicles transmit their global position to the ego vehicle, where the bounding boxes of the preceding vehicles can be inferred under consideration of the a priori knowledge about the vehicle dimensions. The lower part of the figure shows a novel strategy. In this case, each vehicle uses a laser sensor to measure the preceding vehicle. After exchanging the relative locations of preceding objects, the contours of visually occluded objects can be composed in the ego vehicle. On the one hand, this approach allows for generating reference objects without using expensive IMU and DGPS hardware, on the other hand, the obtained bounding boxes are, of course,

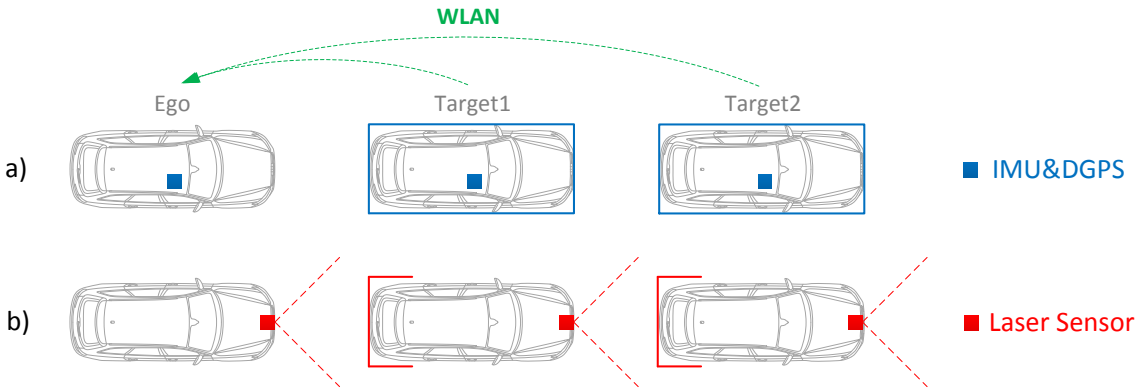


Figure 4.4.: Reference object generation for measurement analysis.

significantly less accurate. Due to the large levers and the propagation of the estimation errors, especially the deduced orientation of laser-based object hypotheses turned out to be imprecise. In the following examinations, a combination of both strategies will be used: Up to three vehicles are equipped with IMU and DGPS hardware, additional vehicles are incorporated by available environment sensors, in order to increase the overall number of reference objects.

An exemplary comparison of the obtained reference objects and the laser-based object hypothesis from the ego vehicle is shown in figure 4.5. The dimensions of the provided bounding boxes strongly depend on the visibility conditions. Due to the output of solely predicted hypotheses and the rule-based innovation of length and width (see [SWBH12]), the objects can temporarily extend into occluded areas. The incorporation of the filtered state variables for the estimation of motion cells will be discussed in detail in section 4.3.3. Due to the optical measuring principle, completely visually occluded objects cannot be detected by the laser sensor.

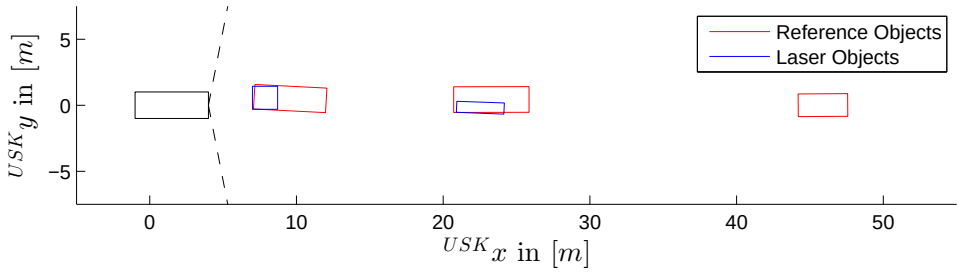
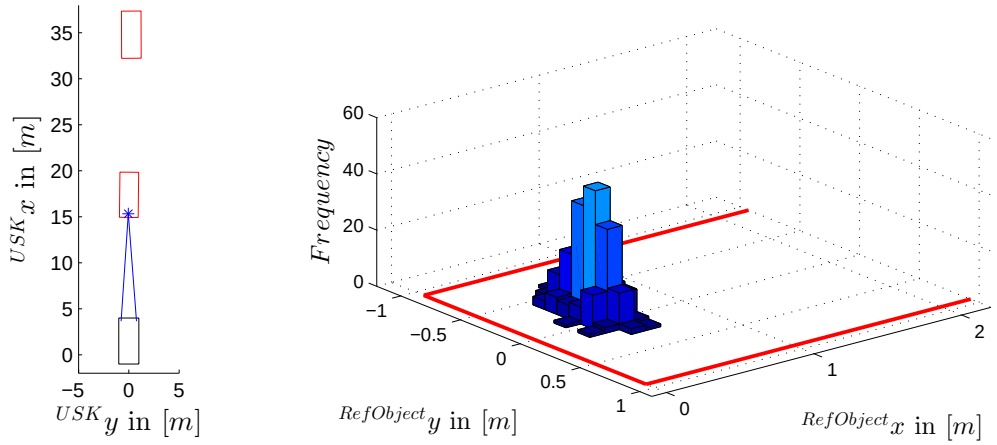
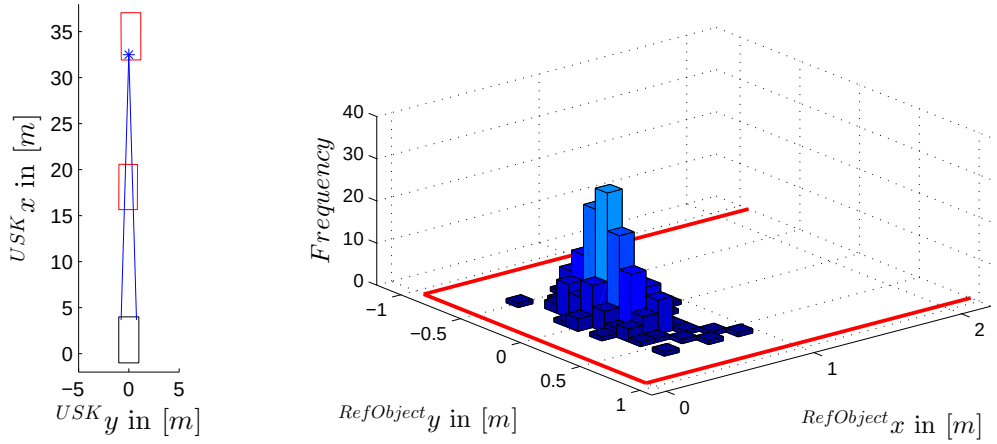


Figure 4.5.: Comparison of reference objects and laser-based object hypotheses.

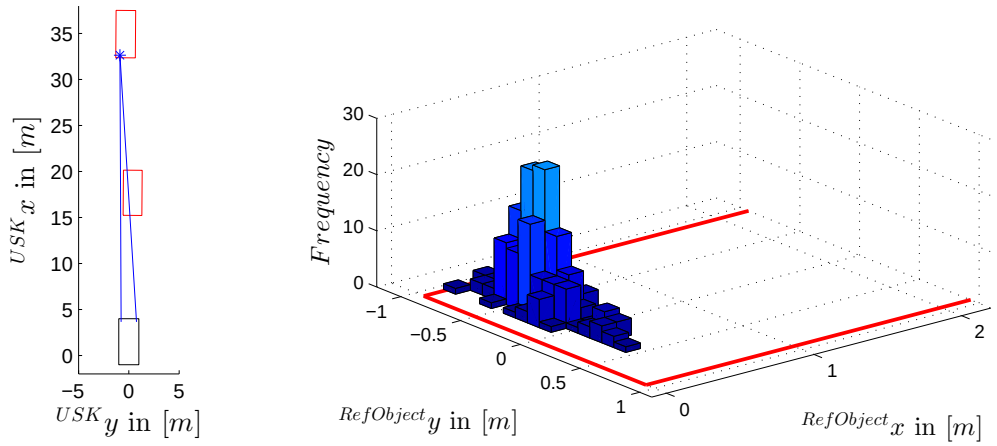
By contrast, the applied double LRR3 system can utilize multipath propagations of the electromagnetic waves to detect several occluded moving targets. On the downside, the sensor is restricted to providing a tracked reflection point instead of accurate bounding



(a) Unoccluded vehicle



(b) Completely occluded vehicle



(c) Partially occluded vehicle with direct line of sight

Figure 4.6.: Comparison of reference objects and radar-based object measurements.

boxes. The sensor's detection range and separation efficiency, as well as the distribution of the provided reflection point have been examined in [16, 104]. Within this thesis, a detailed analysis of radar measurements in dense traffic scenarios with occluded objects has been conducted. An excerpt of the results is shown in figure 4.5, additional graphs are provided in appendix D.3.2. As already indicated in [16], the range of potential reflection point locations is especially large when objects with lateral offsets are detected in close proximity to the sensor, e.g. vehicles in neighboring lanes. Depending on the visibility conditions, the reflection point tends to be located at the corners and wheel arches of the vehicle [104], which considerably complicates the interpretation of the measurement.

In case of a free line of sight to a vehicle that moves in radial direction, the location of the provided reflection point is normally distributed along the vehicle's rear [16]. Figure 4.6 a) illustrates the corresponding test setup and a histogram of the recorded measurements within a 60 second sequence. The peak of the reflection points is located at the lateral center of the reference object with a small longitudinal offset to the border of the bounding box. Regarding the lateral distribution, the hypothesis of a normal distribution with mean 0 can be accepted by a chi-squared test for a significance level of 5%. Similar measurement characteristics result from completely occluded vehicles, as the test results in figure 4.6 b) indicate. Apart from an increased variance, the obtained reflection points are similarly distributed, a chi-squared test again indicates a normal distribution of the lateral offsets. As soon as one of the two radar sensors is able to establish a direct line of sight to parts of the occluded vehicle, the distribution of the reflection points significantly changes. In that case, the number of measurements in the visible part of the object increases, as the example of a partially occluded vehicle in 4.6 c) illustrates. A systematic distribution of the reflection points under these conditions could not be found during the experiments, especially as the provided object hypotheses are subject to tracking delays and constant visibility conditions are hardly reproducible.

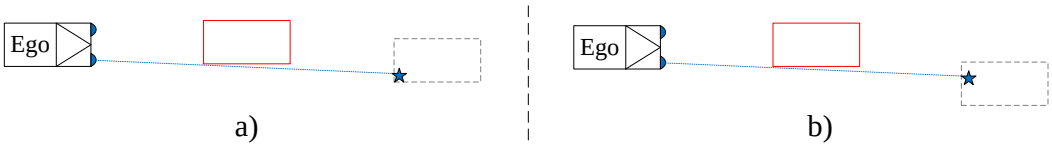


Figure 4.7.: Ambiguity in the interpretation of radar reflection points.

The described effects complicate the interpretation of radar reflection points in typical traffic convoy scenarios. A key problem is illustrated in figure 4.7: Even if the contour of the leading vehicle can be reconstructed from laser measurements, it cannot be derived unambiguously whether a radar measurement rather refers to the corner or center of another vehicle. Due to the radar sensors' mounting positions and the lateral offsets of vehicles in real world traffic convoys, these partial occlusion scenarios occur more frequently than complete occlusions of preceding vehicles. In order to improve the mapping results in these situations, the implemented sensor model additionally increases the lateral variance of the created motion cell whenever a potential partial occlusion is detected.

4.3.2. Embedding in Perception Architecture

For the mapping process, the measurements of both sensors can either be processed to model-based object representations in a central object fusion module or in separate tracking modules, as illustrated in figure 4.8. Regardless of the chosen approach, all object lists are provided by using the best knowledge model interface, which was introduced in section 2.3.2.

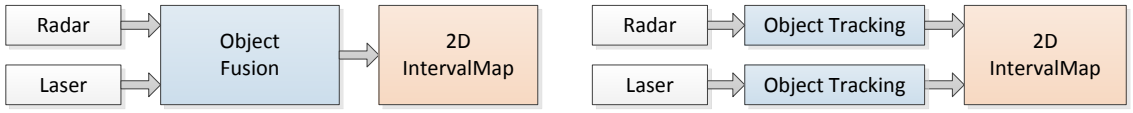


Figure 4.8.: Comparison of possible object list inputs for convoy track detection.

From the 2DIM's point of view, the advantages of processing previously fused object tracks are:

- The measurement interpretation simplifies, as measurements of identical objects from different sensors are already fused.
- The object lists can be processed without further synchronization effort, which otherwise results from out-of-sequence measurements due to the different latencies and cycle times of the laser and radar sensors.
- The data transmission effort is decreased.

On the other hand, the application of several sensor specific object lists also provides benefits:

- The object hypotheses can be unambiguously assigned to the sensors. Hence, sensor-specific effects can be considered in particular.
- There is no preselection of measured objects, all obtained object hypotheses are available for the detection of convoy tracks.
- Additional latencies due to repeated filter applications can be avoided.

Within the scope of this thesis, sensor models for both strategies have been developed. In the sensor model for fused objects, a set of rules was used to assign the provided hypotheses to the measuring sensor. Due to ambiguities in that decision and internal object fusion parameters that could not be influenced, the processing of sensor-specific object lists provided slightly better results. For that reason, the developments in the following sections will restrict to the second approach.

4.3.3. Sensor Model

As the described model-based object representations are already inferred under the assumption of normally distributed measurement and estimation errors, the key idea of the developed sensor models is to reuse the provided object lists for an estimation of the motion cells (4.1) by EKFs. Therefore, the developed algorithms have to consider the identified characteristics of the provided input data and the general requirements for the application of a KF.

Decorrelation of Tracked Objects

Both radar sensor and laser-based object tracking provide independent lists of tracked objects and estimation covariances. When re-using these results as input data for another filter, a possible cross-correlation of the estimated state vectors and error covariances (in this context often together referred to as *tracks* [40]) from different modules as well as different timestamps has to be considered. In particular, the correlation of the provided estimation covariances violates the KF's requirement on temporally uncorrelated white measurement noise [112, 40]. This general problem has been examined extensively for the task of fusing several sensor-level tracks to a global-level track, which is commonly known as *track-to-track fusion* [1]. Drummond et al. identified three possible sources for cross-correlated error covariances in sensor-level tracks [40], from which two can be transferred to the task of estimating 2DIM motion cells in the described scenario:

- If both input sources use similar state transition models, the process noises of the provided local tracks are cross-correlated. Since the process noise affects the estimation results of the local trackers, their estimation errors are also correlated.

- All provided local tracks incorporate the knowledge about previous measurements of the target. If these results are used to repeatedly update the motion cells in the 2DIM, local and global tracks share a common measurement error history and are hence cross-correlated. This effect can be compared to a repeated update of the motion cells by the same measurements, which the KF always considers as new information.

For track-to-track fusion applications, numerous approaches to dealing with cross-correlated tracks have been published, an overview and a categorization can be found in [38]. Besides the basic approach to avoiding correlations by periodically restarting global-level tracks, existing methods can be classified into two major categories [38, 112]:

- The first category of approaches fuses the correlated local tracks at the global level under consideration of the track-to-track cross-correlation resulting from both previously introduced sources. According to Seeliger et al., this category mainly comprises three methods [146]:

  - The recursive calculation of the cross-correlation matrix, which significantly increases the processing and communications effort in track-to-track fusion systems [22].

  - The covariance intersection method, which either requires a convex optimization or the determination of an approximate solution [146].

  - The *Information Matrix Fusion (IMF)* algorithm [24], which is sometimes applied without taking into account the cross correlation between two local tracks [1] in order to simplify the calculations at the expense of neglecting correlations due to common motion models.

- The second approach is to remove the cross-correlation between sensor-level and global-level tracks by exchanging locally decorrelated information between the trackers (sometimes denoted *tracklets* [40, 112]). The most popular representatives of this category are the *inverse information filter* and *Frenkel's methods* [52, 53]. On the one hand, these methods require less adaptations and cause only little computational extra effort, on the other hand, they only remove correlations due to the common measurement history. Enhancements concerning the second matter have been introduced in [39].

Due to the significantly reduced complexity and published results that suggest that the impact of common process noise can be neglected in most object tracking applications [112], the decorrelation of the tracked object lists has been realized by calculating locally decorrelated tracklets. The application of Frenkel's methods especially allows for the integration of the track decorrelation into existing KF or EKF implementations, at which the decorrelation can be easily switched on and off for evaluation purposes. In the following, the main principles of Frenkel's methods will be briefly summarized, for an in-depth discussion of this decorrelation approach, the interested reader is referred to [37, 22].

The key idea of Frenkel's methods is to use two state estimates of a local tracker in order to infer an *equivalent measurement* $\boldsymbol{u}$ and a measurement covariance $\boldsymbol{U}$ which represent the gain of information between both estimates. As the calculated covariances are not correlated, the obtained results can be used to update a global-level track, in this case the motion cells of the 2DIM. For this purpose, Frenkel proposed two different approaches. The first method, which is also known as *inverse KF* [52, 53], allows for the calculation of equivalent measurements in the existing state space to avoid further transformations and will therefore be used in the following derivation. The developed decorrelation implementation is based on a particularly efficient reformulation of this method by Drummond [39].

Since the radar- and laser-based object lists are transmitted to the 2DIM after each sensor measurement, the derivation of the equations is restricted to the consideration of two consecutive local tracker time steps j and k. In this case, the equivalent measurement $\boldsymbol{u}_k$

and measurement covariance $\boldsymbol{U}_k$ can be calculated from the provided estimates $\hat{\boldsymbol{x}}_k$ and $\hat{\boldsymbol{P}}_k$ as follows:

$$\boldsymbol{u}_k = \overset{*}{\boldsymbol{x}}_k + \boldsymbol{A}_k \left(\hat{\boldsymbol{x}}_k - \overset{*}{\boldsymbol{x}}_k \right) \tag{4.3}$$

$$\boldsymbol{U}_k = (\boldsymbol{A}_k - \boldsymbol{I}) \overset{*}{\boldsymbol{P}}_k \tag{4.4}$$

$$\text{with } \boldsymbol{A}_k = \overset{*}{\boldsymbol{P}}_k \left(\overset{*}{\boldsymbol{P}}_k - \hat{\boldsymbol{P}}_k \right)^{-1} \tag{4.5}$$

Hence, a decorrelation of the provided object lists can only be realized if either the predicted object states $\overset{*}{\boldsymbol{x}}_k$ and estimation covariances $\overset{*}{\boldsymbol{P}}_k$ are provided, or these quantities can be inferred from the last cycle's estimates $\hat{\boldsymbol{x}}_j$ and $\hat{\boldsymbol{P}}_j$ by a familiar state transition model. Concerning the applied radar sensor, neither predicted object states nor information about the internally used state transition model were available for the developments in this thesis. As the assumption of an arbitrary motion model does not seem appropriate, the decorrelation of the object tracks has only been realized for the laser-based hypotheses obtained from the environment perception tracking module introduced in section 2.3.2.

In practice, the motion cell representations obtained with activated and deactivated object track decorrelation showed only minor differences. This behavior can be explained by the low inertia of the applied object tracking module as well as the limited duration the motion cells remain within the 2DIM. Furthermore, no feedback between the global and local trackers takes place in this application, which is one of the major challenges in the decorrelation of track-to-track fusion architectures [112]. In the implementation, additional challenges resulted from rule-based adaptions of several object state variables within the applied tracking module. Whenever a provided state variable or related error covariance term does not result from a KF-based innovation, a feasible equivalent measurement cannot be inferred by applying equations (4.3) - (4.5). As a consequence, the motion cell estimation has to rely on correlated input data in these cases.

Update Algorithm

Due to the assumption of normally distributed measurement and estimation errors, the available object tracks respectively deduced equivalent measurements are provided as multivariate Gaussian distributions and hence satisfy the input requirements of an EKF-based estimation of motion cells. Besides the preceding decorrelation of the input data, the overall estimation cycle conforms to the update of the 2DIM with occupancy information, which was introduced in the previous chapter, see figure 3.6.

First of all, the represented map has to be compensated, either due to a movement of the ego vehicle or because of changes in the reference path estimation. Both procedures will be explained in detail in section 4.4.1. Then, the represented motion cell states are predicted according to the EKF equations which will be derived in following. After having extracted the relevant features from the object lists, the first task in the correction step is to associate the measured motion cell hypotheses with the existing ones. For this

decision, the *Mahalanobis Distance* [137] between the complete state vectors of existing and new cells is calculated, which significantly improves the association and extraction results in comparison to a restriction to the lateral positions. All associable measurements are incorporated into existing motion cell state vectors according to the rules that will be derived in the following two sections. The remaining measurements are used to establish new motion cells.

Finally, also a subsequent *cell management* step takes place, in which similar motion cells are fused to simplify the further processing of the map. Furthermore, cells whose predicted variances exceed a predefined threshold are rejected, as the represented information about traffic convoys might be outdated.

Sensor Model for Radar Object Lists

The application of an EKF requires the formulation of a measurement equation which extracts the relevant information from the provided object lists while taking into account the provided uncertainties. Although the obtained radar object lists may contain rough estimates about the lateral extent of detected vehicles, the map update is restricted to the lateral center of the motion cells' state vectors (4.1) in this case. The previous analysis showed that the location of radar reflection points on vehicles cannot be determined unambiguously. Concerning this matter, the developed approach assumes that the radar objects represent the lateral center of vehicle rears, which applies at least to the majority of the provided objects, especially in scenarios without occlusions. In case a partial occlusion is detected, the lateral variance of the measurement is additionally increased. Formally, the state variables and the measurement vector are defined as:

$$\boldsymbol{x}_k = [c, v, \phi]^T \tag{4.6}$$

$$\boldsymbol{z}_k = \left[\boldsymbol{p}, {}^{(rel)}\boldsymbol{v}, \delta, L\right]^T \tag{4.7}$$

with $\boldsymbol{p}$ being the position of the radar reflection point in the USK, ${}^{(rel)}\boldsymbol{v}$ describing the object's relative velocity, δ being the object's direction of movement and L denoting the object's length. As the motion cells' state variables always refer to the longitudinal center of the intervals, the input objects have to be extrapolated to the next center line. Whenever a bounding box length is specified, the object's center line can be used to calculate intersections with all overlapped interval center lines, as illustrated in figure 4.9. In this way, each intersection point can be quantified by the object's normalized directional vector and an inferred λ-value, see also figure 4.9.

Similar to the sensor model in the previous chapter, this derivation does not allow to formulate a measurement function of the form $\boldsymbol{z} = f(\boldsymbol{x})$. Again, the relation between the

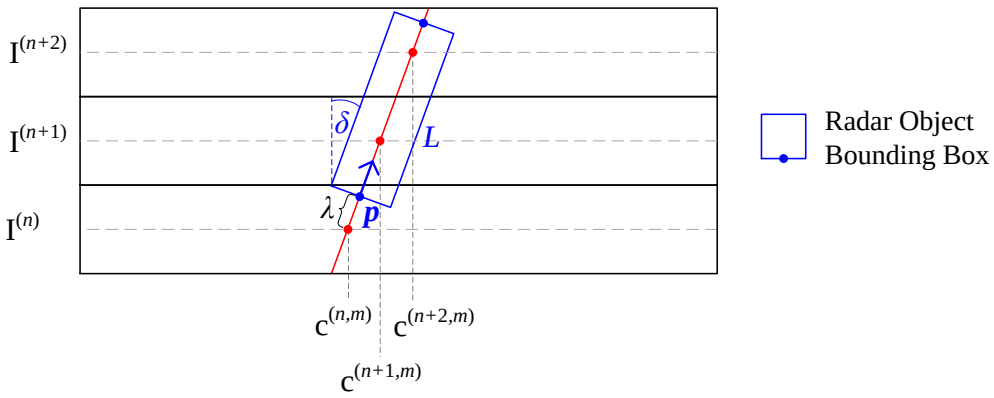


Figure 4.9.: Feature extraction from radar object bounding boxes.

introduced measurement and state vector can be used to formulate an implicit measurement equation [6, 152, 112] for each intersected interval [2]:

$$\boldsymbol{g}\left(\check{\boldsymbol{z}}_k, \boldsymbol{x}_k\right) = \begin{bmatrix} c \\ v \\ \phi \end{bmatrix} - \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{I^{(n)}}\mathcal{P}_{USK} \left(\tilde{\check{\boldsymbol{p}}} + \lambda \begin{bmatrix} \cos(\check{\delta}) \\ \sin(\check{\delta}) \\ 0 \end{bmatrix} \right) \\ {}^{Obj}\mathcal{P}_{USK} \cdot^{(abs)} \tilde{\check{\boldsymbol{v}}} \\ \check{\delta} - \psi^{(n)} \end{bmatrix} = \boldsymbol{0} \quad (4.8)$$

with $\psi^{(n)}$ denoting the orientation of the n-th interval, as introduced in the previous chapter. In this equation, the object's yaw angle and the calculated coordinates at an intersection point are transformed from the USK in the interval coordinate system. Furthermore, the provided relative USK velocity ${}^{(rel)}\boldsymbol{v}$ is first transformed into an absolute velocity ${}^{(abs)}\boldsymbol{v}$, and then converted into the object's longitudinal velocity, which corresponds to the velocity magnitude if we assume negligible slip angles. The representation of this direction-independent value allows an intuitive interpretation of the velocities along an extracted convoy and avoids further compensations of this state variable.

Similar to the occupancy map sensor model, the partial derivative of the implicit measurement equation with respect to the state vector results in an identity matrix $\boldsymbol{I}$, which simplifies the prediction step to:

$$\overset{*}{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_{k-1} \qquad \overset{*}{\boldsymbol{P}}_k = \hat{\boldsymbol{P}}_{k-1} + \boldsymbol{Q}_{k-1} \quad (4.9)$$

with $\boldsymbol{Q}_{k-1}$ denoting the process noise, which can be used to compensate for the inherent approximations of the developed approach. It is important to note that the proposed state transition model does not change the motion cells' states, as the cells represent the stationary information about the detected motion instead of the moving object itself.

[2] Due to the available input data, the derivation of sensor models is restricted to 2D homogeneous coordinates in this chapter

Based on the implicit measurement equation (4.8) and its partial derivative with respect to the measurement $\boldsymbol{C}_{\boldsymbol{z}_k}$, the following EKF correction step for state vectors of existing motion cells can be deduced:

$$\hat{\boldsymbol{x}}_k = \overset{*}{\boldsymbol{x}}_k + \boldsymbol{K}_k \left(-\boldsymbol{g}(\boldsymbol{u}_k, \overset{*}{\boldsymbol{x}}_k) \right) \tag{4.10}$$

$$\hat{\boldsymbol{P}}_k = (\boldsymbol{I} - \boldsymbol{K}_k) \overset{*}{\boldsymbol{P}}_k \tag{4.11}$$

$$\text{with } \boldsymbol{K}_k = \overset{*}{\boldsymbol{P}}_k \left(\overset{*}{\boldsymbol{P}}_k + \boldsymbol{C}_{\boldsymbol{z}_k} \boldsymbol{U}_k \boldsymbol{C}_{\boldsymbol{z}_k}^T \right)^{-1} \tag{4.12}$$

with $\boldsymbol{u}_k$ and $\boldsymbol{U}_k$ denoting the inferred equivalent measurement respectively equivalent measurement error covariance. As in the previously introduced sensor model, the implicit measurement derivation leads to a standard EKF-update equation, in which the measurement uncertainties are transformed into the state space by the Jacobian $\boldsymbol{C}_{\boldsymbol{z}_k}$.

In practice, further improvements can be achieved by additionally generating motion cells from an object's trajectory between two map updates. Taking into account the previous object location, a virtual measurement along the driven trajectory can be generated. Especially in scenarios with high object velocities, this approach improved convoy extraction results.

Sensor Model for Laser Object Lists

As shown in the previous sensor data analysis, the applied laser sensor is able to accurately detect object contours of completely visible preceding vehicles. According to the introduced object interface (2.3.2), all objects are described by using the coordinate system in which the detected bounding box can be best described. However, the object tracking module does not provide information on whether the bounding box is an estimate of the entire dimensions of an object. Only if a bounding box represents the complete width of an object, both border state variables l and r of an associated motion cell are *observable* and should be updated. If this condition is not considered for the 2DIM update, partial object bounding boxes will significantly deteriorate the motion cell width estimations.

In order to derive the information about the completeness of the bounding box estimations, the developed sensor model processes a simple occlusion analysis, whose result in an example scenario is illustrated in figure 4.10. For each provided bounding box, the radial distance and the covered angular range in the sensor's polar coordinate systems are calculated. By comparing the angular range of an object's rear to all closer located objects, potential occlusions can be detected. Whenever these angular ranges border on each other, an unobstructed view of the object's rear cannot be ensured. In this way, complete and potential left- respectively right-sided occlusions can be inferred. Fragmented occlusions have not been observed in the available object lists and have therefore not been considered in detail.

Only if the occlusion analysis indicates that a bounding box represents an estimate about the entire object width, the complete motion cell state vector including the borders l and

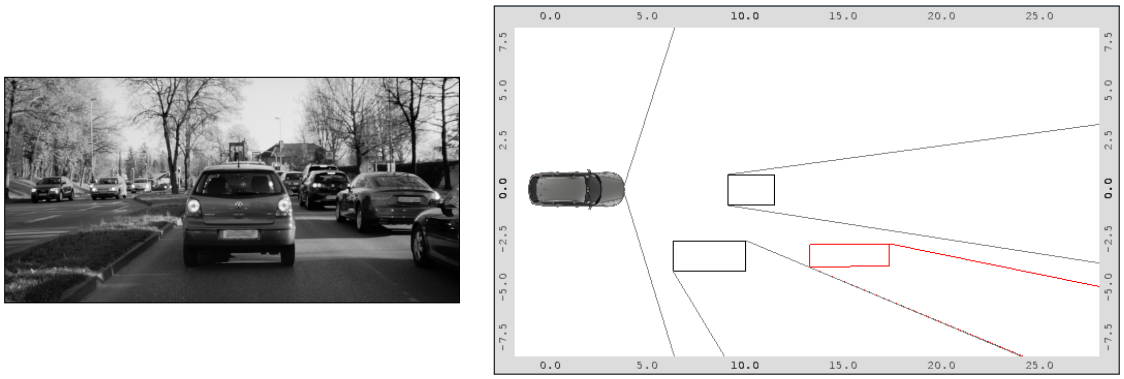


Figure 4.10.: Occlusions analysis of laser-based object bounding boxes.
Due to the illustrated laser beams, a potential right-sided occlusion of the red bounding can be inferred.

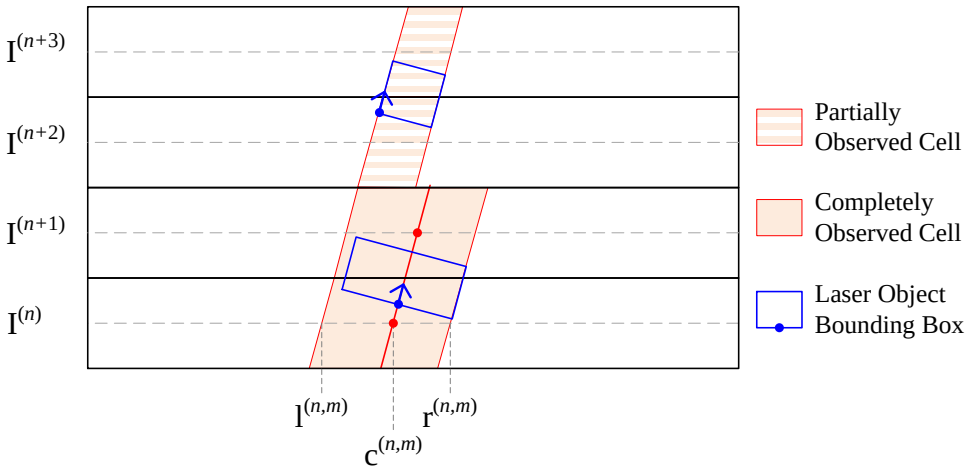


Figure 4.11.: Feature extraction from laser object bounding boxes.
The moving object in the upper part is considered occluded and can therefore only be used to estimate the cell's left border, motion direction and magnitude. By contrast, the lower object is considered completely visible, all motion cell state variable are updated.

r can be estimated. In this case, the reduced state vector (4.6) is updated similar to the radar sensor model. By contrast, the left and right boundary are inferred by using a rule-based heuristic, as they represent the maximum outer boundaries of all vehicles that have contributed to a motion cell. An example of this procedure is illustrated in the lower part of figure 4.11. Whenever the preceding analysis infers a possible occlusion of a bounding box, the observability of the motion cell's center and both borders cannot be guaranteed. Consequently, the estimation has to be restricted to the motion direction, motion magnitude and the visible border. An example for this behavior can be seen in the upper part of figure 4.11. Although the potentially occluded border can be temporarily stored in the cell, it should not be used as an origin for further estimations. By this distinction, the quality of the resulting border estimates can be improved substantially.

Similar to the radar sensor model, the driven trajectories between two map updates can also be used to establish virtual object measurements. Due to the reduced cycle time of the laser sensor, this approach is only reasonable in scenarios with high object velocities and small interval height configurations.

4.4. Development of Compensation Mechanisms and Extractors

Due to the convenient cell alignment, the 2DIM framework allows to represent information about common object motion with low interpretation and computational effort. On the downside, this approach requires the ascertainment of the spatio-temporal consistency of the represented information whenever the vehicle moves respectively the reference path estimation changes. After the development of the necessary compensation mechanisms, this section further describes the key principles of the convoy track extractor, which has already been introduced in the system architecture, see figure 4.2. By the application of this module, the required continuous information about the shape of surrounding traffic convoys can finally be inferred.

4.4.1. Modeling Longitudinal Information in Motion Cells

Due to the modified cell state vectors and semantics of the motion cells, the previously developed occupancy map compensation mechanisms need to be adapted. The derivations in the occupancy map chapter (3.4.1) showed that the translational ego motion between two straight 2DIM updates can be conveniently compensated by shifting the lateral cell coordinates and the map's circular interval buffer. However, the compensation of vehicle rotations and reference path changes in case of the curved 2DIM require the inclusion of longitudinal information within the interval cells.

Similar to the occupancy cells, one can think of several possible strategies with varying complexities to model this additional information. As the motion cells have a limited width and incorporate information about their orientation, the rotational changes can

be compensated with less approximation errors than in the occupancy 2DIM. Figure 4.12 illustrates three different approaches and the resulting transfer of information across interval borders. The most general and accurate solution would be to represent a cell's

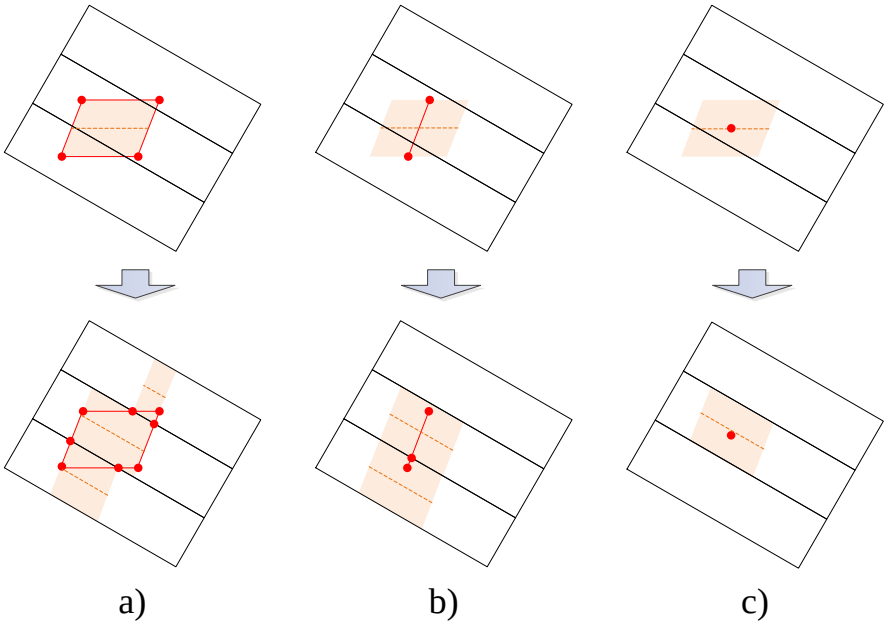


Figure 4.12.: Different strategies to model longitudinal information in 2DIM motion cells: a) Longitudinal extent of both borders, b) longitudinal extent of center line and c) reduction to a single offset value.

longitudinal extent by two values for each cell border, as shown in part a) of figure 4.12. On the downside, this approach would require a large amount of memory and complex merge operations whenever parts of cells are transferred across intervals. As the detailed shape of the convoy tracks' ends is not of major importance, strategy b) seems a reasonable compromise between computational effort and accuracy. In this case, the longitudinal start and end of the motion cell's center line have to be represented. Still, this method requires frequent cell transfer and merge operations. This disadvantage is eliminated in strategy c), where the longitudinal information is restricted to a single x-offset per cell. However, this approximation may lead to gaps between bordering cells after rotations with large angles or lever arms, as already pointed out in section 3.4.1. Within the scope of this thesis, strategies b) and c) have been implemented and evaluated. Although the required computation times of both approaches differed significantly, the obtained representations led to similar convoy results, as the developed extraction algorithm is able to bridge motion cell gaps. Detailed comparisons will be shown in section 4.5.2.

4.4.2. Compensation Algorithms for Straight and Curved 2DIM

For reasons of clarity, the derivations in this section will be restricted to the representation of a single longitudinal x-offset, as also illustrated in figure 4.13. The remaining compensation strategies can easily be deduced from the presented formulas. As all cell state variables refer to the longitudinal interval center, the first step during the compensation is to infer *subcell rotation points* from the last x-offset and direction φ. In case of the center state variable c, the rotation point within the interval coordinate system $I_{k-1}^{(n)}$ of the last map $\boldsymbol{M}_{k-1}$ can be calculated by:

$$\left(I_{k-1}^{(n)}\right)\tilde{\boldsymbol{c}}_{k-1} = \begin{bmatrix} x_{k-1} \\ c_{k-1} + \tan(\phi_{k-1}) \cdot x_{k-1} \\ 1 \end{bmatrix} \tag{4.13}$$

This point can then be transformed into an interval coordinate system $I_k^{(n)}$ of the new map $\boldsymbol{M}_k$:

$$\begin{bmatrix} \check{x} \\ \check{c} \\ 1 \end{bmatrix} = {}^{\left(I_k^{(n)}\right)}\tilde{\boldsymbol{c}}_{k-1} = {}^{\left(I_k^{(n)}\right)}\mathcal{P}_{(USK_k)} \cdot {}^{(USK_k)}\mathcal{P}_{(USK_{k-1})} \cdot {}^{(USK_{k-1})}\mathcal{P}_{\left(I_{k-1}^{(n)}\right)} \cdot {}^{\left(I_{k-1}^{(n)}\right)}\tilde{\boldsymbol{c}}_{k-1} \tag{4.14}$$

In case of the straight 2DIM, these transformations simplify to a rotation between both USKs and translations into the interval coordinate systems. After calculating the cell orientation $\hat{\phi}$ in the new interval, the rotated point has to be extrapolated to the longitudinal interval center in order to derive the compensated state variables:

$$\begin{bmatrix} \hat{x}_{k-1} \\ \hat{c}_{k-1} \end{bmatrix} = \begin{bmatrix} \check{x} \\ \check{c} - \tan(\hat{\phi}_{k-1}) \cdot \check{x} \end{bmatrix} \tag{4.15}$$

$$\text{with } \hat{\phi}_{k-1} = \phi_{k-1} - (\psi_k^{(n)} - \psi_{k-1}^{(n)}) \tag{4.16}$$

The resulting x-offset indicates whether a cell has to be transferred into an adjacent map interval. Similar to the center state variable, the border state variables l and r can also be compensated. When using compensation strategy c), all three lateral state variables are rotated under the assumption of a common x-offset.

Based on the example of a curved 2DIM, figure 4.13 illustrates the entire compensation procedure. Overall, the compensation algorithms for motion cells provided even more accurate results than those for occupancy cells. On the one hand, this can be explained by the reduced cell widths and the additional consideration of cell alignments. On the other hand, the represented information is not strictly limited in the longitudinal direction and mostly continues along interval borders. Consequently, occurring inaccuracies in the longitudinal direction do not necessarily distort the extraction result.

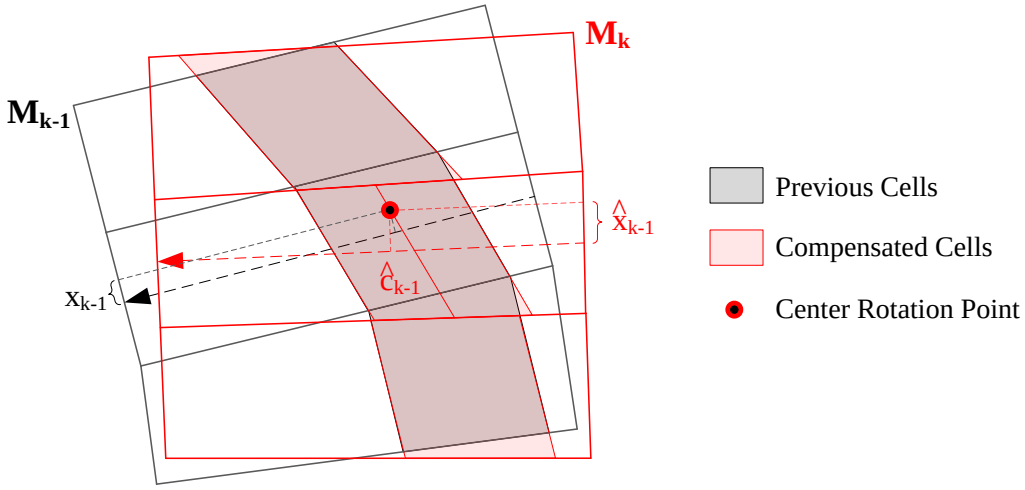


Figure 4.13.: Illustration of compensation algorithm for 2DIM with motion cells.

4.4.3. Convoy Track Extraction

The pure estimation of 2DIM cells that represent the locations of common vehicle motion does not yet provide the required information about existing traffic convoys and their center tracks respectively outer boundaries. Only after connecting several cells with similar attributes along a common path, the represented information can be used to infer the desired continuous convoy information. According to the architecture presented in figure 4.2, this step takes place in a separate extractor.

As the focus of this thesis mainly lies on the development of new environment representations, the extraction algorithm will be only briefly outlined in the following. The key idea of the algorithm is to identify traffic convoys by concatenating motion cells along several map intervals depending on the similarity of their state vectors (4.1). In challenging scenarios, the concatenation results can be further improved by incorporating a priori models about convoy shapes, road courses or extraction results from previous cycles. In order to fulfill the requirement of detecting the common motion behavior of several traffic participants, the extraction has to be restricted to motion cells which have been *confirmed by at least two vehicles*. Whenever a determined chain of confirmed cells exceeds a predefined limit, the combination of all cells is considered a valid traffic convoy. For each convoy, the contained state vectors can be assembled to polygonal chains representing the average center line and maximum outer boundaries. Moreover, an average velocity profile along the convoy can also be inferred.

For further processing during the secondary environment perception and behavior generation, the resulting polygonal lines can optionally be converted into a parametric representation. Due to their wide dissemination in the geometric design of roads, which mainly results from the assumption of continuous steering angles, clothoid models have been ap-

plied for this purpose in the scope of this thesis. The detailed approach to inferring the clothoid parameters from the previously determined polygonal chains is described in appendix A.6. An example of the extracted convoy track center clothoids from a 2DIM in an urban scene is illustrated in figure 4.14.

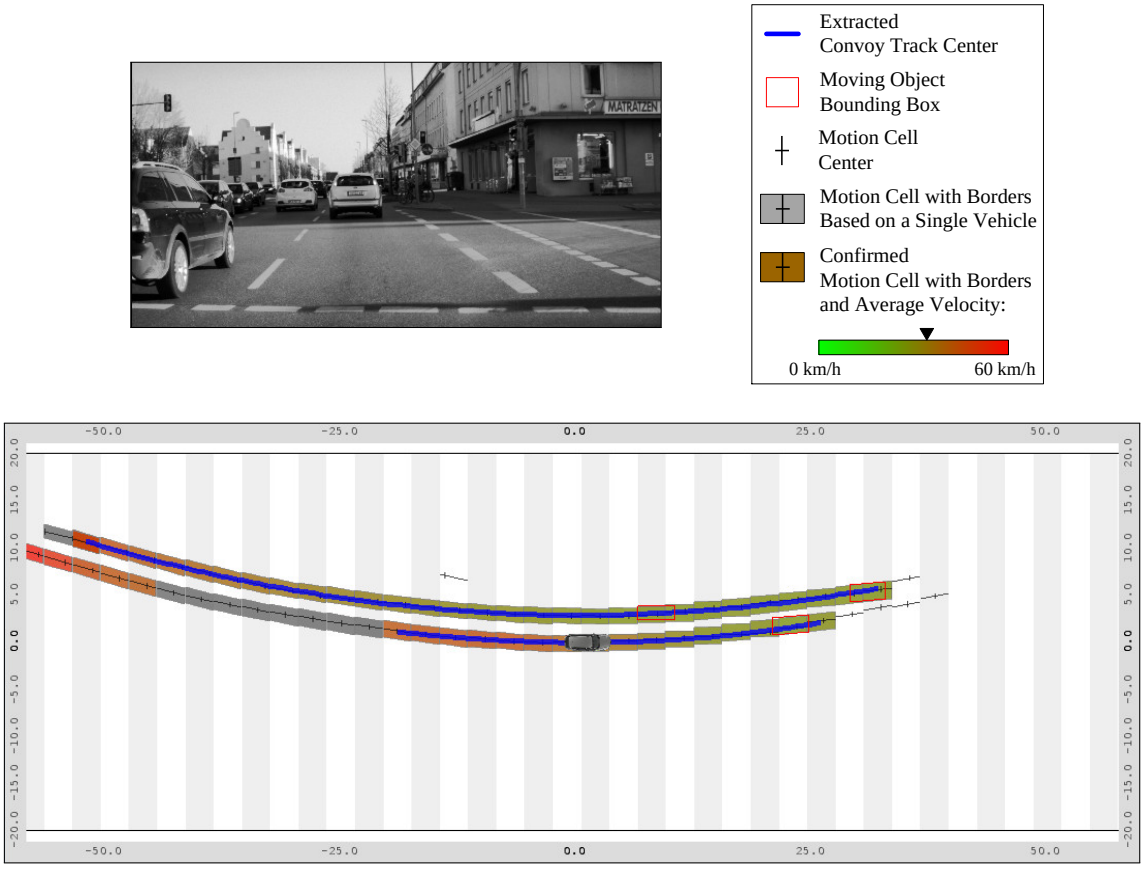


Figure 4.14.: Convoy track extraction results from a 2DIM with motion cells.

4.5. Evaluation

As the representation of common object motion behavior has not been considered so far, the presented contributions cannot be evaluated by a direct comparison to state-of-the-art approaches. Nevertheless, the validation of the obtained results requires the development of an evaluation system by which the quality of the representation and the extraction results can be quantified. By this contribution of this thesis, the performance of different sensors, the developed algorithms and future enhancements can be compared.

After identifying the objectives of the evaluation, a quantitative evaluation scheme for representations of common object motion will be developed in the following sections. As this approach has to be restricted to a limited number of reference scenarios, additional qualitative evaluations will be described afterwards. Finally, also the beneficial effects of using common object behavior information in highly automated vehicles will be illustrated and the evaluation will be summarized.

4.5.1. Objectives

The guiding questions of the evaluation are similar to the objectives identified in section 3.5.1:

- Can the information about common object motion be suitably described in a map-based environment representation?
- What are the impacts of typical laser and radar measurement errors on the obtained results?
- How does the incorporation of erroneous ego motion estimation affect the obtained results?
- What are the memory and computational requirements of the representation?
- How do the introduced approximations and possible algorithm variations, especially
  - the decorrelation of the input data
  - different ego motion compensation approximations

  influence the quality of the map?

4.5.2. Quantitative Evaluation

Similar to the previously developed occupancy map, the key idea of the evaluation approach is to quantify the quality of the obtained results by a comparison to an inferred reference representation. For this purpose, the introduced combination of reference system and environment sensors from section 4.3.1 can be applied, as it allows to simultaneously record real sensor measurements and highly accurate reference data of multiple surrounding moving objects.

Taking into account the system architecture from figure 4.2, the quality of the obtained results can be evaluated on different levels of abstraction. Figure 4.15 gives an overview of several possible evaluation strategies. On the one hand, an ideal reference map can be derived from the captured reference data and be used to evaluate 2DIMs that result from real sensor measurements and different possible algorithms (figure 4.15 A). In this way, the quality of the map-based representation of detected object motions can be quantified.

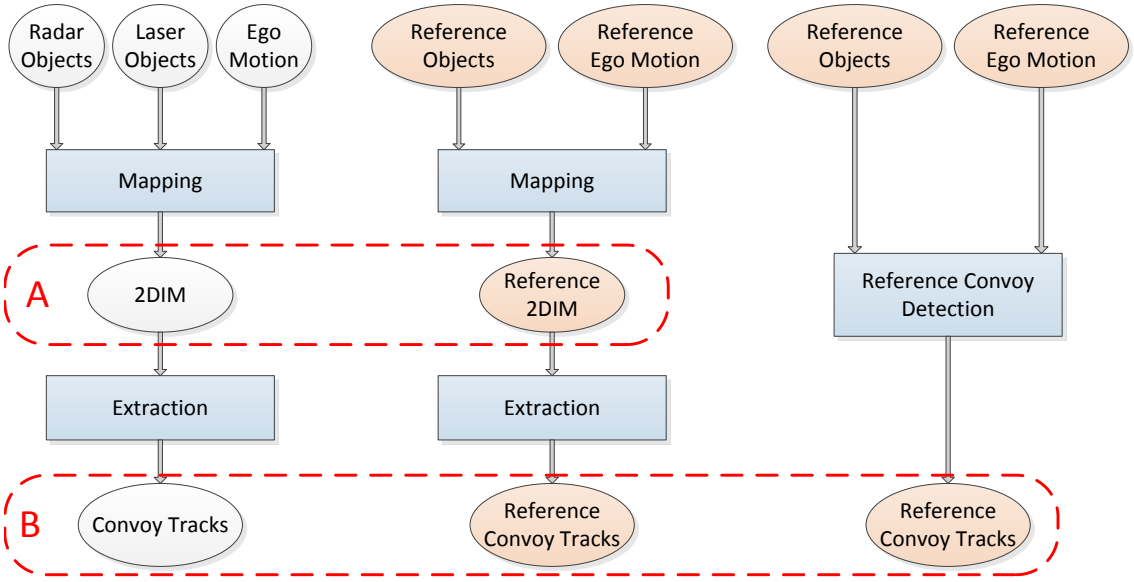


Figure 4.15.: Different evaluation strategies concerning the representation of common object motion behavior.

On the other hand, the evaluation of the obtained results can also be realized by analyzing the resulting convoy track parameters, for instance the average center line clothoid approximations (figure 4.15 B). This information can be determined by either extracting information from the reference map or directly inferring the desired parameters from the object lists without map-based representations, as illustrated in the right part of figure 4.15.

As the focus of this thesis mainly lies on the development of new map-based environment representations, the following approach is restricted to an evaluation of the obtained 2DIMs. For further works concerning the evaluation of clothoid-based representations of traffic convoy information, the interested reader is referred to [19]. The structure of the developed map-based evaluation conforms to the architecture introduced in figure 3.21 and consists of the reference map creation, the establishment of comparability and the quantification of deviations by newly developed metrics. The different steps will be described in the following sections. The results of selected evaluation experiments will be shown and discussed afterwards.

Reference Map

The goal of this step is to infer a map-based representation of the provided reference object motions, which serves as a ground truth for the evaluation of different 2DIMs. Due to the bounding box shape of the provided moving objects, the desired information can be described with sufficient accuracy within an interval-based map with directional cells. As

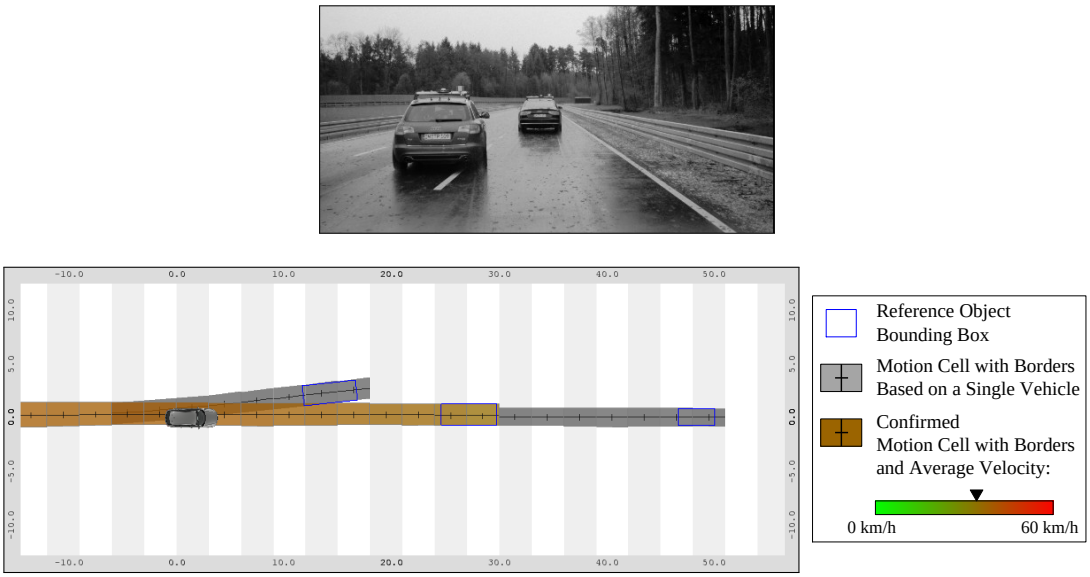


Figure 4.16.: Reference 2DIM with motion cells showing a lane change maneuver.

not all test setups were able to provide a reference path estimation, the ideal map was realized by a straight 2DIM with decreased interval sizes in order to reduce the impacts of discretization errors.

However, the reference map should not be created by using the previously developed sensor models and compensation algorithms, as they are adapted to sensor-specific data and make use of approximations. In order to obtain the most accurate result, all received reference objects are stored within a pre-processing module of the 2DIM. After transforming all stored object states into the current USK, the object rectangles can be assigned to the different map intervals. Within each interval, the measurements of different vehicles are added to a common motion cell whenever their average state vectors are similar. By this computationally expensive approach, an accurate reference map can be inferred without using cell compensation approximations. An example of a reference map showing a lane change maneuver is illustrated in figure 4.16, the corresponding scenario is described in appendix D.2. The resulting map already indicates the good suitability of the developed approach to represent the required information: Due to the consideration of the motion directions in the map cells, the course of the lane-changing vehicle can easily be separated from the remaining trajectories. As a result, the common motion of the vehicles continuing straight can be inferred from the confirmed motion cells in the map.

Establishing Comparability

As the quality of both straight and curved interval-based maps needs to be evaluated, the next step is to generate evaluation features which allow for the comparison of maps with

different geometries. The key idea of the developed approach is to identify the represented motion cells along lateral lines in equidistant longitudinal USK distances, as the extraction from a straight 2DIM is then simplified to the determination of the intervals within which the sample lines are located. The entire sample generation process for both straight and curved 2DIMs is shown in figure 4.17. Assuming the evaluation of a curved 2DIM, the

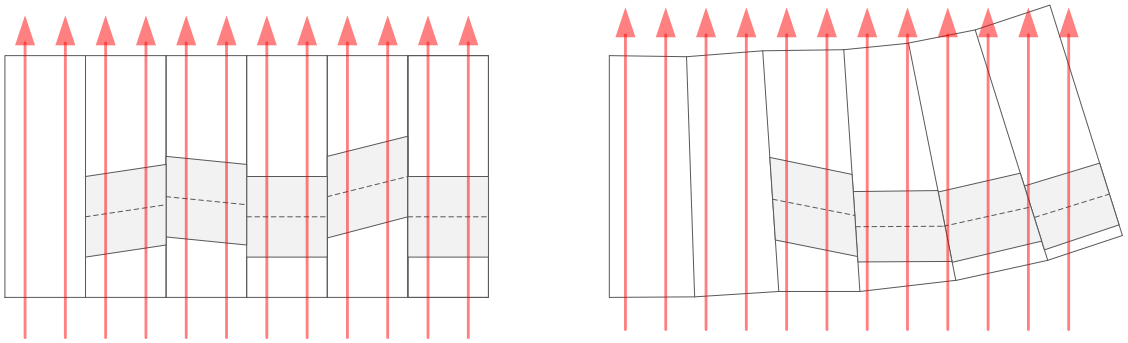


Figure 4.17.: Extraction of evaluation features from straight and curved 2DIMs containing motion cells.

required information needs to be determined by intersecting the sample lines with the represented motion cells, as also illustrated in the right part of figure 4.17. The extracted motion cells along the sample lines can then be used as input features for the comparison of a reference map and evaluated map.

Calculation of Metrics

As the representation of motion cells within a map interval can be characterized as model-based object representation, the problem of comparing extracted motion cells is strongly related to evaluations of multi-target object tracking systems. For this application, the development of metrics to quantify the difference between a set of ground truth objects and determined tracks has been investigated in numerous publications. According to Bashir et al., they can be classified into *frame-based metrics*, which are computed independently for each frame of a sequence, and *object-based metrics*, which consider the complete trajectory and lifespan of an object [12]. As the 2DIM motion cells do not obtain a trajectory and their meaning considerably differs from tracked objects, the analysis of existing implementations is limited to frame-based metrics. They are mainly derived from the well-known confusion matrix entries of a binary classifier and calculate the number of True Positives (TP) (*hits*), False Negatives (FN) (*misses*) and False Positives (FP)[12, 150]. By contrast, the number of correct model-based track absences (True Negatives (TN)) can only be derived indirectly. Several popular works deal with the incorporation of the de-

scribed values into a single measure, e.g. the widespread evaluation metrics MOTP[3] and MOTA[4] [13].

However, most of these approaches assume an unambiguous association between the ground truth and tracked objects, which is not tenable especially for the evaluation of motion cells. Possible association ambiguities are mostly evaluated by object-based metrics [151, 153], or by counting the number of multiple assignments, e.g. in case of the metrics *multiple trackers* and *multiple objects* [150]. A more sophisticated approach is presented by Kasturi et al. in [81]: In order to quantify spatiotemporal overlaps, they establish an optimal one-to-one correspondence between ground-truth and system output objects by calculating a metric distance matrix.

The idea of finding an unambiguous mapping between ground-truth and estimated objects also forms the basis of the newly developed motion cell evaluation metrics. As this problem corresponds to finding a matching with minimal costs in a complete bipartite graph, a one-to-one mapping with optimal Mahalanobis Distances can be found by using the Hungarian algorithm in $O\left(n^4\right)$ [157, 92]. The association between reference and estimated cells can then be accomplished by comparing the inferred optimal distances to a predefined threshold. Based on this result, the quantities TPs, FNs and FPs can be inferred despite possibly unambiguous associations. In order to summarize the obtained results, these values can further be used to calculate the combined metrics *precision* and *recall*, which represent the number of correctly detected cells in relation to all represented cells respectively reference cells:

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN} \tag{4.17}$$

Optionally, both metrics can also be combined into the so called F_1-Score [130]:

$$F_1 = 2 \cdot \frac{Precision * Recall}{Precision + Recall} \tag{4.18}$$

Besides these descriptions of the availability of motion cells, a second evaluation component aims to quantify the similarity of the associated cells. For all motion cell state variables, mean distance values are calculated. They will be denoted as center, border, angle respectively velocity error and can be quantified either for individual intervals or as an average value for the complete map.

Experiment 1: Straight motion cell 2DIM quality evaluation

The aim of the first investigation in this section is to analyze and compare the quality of several straight 2DIMs obtained by real sensor measurements in different scenarios. A detailed description of the evaluated test cases can be found in appendix D.2, the scenarios will be referred to as S2a to S2e.

[3] Multiple Object Tracking Precision
[4] Multiple Object Tracking Accuracy

For scenario S2a, a simulated traffic convoy without lateral offsets, figure 4.18 gives an impression of the characteristic differences between the ideal reference map and the actually achievable representation. The most obvious deviations in the front section of the maps result from the available object information: While the reference map incorporates the knowledge about the entire object dimensions at any point in time, the evaluated map has to rely on the provided laser and radar sensor measurements. Due to the visibility conditions in scenario S2a, the laser object list is restricted to the preceding vehicle, the remaining vehicles can be deduced from the corresponding radar reflection points, which are neither continuously available nor provide information about the vehicle shapes (see section 4.3.1).

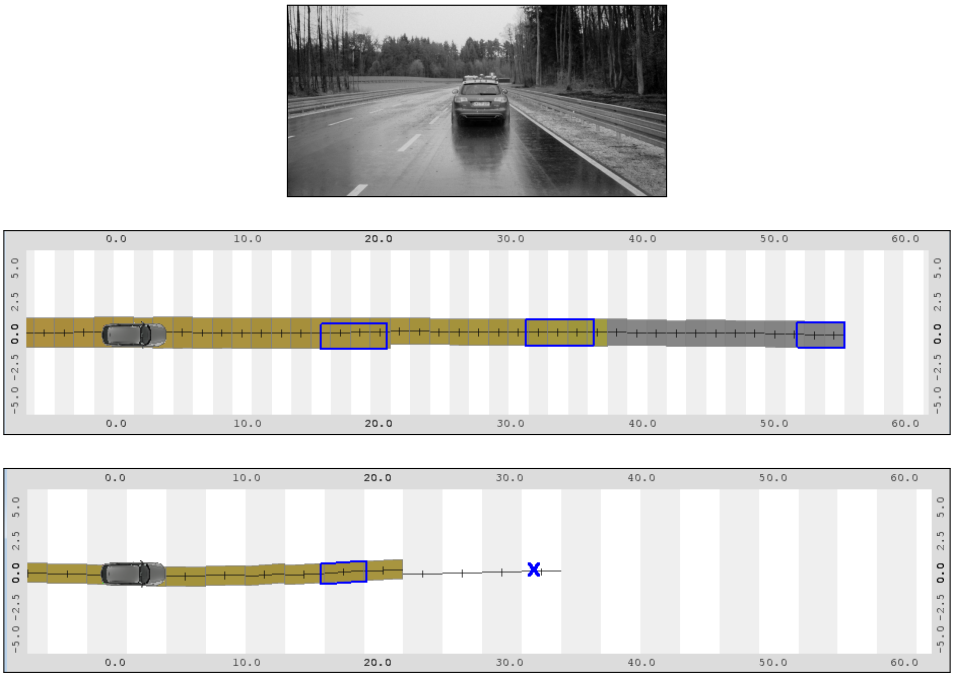


Figure 4.18.: Comparison of reference (top) and test map (bottom) in scenario S2a.

In order to generate comparable evaluation features, the previously introduced sampling mechanism was configured to extract motion cell information at a distance of 1 m in the range between 25 m behind and 75 m in front of the vehicle. If we apply the developed metrics to compare the extracted samples, the resulting values reflect errors caused by sensor measurements and the developed processing algorithms. Figure 4.19 shows the development of several selected metrics during an exemplary recording of S2a. Due to the absence of false positive motion cells in all resulting maps, the precision value is constantly 1, whereas the recall value fluctuates around a rate of 0.8. As the visibility conditions are constant in the analyzed scenario, all depicted metrics are relatively stable over the entire recording. The average distance error of the estimated motion cell centers is below 0.2 m, the average border error below 0.5 m, the average alignment error below 3°.

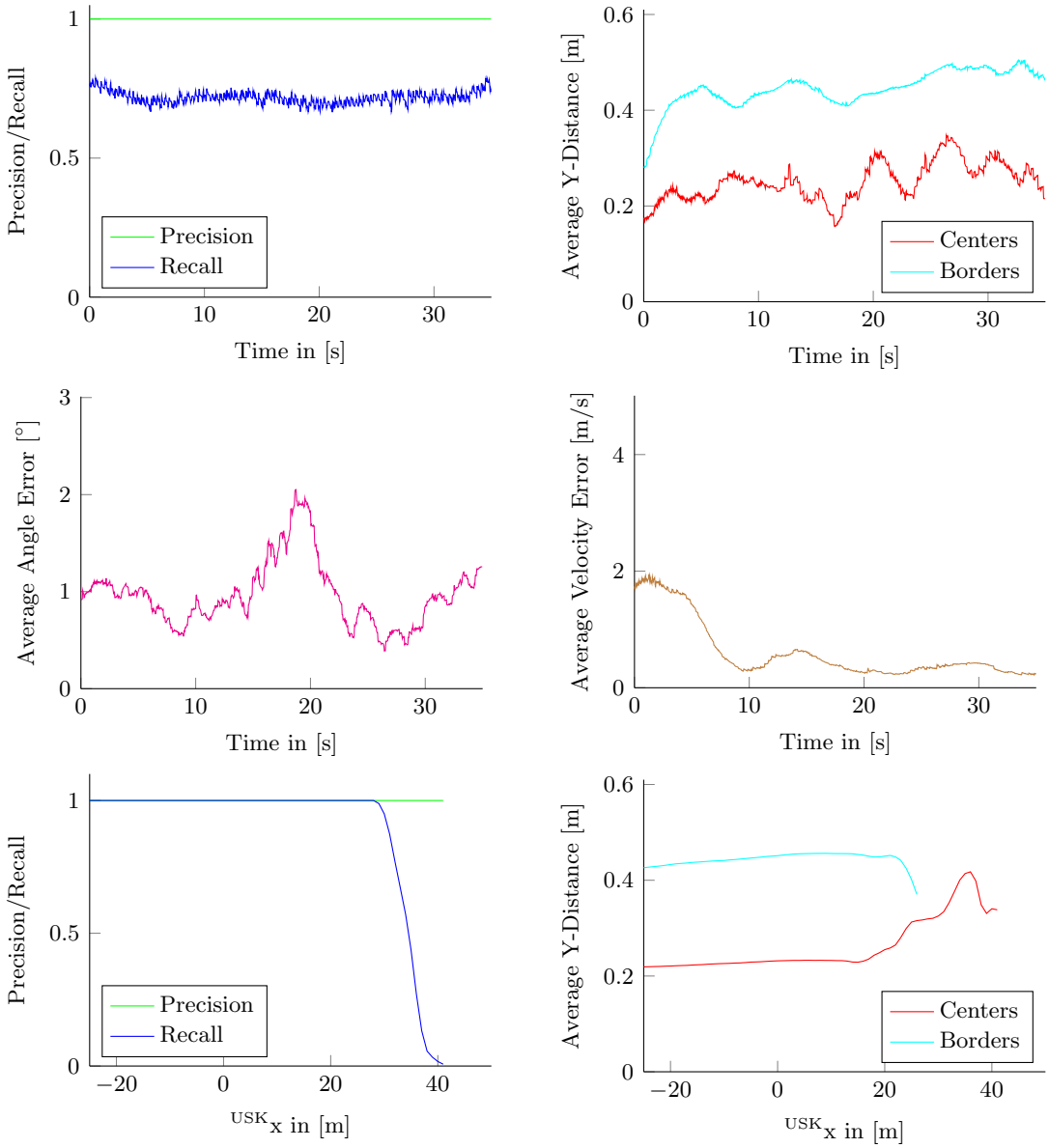


Figure 4.19.: Resulting evaluation metric values in scenario S2a.

For a better understanding of the resulting values and the underlying differences between reference and evaluated maps, figure 4.19 also illustrates the average precision and recall values along the longitudinal sampling positions. According to that, the recall value decreases with an increasing distance from the vehicle, which indicates that the deviations between both maps are mainly caused by the limited detections of the applied sensors in large distances. For all sample lines below a longitudinal distance of 5m to the vehicle, neither false positive nor false negative motion cells were detected.

The analysis of the remaining scenarios S2b-S2e provided overall similar results, the averaged metric values are summarized in table 4.1. In comparison to S2a, the evaluation of the scenarios S2b and S2c resulted in increased recall values, which can be explained by the improved radar sensor detections of laterally shifted preceding vehicles. On the downside, the evasion maneuver in S2b has a negative impact on the average motion cell alignment error, whereas the shifted vehicles in S2c decrease the quality of the extracted lateral positions, which is mainly the result of misinterpreted radar reflection points. Concerning the precision and recall values, the evaluation of S2d provided identical values as S2a. However, the motion cells originating from the vehicle in the neighboring lane are sometimes fused with motion cells from the preceding vehicles, which impacts the overall lateral position errors. The additional consideration of lane change maneuvers in scenario S2e increased the resulting recall values but also the cell alignment errors due to inaccurate object alignment estimations during the turning maneuvers. As table 4.1 also indicates,

Scenario	F1	Prec.	Rec.	CE [m]	BE [m]	AE [°]	VE [m/s]	CT [ms]	MR [kByte]
S2a	0.83	1.00	0.71	0.25	0.45	0.96	0.61	0.11	3.0
S2b	0.88	0.95	0.81	0.19	0.36	1.77	1.02	0.11	3.4
S2c	0.88	0.99	0.82	0.48	0.69	1.08	0.47	0.11	3.8
S2d	0.83	1.00	0.71	0.51	0.69	1.32	0.59	0.12	5.7
S2e	0.92	0.99	0.83	0.29	0.42	2.39	0.49	0.12	5.6

Table 4.1.: Comparison of calculated metric values in different scenarios.
(F1 = F1-Score, Prec. = Precision, Rec. = Recall, CE = Cell Center Error, BE = Cell Border Error, AE = Cell Angle Error, VE = Cell Velocity Error, CT = Computation Time [5], MR = Memory Requirement)

the average computation times for the complete 2DIM update cycle were considerably below 0.2 ms. Similar to the occupancy map evaluations, the provided memory sizes refer to the maximum memory required for the internal data representation during the processing of a recorded file. For these evaluations, the map size was set to 180 × 40 m (x × y) at a longitudinal interval discretization of 3 m.

[5] All computation times in this thesis were measured on a laptop with Intel Core i7-3840QM CPU with 16 GB RAM.

Experiment 2: Variation of Interval Discretization

In order to investigate the impact of the longitudinal discretization on the map quality, scenario S2e has been additionally evaluated with different interval height configurations, this time at a map size of 100 × 40 m. The resulting metric values are summarized in

Interval Height [m]	F1	Prec.	Rec.	CE [m]	BE [m]	AE [°]	VE [m/s]	CT [ms]	MR [kByte]
1	0.90	0.99	0.82	0.27	0.40	2.31	0.49	0.12	10.0
2	0.90	0.99	0.83	0.27	0.40	2.30	0.49	0.12	5.1
3	0.90	0.99	0.83	0.29	0.42	2.39	0.49	0.11	3.5
4	0.90	0.98	0.84	0.29	0.41	2.32	0.49	0.11	2.7
10	0.91	0.96	0.86	0.32	0.45	2.32	0.56	0.11	1.2

Table 4.2.: Comparison of average metric values with different interval height configurations in scenario S2e.

table 4.2. For interval heights between 1 and 4 m, almost identical metric values were measured, which indicates that the required information can be represented very roughly in longitudinal traffic scenarios. Negative discretization effects occur at a large interval height of 10 m, when the cell center, border and velocity errors are slightly increased. The measured computation times and memory requirements are overall very low, which is also an effect of analyzing simplified reference scenarios with a limited number of vehicles. The effects of applying the developed mechanisms in real world traffic scenarios will be shown section 4.5.3.

All in all, the calculated metric values do not sufficiently represent the ability of the map to describe longitudinal positions, e.g. the starting position of lane change maneuvers. In order to represent this information adequately, the interval discretization was set to 3 m in the remaining experiments.

Experiment 3: Variation of Ego Motion Estimation

The goal of the third investigation is to determine the sensibility of the developed approach to characteristic ego motion estimation errors. For that purpose, several test scenarios have been performed by using the conventional ego motion estimation (see section 2.3.1) and the highly accurate reference system (see section 2.3.3). For the calculation of the metric values, the map interval height has been set to 3 m, the evaluation samples were drawn at a distance of 1 m in a range between -25 m and +70 m in the USK. The resulting values for the test scenarios S2a and S2f are listed in table D.1 in appendix D.3.1.

For scenario S2a, the averaged metric values show hardly any difference. As the ego vehicle is moving in a straight line in this case, the ego motion estimation errors are

very low and mainly concern the estimation of the longitudinal movements, which does obviously not affect the measured map quality. Due to the curve in scenario S2f, the ego motion estimation becomes more challenging, especially the determination of the yaw angle differences between two map updates. These errors affect the results of the motion cell association, consequently also the measured metric values deteriorate, as table D.1 illustrates.

Experiment 4: Decorrelation of input data

The next experiment aims to quantify the effects of using correlated and decorrelated object data for the estimation of 2DIM motion cells. As already stated in section 4.3.3, the track decorrelation could only be realized for laser-based object tracks, which were provided by the introduced object tracking module (see section 2.3.2). For the evaluation, scenario S2e has been tested with two different approaches: In the first approach, the estimated object states were directly used as input data for the motion cell estimation, whereas in the second approach, they were replaced by the calculated equivalent measurements, according to the equations derived in section 4.3.3.

The resulting metric values in table D.2 in the appendix do not indicate a clear advantage of one of both strategies. While the lateral distance errors marginally increase when using object tracks, the precision and recall values deteriorate when using equivalent measurements. This particular disadvantage can be explained by missing object detections, which may be compensated by predicted hypotheses in case of using tracked objects. On the other hand, using correlated input data and hence violating the KF application requirements only has little effect on the resulting map qualities, which can be attributed to the strong influence of the measurements to the provided object hypotheses and their partially inaccurate error covariance values, as already suggested in section 4.3.3. Concerning the computation times, the calculation of equivalent measurements causes a manageable additional effort.

Experiment 5: Different Modeling of Longitudinal Information

In chapter 4.4.1, two different strategies for modeling longitudinal information in motion cells have been proposed: A simplified approach, in which the longitudinal information is reduced to a single offset value per cell, and a more complex variant, in which the longitudinal extent of the center line is modeled by a start and end point. For a better understanding of both strategies, figure 4.20 compares two exemplary maps recorded in scenario S2f. For the sake of clarity, the visualization is restricted to the motion cell center lines in both cases. While the x-offset positions in the upper map in figure 4.20 are indicated by a horizontal bar, the lower map illustrates the cells' longitudinal extent by two arrows. The main disadvantage of the simplified approach is obvious: Due to the ego motion compensation in this curved scenario, some intervals contain duplicate cells with

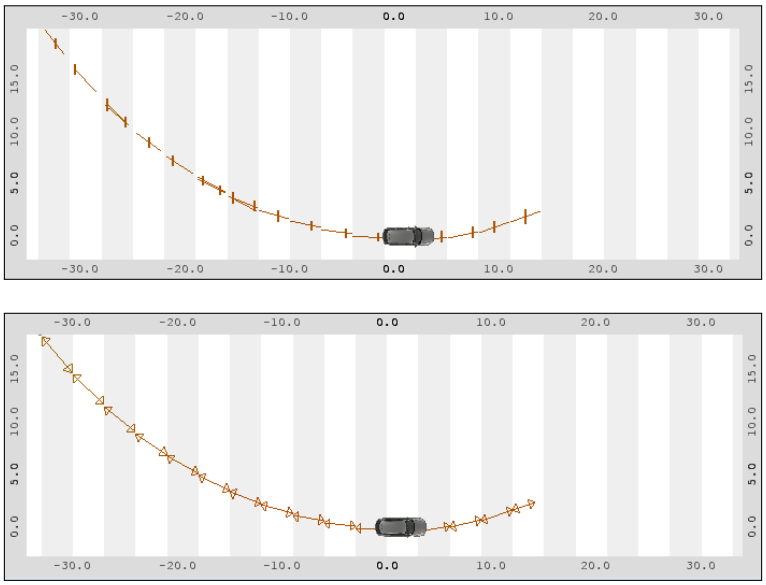


Figure 4.20.: Comparison of different approaches for modeling longitudinal information in motion cells: point (top) and line (bottom) approach.

identical lateral positions but different x-offsets. By explicitly modeling the longitudinal extent of a cell, this effect can be avoided.

The calculated metric values in table D.3 confirm this impression. By using the longitudinal line approach, precision and recall ratios as well as the average state variable errors are improved. On the downside, this strategy also increases the memory demand and computation times. Although the measured figures are still very low, the calculation effort increases by approximately 30 %. As already indicated in section 4.4.1, both maps in figure 4.20 lead to identical convoy track extraction results, as the developed extraction algorithm is robust against duplicate cells and gaps.

Experiment 6: Curved 2DIM evaluation

The great advantages of attaching a map-based representation of common object motions to the road infrastructure can be recognized from the example in figure 4.21. The illustration compares two different 2DIMs which have been created during the approach of a roundabout. While the straight map in the upper part of the figure is carried along the ego vehicle, the curved 2DIM in the lower part of the figure has been attached to the approached roundabout. For this experiment, the information about the position and radius of the roundabout originated from a preceding measurement with a highly accurate localization system. Similarly, the required parameters can also be inferred by using onboard sensors and digital map information [131].

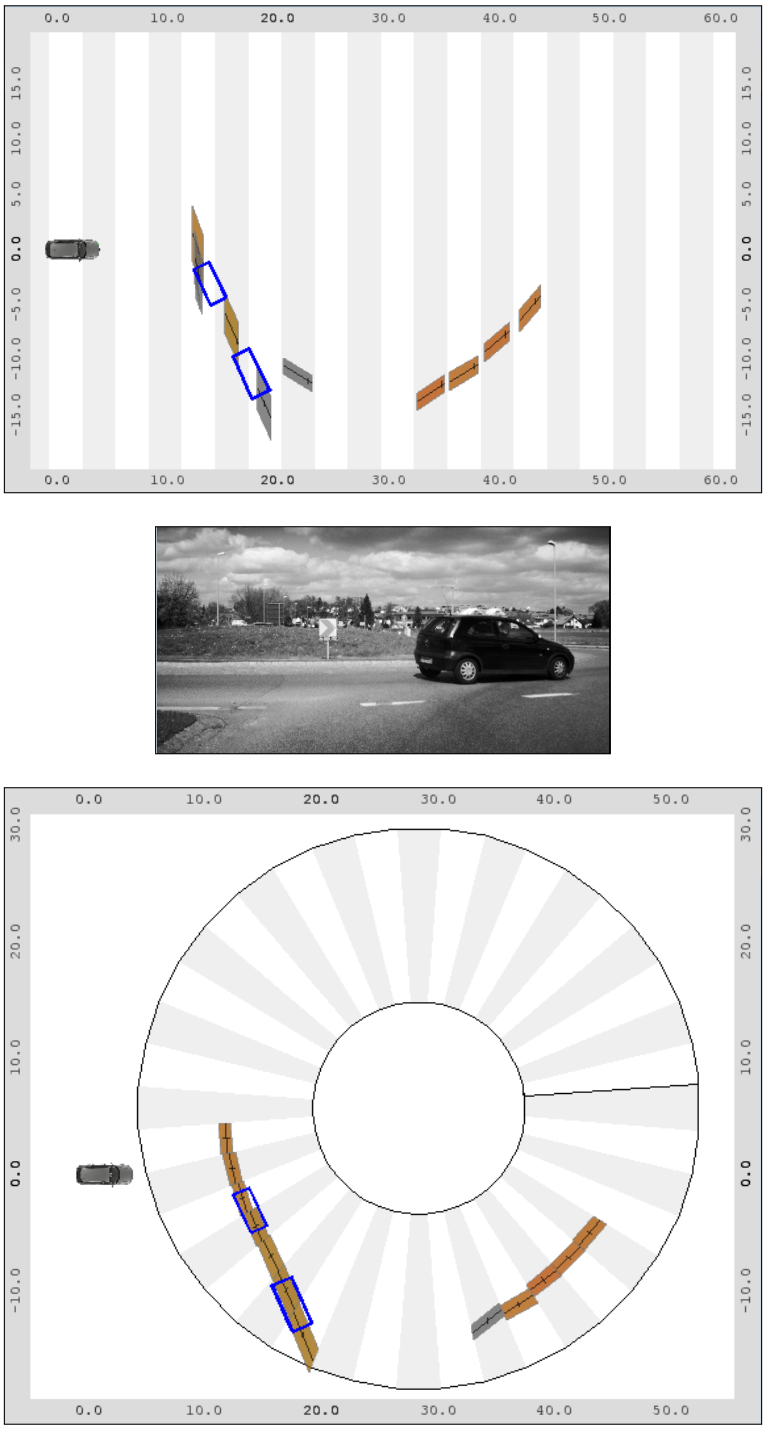


Figure 4.21.: Comparison of straight (top) and curved (bottom) 2DIM at a roundabout scene.

The comparison of both maps illustrates the advantages of aligning the 2DIM intervals orthogonal to the road direction. In case of the straight map, the crossing vehicles move parallel to the interval direction, which complicates the association and interpretation of the emerging motion cells. By contrast, the curved map alignment allows to conveniently accumulate the common motion of the surrounding vehicles and the extracted convoy information can e.g. be used to determine the exits from the roundabout.

This qualitative impression can be confirmed by the resulting evaluation metric values in the curved scenario S2f, see table 4.3. For this experiment, the alignment of the curved map was based on a simple estimation of the future trajectory depending on the current ego yaw rate. All listed map quality metric values show an improvement by applying the

Approach	F1	Prec.	Rec.	CE [m]	BE [m]	AE [°]	VE [m/s]	CT [ms]	MR [kByte]
Straight 2DIM	0.79	0.81	0.77	0.72	0.85	2.98	0.48	0.11	3.0
Curved 2DIM	0.83	0.89	0.78	0.70	0.83	2.80	0.48	0.11	7.9

Table 4.3.: Comparison of average metric values with straight and curved 2DIM in scenario S2f.

curved mapping approach. Of course, the description of the curved interval geometries also considerably raises the memory consumption of the map. Concerning the computation times, a similar value was measured for both variants, which can be explained by two effects which offset each other: On the one hand, the curved map requires more complex coordinate transformations, on the other hand, the compensation effort of transferring motion cells across interval borders is reduced.

4.5.3. Qualitative Evaluation in Traffic Jam Scenarios

Besides the quality evaluations in artificial reference scenarios, the detection and processing of convoy tracks has been tested extensively in complex real-world scenarios to improve the perception of the road infrastructure. In the secondary environment perception, the resulting information can be considered as an additional input to infer corridors, within which an automated vehicle has to stay. This especially offers the advantage of replacing missing information and validating the plausibility of conventional information sources for lateral vehicle control e.g. detected road markings from camera images or road boundaries in occupancy maps. As the details about the further processing of convoy information would exceed the scope of this thesis, the interested reader is referred to the publication [BHN+12], which presents a rule-based algorithm for fusioning convoy track, road marking and road boundary information.

A characteristic traffic situation, in which the additional consideration of convoy tracks turned out to improve the robustness of the lateral vehicle control are highway traffic

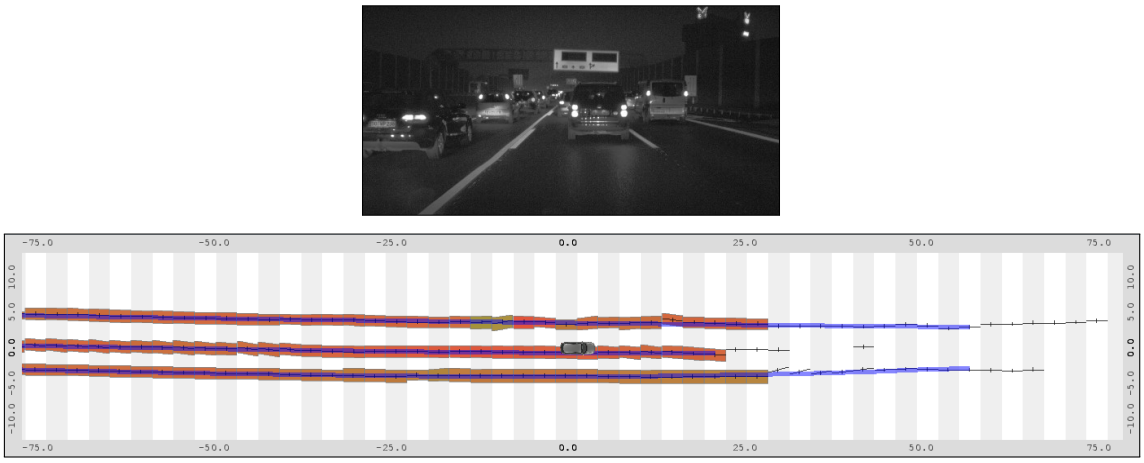


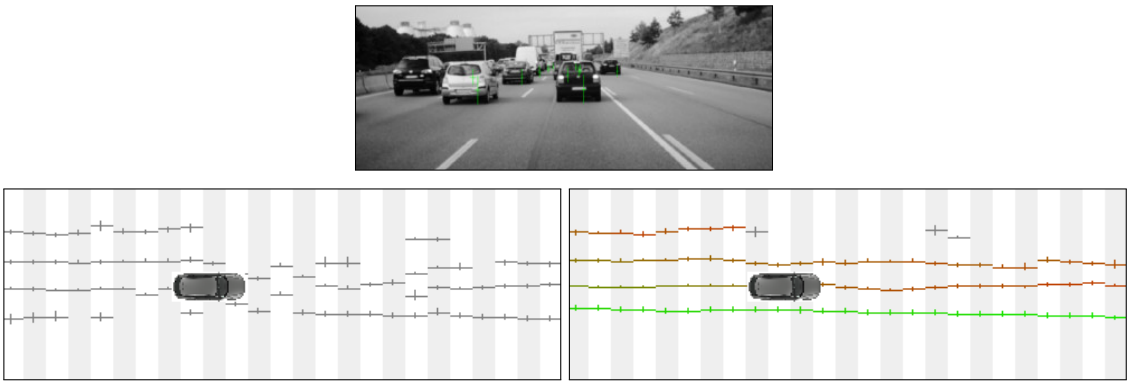
Figure 4.22.: 2DIM with motion cells and extracted convoy tracks (thin blue lines) in a highway traffic jam scenario.

jams. In these situations, the detection of road markings and road boundaries is especially complicated due to occlusions by surrounding vehicles. Furthermore, the collective motion of the vehicles may also deviate from the lanes that are defined by the road markings, as already introduced by the accident scene example in figure 1.7.

Due to the high number of surrounding vehicles in traffic jam scenarios, the developed approach usually succeeds in detecting convoy tracks in both ego and neighboring lanes. Figure 4.22 shows an example of a 2DIM and the extracted convoy track center lines in a three-lane highway traffic jam. The major differences in the obtained test results mainly concern the preview length of the detected convoy tracks. Lane change maneuvers, changing occlusion conditions and the described effects of the radar reflection points can complicate the correct association of the motion cells at large distances.

An important measure to improve the extracted convoy track lengths in these situations has already been outlined during the definition of the motion cell state vectors (see section 4.2.2): By incorporating the motion direction and velocity for the association of new object measurements to existing motion cells, the separation of different queuing lanes can be simplified. To give an impression of the impacts of this enhancement, figure 4.23 compares two 2DIMs obtained with and without using object velocities for the association decisions.

In comparison to the artificial reference scenarios, the high number of surrounding vehicles in traffic jams also causes slightly increased resource requirements. During the tests, the maximal memory demand for the internal representation of the map was about 17 KB whereas the average computation time was below 0.2 ms.



a) b)

Figure 4.23.: Resulting 2DIMs in a highway traffic jam scenario a) without and b) with using velocities for the association decisions.

4.5.4. Summary

Based on the described experiments, the following conclusions concerning the previously identified goals of the evaluation can be made:

- The developed concept of mapping common object motions in a 2DIM and extracting continuous convoy tracks succeeded in gaining the required information in artificial reference scenarios and real world traffic jams.
- When comparing the reference maps to 2DIMs with radar- and laser-based motion cells, the following main differences can be observed:
  - Due to occlusions, the estimation of motion cell borders is usually restricted to the direct environment of the vehicle.
  - In comparison to the reference maps, also the preview area of the estimated motion cell centers is reduced, as the radar sensors are not able to reliably detect all surrounding occluded vehicles.
  - In large distances, the uncertainties about the actual positions of the radar reflection points can lead to wrong associations and consequently reduced convoy track preview lengths.
  - Inaccuracies in the provided object motion directions from the laser and especially radar sensors can lead to difficulties in separating lane change maneuvers from convoys.
- In longitudinal traffic scenarios, the characteristic ego motion estimation errors did not have a negative impact on the 2DIM quality. In curved scenarios, inaccurate estimations of the vehicle's yaw movement may impede the motion cell association and hence deteriorate the mapping results.

- In all tests, the measured computation times of the mapping process were below 0.2 ms, while the memory demand for the internal map representation was below 20 KB. Both quantities are very low, even in the context of automotive control units.
- The decorrelation of laser-based object tracks improved the motion cell state vector errors in the artificial reference scenarios. In real-world scenarios, no characteristic visual differences could be determined.
- The explicit modeling of the motion cells' longitudinal extent improves the map quality at the expense of also increasing the computational requirements. However, the advanced and simplified compensation approach lead to similar convoy track extraction results.
- Similar to the representation of occupancy information, the curved mapping approach demonstrates great potential, as the mapping results are improved, the compensation effort is reduced and the interpretation of the represented information is further simplified.

Beyond that, the general approach of inferring convoy tracks from common object motion turned out as an essential additional input for the behavior generation of highly automated vehicles. Especially in dense traffic application scenarios, the additional consideration of convoy track information significantly improved the robustness of the vehicle's lateral control.

5. Combinations of Map-based Environment Representations

In the preceding chapters, the conventional grid mapping approach has been enhanced to a new generic map-based environment representation, which is capable of describing occupancy states and moving object motions. These advancements raise the question of how to integrate several map-based environment representations into the environment perception architecture of a highly automated vehicle. The aim is to optimally apply the newly developed approaches with regard to the identified representation characteristics and the overall requirements on the environment perception process.

5.1. Related Works

In the literature, several strategies concerning the combination of multiple occupancy grids for robotic and ADAS applications can be found. In one of the first publications about occupancy grid mapping [45], Elfes proposed a system architecture for mapping unstructured robot environments, which is also depicted in figure 5.1. He introduced a multi-level

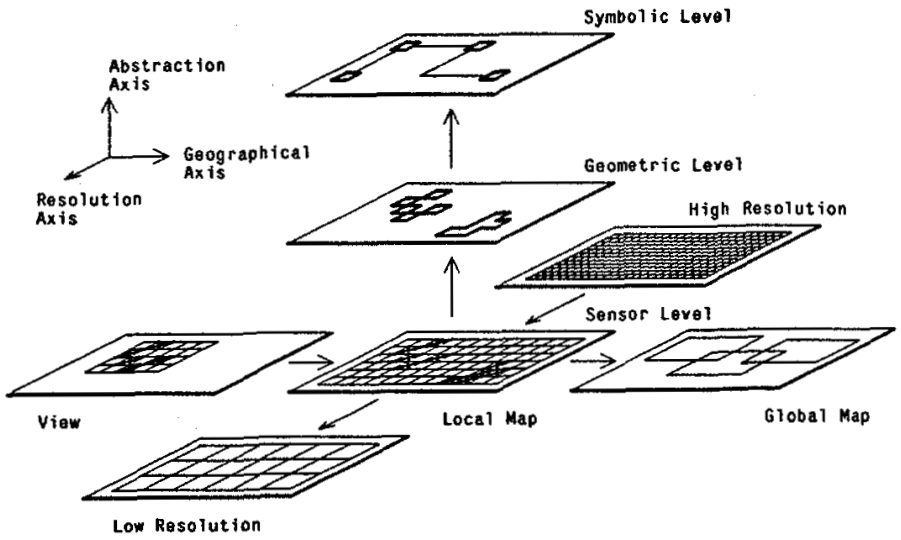


Figure 5.1.: Multiple axis of occupancy map representations, from [45].

description of the surrounding occupancy states by several maps that differ along an *abstraction*, *geographical* and *resolution axis*. In this terminology, the accumulation takes place in the sensor-level local map, which can be restarted with either increased or decreased resolution if desired. Similar to the introduced extraction concept (see figure 1.5), representations with higher levels of abstraction are inferred to fulfill the requirements of the path planning. Along the geographical axis, several local maps can be combined to a global map which covers wider spaces, e.g. the robots entire field of application.

Related concepts can also be found in more recent publications. In general, tree-based mapping approaches, whose advantages and disadvantages have been analyzed in detail in section 3.1, can be interpreted as a combination of several maps which have been varied along the resolution axis. Montemerlo and Thrun introduced a similar approach called multi-resolution pyramids for outdoor robot terrain mapping, in which the subdivision into multiple representations becomes more obvious [107]. In order to improve the real-time capability and to optimize the memory consumption of mapping applications, further variations along the geographical axis have been proposed. In [89], Konrad et al. presented a subdivision of the overall mapped area into grid patches, whose locations are identified in a global coordinate system. Whenever the vehicle leaves a grid patch, the left patch is written to the hard disk and a new patch is initialized or reloaded depending on the vehicle's previous locations. With the same objective and mechanisms, Wang proposed to generate several stationary local maps along the trajectory of a vehicle [178]. Besides that, several independent object-local maps are used to accurately describe the shape of moving objects. The idea of attached object-local maps has also been presented in [140, 42, 5].

To the author's knowledge, the presented approaches concerning this matter are restricted to the introduced combinations of multiple occupancy grids. Although grid-based maps have also been used for other contents, e.g. to represent road markings [70], height information [72, 16] or perspectively mapped camera images [88], a combination and simultaneous estimation of these map contents has not been considered so far. Due to the restriction to grid-based representations, combinations of maps with different geometries have neither been investigated in existing approaches.

5.2. Architecture of Combined Representations

The underlying idea of the developed approach is to consider the entire map-based environment representations as a set of several *map instances*, which, themselves, contain multiple *map layers* with different contents. One exemplary realization of this concept is illustrated in figure 5.2.

The contained map instances can use different environment discretizations, either the conventional grid-based or the newly developed interval-based approach. All included map layers have to conform to the chosen discretization type, but may offer deviating resolutions, at least within certain limitations that will be derived in the following section.

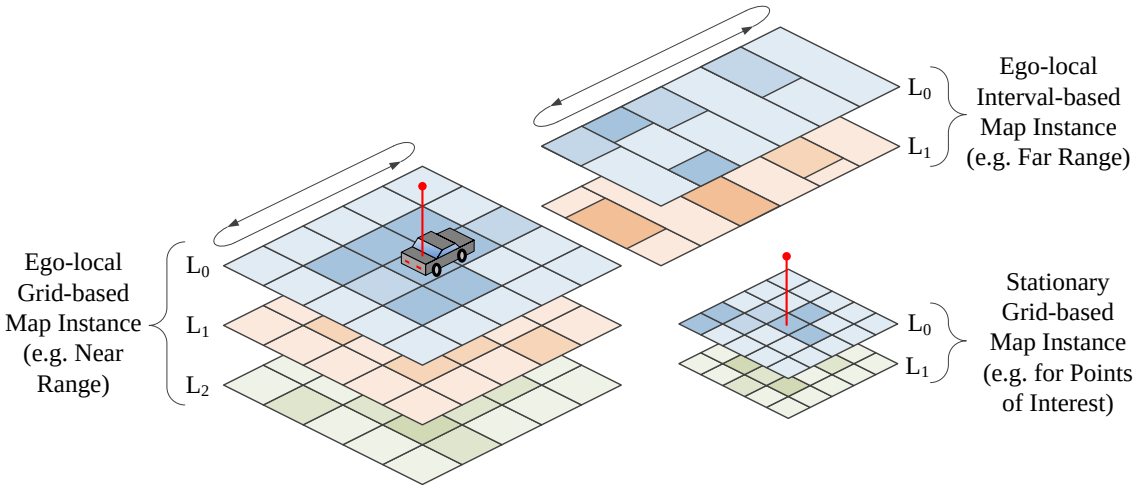


Figure 5.2.: Exemplary combination of several map-based environment representations.

Besides the previously introduced map contents about occupancy information and moving object motion, an existing implementation for the accumulation of perspectively mapped brightness transitions in camera images [Dü15] has been used for the evaluation of the developed concept.

The location of a map instance can be defined either in the global coordinate system, which results in a stationary environment representation, or in the USK, which yields an ego-local representation, as the reference system is carried along one's own vehicle. In the illustrated example, the high-resolution grid in the lower part of the figure is assumed to be stationary, while the remaining map instances are attached to the vehicle. Globally fixed maps are especially useful to represent detailed knowledge about stationary points of interest. On the other hand, ego-local maps allow for providing representations with constant preview areas and comparatively low memory requirements.

Overall, this concept offers several advantages for the application in the environment perception of a highly automated vehicle: Due to the combination of map instances with different representation types, the developed approaches can be deployed according to their identified strengths and weaknesses as well as the available computational resources. Furthermore, the aggregation of several ego-local and stationary maps allows to considerably improve the scalability of the entire map-based environment representation, which has been identified as an essential requirement earlier in this thesis. By considering several map layers within the map instances, the represented environment features can be estimated under consideration of the remaining map layers and hence mutually benefit.

The following sections first explain the combination of layers with different contents within a map instance. Afterwards, the aggregation of multiple map instances and the transfer of information between them will be discussed. Both major sections follow the general structure that was provided in the introduction, see figure 1.9.

5.3. Combinations of Map Layers

The motivation for estimating multiple map-based environment representations within a single module is to prevent the repeated execution of common tasks, e.g. buffering and interpolating ego motion data, and to improve the estimation of the represented environment features. A key aspect in improving the representations' estimation is to enable an easy and efficient access between different map contents. In this regard, the combination of several content layers within a single map module in the environment perception architecture offers two major advantages. First, the information between the layers can be exchanged without causing additional data transmission effort, which is an important aspect especially due to the memory demands of map-based representations. Second, the module-internal layer integration enables to ensure special constraints concerning the compensation and geometries of the layers, which can be used to simplify the comparability of the represented map cells.

The developed concept allows differing cell sizes in the map layers, but requires that the ratio between each two discretization sizes is a whole number. Furthermore, it specifies that all map layers are equally compensated and hence provide a consistent alignment at any point in time. Assuming an example configuration with two GM layers with large and small cell sizes (see figure 5.3 a), these rules guarantee that the area of any larger cell can always be congruently represented by a fixed number of small cells. Consequently, the probability value that corresponds to a large cell can be inferred by simply averaging all determined small cell probabilities, without using further interpolation calculations. These principles can be applied to both grid- and interval-based map representations. In case of the 2DIM these rules refer to map interval sizes, whereas in case of the GM the cell sizes are regulated. An illustration of possible map layer configurations can be found in figure 5.3.

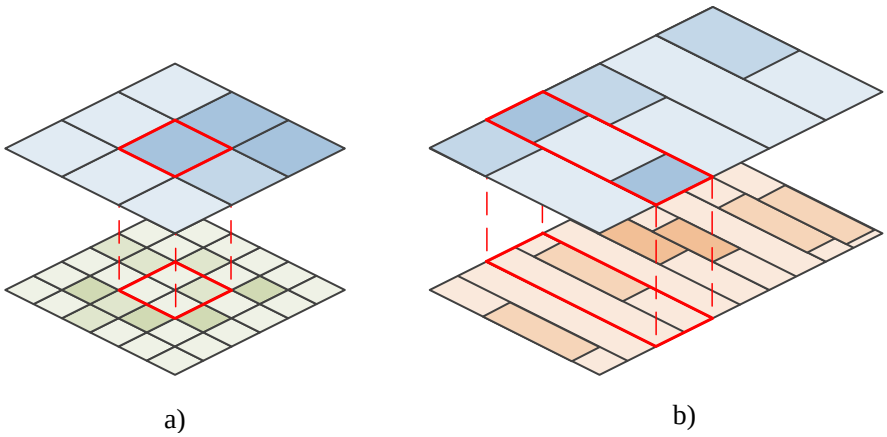


Figure 5.3.: Possible configurations of a) grid- and b) interval-based layer discretization sizes.

5.3.1. Development of Sensor Models and Update Mechanisms

In order to enable a mutual access between layers with different contents, the map instance has to be extended to include layer-specific coordinate system locations ${}^{USK}\mathcal{P}_{L_i}$. If the update of map layer i can be improved by considering the estimated environment feature in layer j, the first step is to infer the pose between the layers' coordinate systems:

$${}^{L_j}\mathcal{P}_{L_i} = {}^{L_j}\mathcal{P}_{USK} \cdot {}^{USK}\mathcal{P}_{L_i} \tag{5.1}$$

For reasons of efficiency, this static pose can be calculated before the map update takes place in each processing cycle. In case of a GM, this pose is used to transform the corner coordinates of a cell in layer i to layer j. Due to the previously described constraints, this transformation either yields a single cell in layer j or defines the corners of a rectangular cell grid, whose probability values can be averaged. In case of the 2DIM, the pose between the layers can be used to determine the corresponding map interval in layer j. The required interval cells have to be determined by comparing the continuous lateral cell border positions.

The so inferred existence probability of the environment feature $x^{(j)}$ can be taken into account for the estimation of existence probability of feature $x^{(i)}$. Formally, the additional feature consideration modifies the required probability density to:

$$p(x^{(i)}|\boldsymbol{z}_{1:k}) = \sum_{x^{(j)}} p(x^{(i)}|x^{(j)}, \boldsymbol{z}_{1:k}) \cdot p(x^{(j)}|\boldsymbol{z}_{1:k}) \tag{5.2}$$

$$= p(x^{(i)}|x^{(j)}, \boldsymbol{z}_{1:k}) \cdot p(x^{(j)}|\boldsymbol{z}_{1:k}) + p(x^{(i)}|\neg x^{(j)}, \boldsymbol{z}_{1:k}) \cdot (1 - p(x^{(j)}|\boldsymbol{z}_{1:k})) \tag{5.3}$$

Whereas the term $p(x^{(j)}|\boldsymbol{z}_{1:k})$ is given by the represented probability values in layer j, the remaining terms would require further recursive estimation processes, e.g. by using the inverse sensor model equations (A.56) and (A.57). The additional estimations of both values would increase the original computation time and memory demand by roughly a factor of three. An application of the efficient log-odds estimation scheme would further require repeated back-transformations of the resulting log odds ratios in order to determine the finally desired probability $p(x^{(i)}|\boldsymbol{z}_{1:k})$ according to equation (5.3). Especially in case of large grid-based representations, this additional effort heavily impacts the real-time capability of the entire approach. Therefore, the incorporation of additional environment features was limited to simplistic rule-based adaptions of the conventional inverse sensor-model probabilities in the experiments for this section.

5.3.2. Development of Mechanisms for Spatio-Temporal Consistency

The consideration of multiple map layers further requires to adapt the compensation mechanisms of ego local maps. In order to assure an identical alignment of the compared cells, all map layers have to be simultaneously compensated before any update takes place. The same principle applies to the compensation of the curved 2DIM due to a changed

reference path. The entire sequence of processing steps within a map instance is illustrated in figure 5.4. In contrast to a separate mapping of different environment features, the repeated compensation of all map layers increases the combined computational effort of all map-based representations. However, due to the efficiency of the introduced grid- and interval-based map compensation mechanisms, the advantages of the simplified access between the map layers outweigh this additional compensation effort.

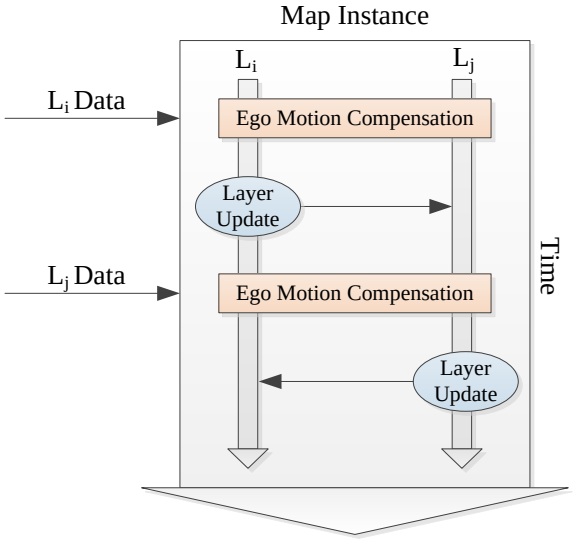
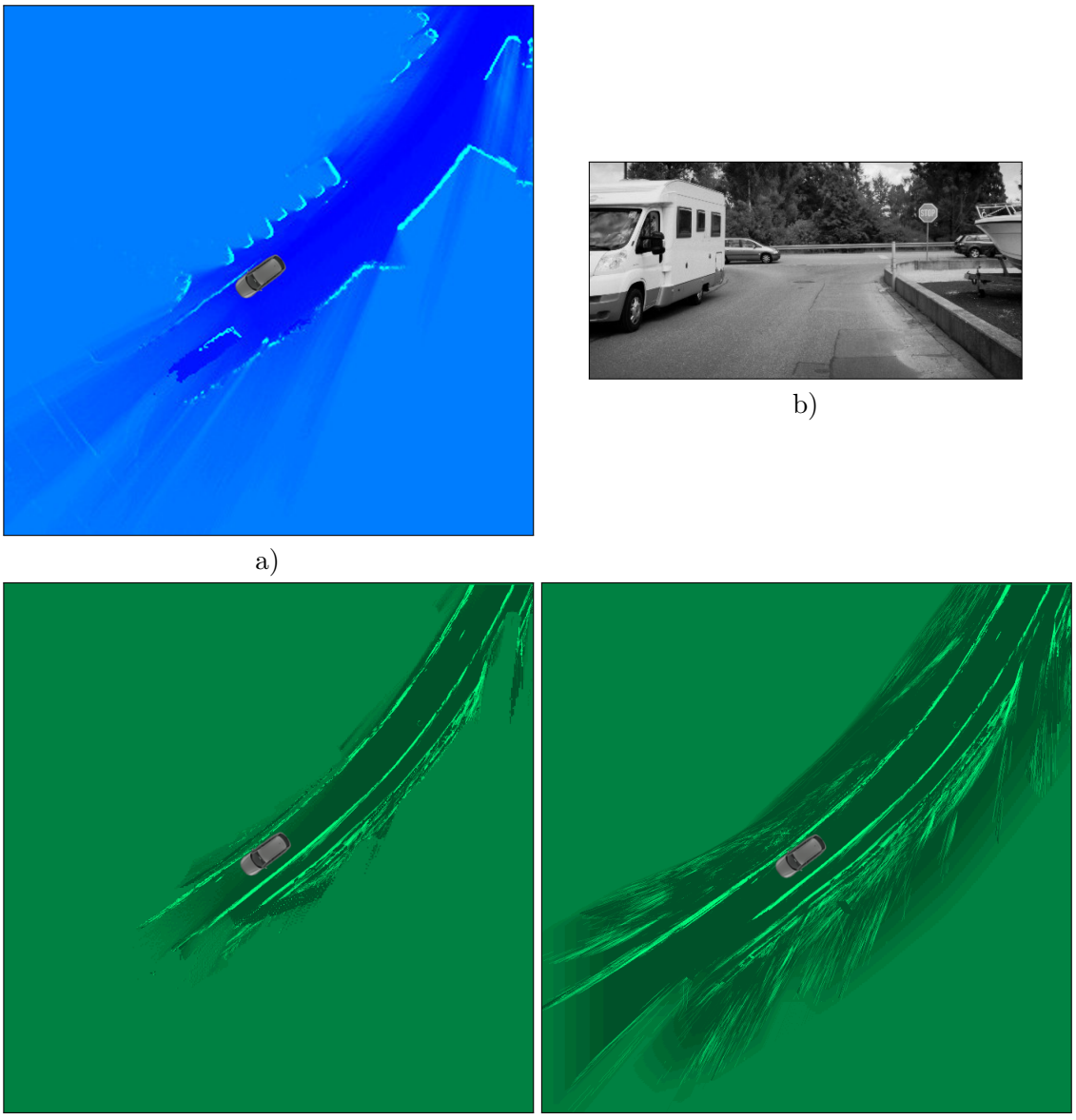


Figure 5.4.: Compensation procedure in multi-layer map-based environment representation.

5.3.3. Exemplary Application

Possible improvements of simultaneously estimating several map content layers can be found in an example illustrated in figure 5.5. In the first layer of this map instance, conventional laser-based occupancy states were mapped. The second layer was used to accumulate *unspecific brightness transitions* [112], which were detected in camera images and projected onto the map ground plane. This information can be used to detect the borders of accessible road areas, as e.g. the boundaries of road surfaces and ground vegetation typically result in camera image brightness transitions [Dü15]. However, these transitions can also be detected on elevated obstacles, e.g. preceding vehicles or roadside structures. In this case, the information of the occupancy layer can be used to improve the estimation of the desired environment feature: As elevated objects are represented by occupied areas in the first map layer and the required road surface boundaries are located on the assumed ground plane, the update of the brightness transition layer can be reduced to free areas. Figure 5.5 compares the resulting representations c) with and d) without additionally considering occupancy layer information. The illustrated maps demonstrate



a) b) c) d)

Figure 5.5.: Comparison of a) occupancy representation and brightness transition maps c) with and d) without using occupancy information. b) shows the corresponding video screenshot.
Similar to the occupancy map, bright green colors in the brightness transition map indicate high existence probabilities of relevant transitions.

that the combination of both layers significantly reduces the number of unfeasible brightness transitions on the side of the road. Consequently, the results of the subsequent road infrastructure detection can be improved.

5.4. Combinations of Different Representations

Having identified the internal structure of grid- and interval-based map instances, the following sections propose an approach for the aggregation and interaction of several map instances, which is another major contribution of this thesis. As ADAS functions typically have different perception requirements within certain areas in the vehicle's environment, a combination of multiple map instances can help to decrease the resulting computational effort while still providing sufficiently accurate data. By this strategy, one of the main disadvantages of solely grid-based environment representations can be eliminated. In order to achieve this goal, the first task is to extend both grid- and interval-based representations to support uniformly configurable map locations within the vehicle's environment. The quality of a combined representation can then be improved by transferring information about common environment features between neighboring map instances. For that purpose, *virtual sensor models*, which transform environment features between different kinds of representations, have to be developed. Finally, the previously introduced compensation mechanisms have to be enhanced in order to assure the spatio-temporal consistency of the combined representations while taking into account possible correlation issues.

5.4.1. Flexible Positional Relationships of Representations

In the previous course of this thesis (see section 3.1), existing occupancy grid approaches have been reviewed concerning their suitability for ego-local representations. The ego compensation mechanism presented by Bouzouraa and Weiss [16, 182] turned out to be particularly efficient for this purpose. Their approach is based on locating the vehicle in the map's middle cell and compensating translational movements by shifting a two-dimensional circular buffer, whereas rotational changes are used to adapt the alignment of the vehicle to the data structure. In the following, this mechanism will be extended and common positioning principles for both available representations will be developed, which is an essential prerequisite for the realization of flexible combined map-based representations.

In order to provide a consistent formal description, the location and temporal behavior of a grid map will also be described by using the pose ${}^{Map}\mathcal{P}_{USK}$. The movement of the vehicle between two time steps k and $k+1$, which corresponds to the modification of the USK, changes this pose in the following way:

$$ {}^{Map}\mathcal{P}_{(USK_{k+1})} = {}^{Map}\mathcal{P}_{(USK_k)} \cdot {}^{(USK_k)}\mathcal{P}_{(USK_{k+1})} \tag{5.4} $$

Depending on the desired behavior of the map, there are two possible ways to deal with the updated pose. In case of a stationary map, this pose and the corresponding map data structure do not need to be further modified. In this way, the map can be located at a fixed global position, to which the ego vehicle moves relatively.

In case of an ego-local map, the data structure has to be compensated until the resulting pose ${}^{(Map_{k+1})}\mathcal{P}_{(USK_{k+1})}$ satisfies the predefined criteria. Assuming the compensation principle from [16, 182], this criterion corresponds to the requirement that the vehicle has to be located at the center of the map. Consequently, the map's circular buffer is shifted until the pose's translational component conforms to the middle cell. This approach can also be extended to enable larger or smaller preview widths than half the map size. In this case, the ego compensation mechanism consists of two steps, as also illustrated in figure 5.6:

- In the first step (see figure 5.6 b), the desired ego vehicle cell has to be determined under consideration of the required preview area by virtually rotating the map center around the vehicle.
- In the second step (see figure 5.6 c), the map's circular buffer is shifted until the desired center cell reaches the actual center of the map.

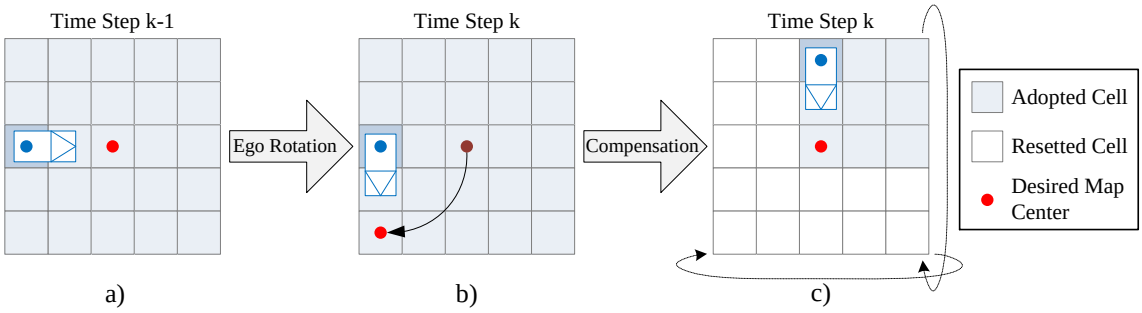


Figure 5.6.: Compensation of ego-local grid maps with increased preview area.

Due to the fixed coupling of the map's orientation to the ego vehicle, the developed ego compensation mechanism for the straight 2DIM is slightly more complex. The rotational component of the pose between map and USK is eliminated in every update cycle by rotating back all represented cells. The translational changes of the pose are compensated by laterally shifting the map cells and longitudinally shifting the map's circular interval buffer. With this technique, ego-local 2DIMs can also be located with arbitrary translational offsets to the ego vehicle. Similar to grid-based maps, this principle can easily be extended to enable stationary interval-based representations. In this case, the pose between map and USK is updated in every processing cycle without changing the map contents.

By using the introduced methods, several ego-local and stationary map instances of different types can be aggregated. Figure 5.7 shows an example of grid- and straight interval-

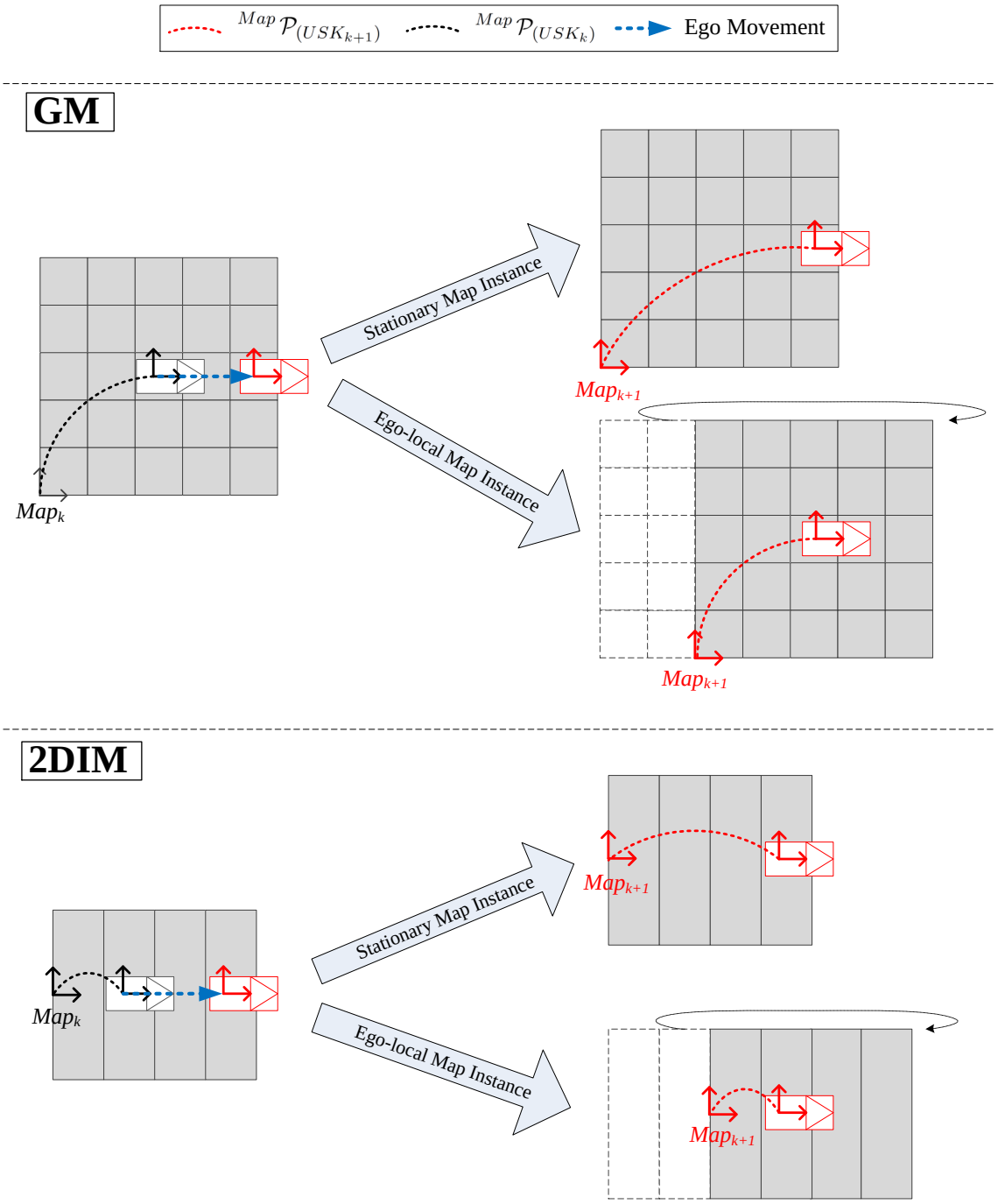


Figure 5.7.: Illustration of coordinate systems and poses for stationary and ego-local GMs and 2DIMs.

based map instances including their coordinate systems respectively poses and illustrates the different behavior of ego-local and stationary map instances in two consecutive time steps. Curved 2DIMs will not be considered for the combination of map instances in this chapter, as their location and geometrical configuration are defined by the reference path estimation module and hence cannot be modified. However, the derived principles can easily be adapted to incorporate curved 2DIMs.

5.4.2. Development of Sensor Models and Update Mechanisms

During the operation of aggregated map-based representations, several contained map instances may temporarily or permanently overlap. In these situations, overlapping map instances can be used as a virtual sensor in addition to the already available sensors. If this transfer correctly considers the requirements for fusing estimated environment states and the characteristics of the contained representations, the quality of the combined representation can be further improved. Therefore, the following section deals with the question of how to exchange map contents while avoiding the fusion of correlated state estimates. Afterwards, the required algorithms for transforming contents from different representations will be developed. While a transfer between two transformed regular grids can be accomplished by well-known image transformation techniques (e.g. bilinear interpolation), the transformation between grid- and interval-based contents requires to develop new algorithms, which correctly consider the represented uncertainties and satisfy the introduced computational requirements. The examples and illustrations in this section are restricted to the combination of several occupancy representations, as both grid- and interval-based map instances for this application were developed and described in the previous course of this thesis. However, the derived principles can easily be adapted to other map contents, e.g. motion cells or video-based brightness transitions.

Exchanging Map Contents

Each individual map instance accumulates the knowledge about the contained environment features by using all available sensor measurements up to the current time step. Similar to the previously discussed track-to-track fusion applications (see section 4.3.3), the cross-correlation of the estimated map states has to be considered when exchanging information between map instances. The impact of neglecting this correlation is best explained by using the example illustrated in figure 5.8. This figure shows a grid cell that incorporates the state estimate of an overlapping interval cell for its own estimation of the same environment feature in every cycle of the system. As both cells use the same sensor measurements, the state estimates at time step k share a common measurement history and are thus correlated. As the illustrated measurement histories in the lower part of the figure show,

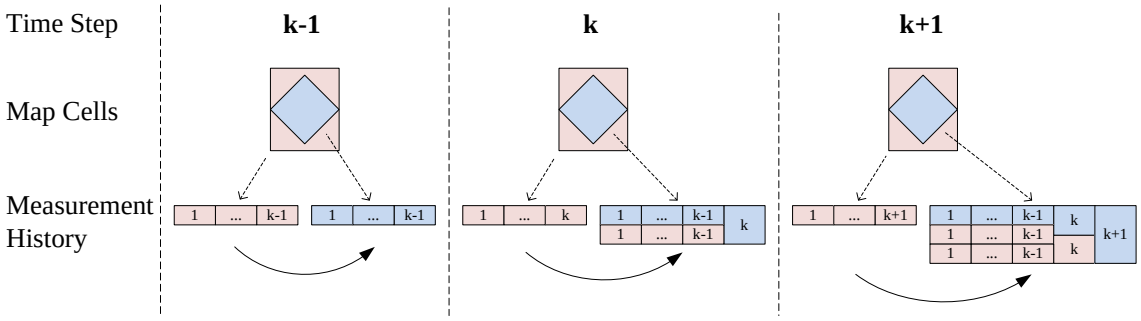


Figure 5.8.: Impacts of neglecting common measurement histories of exchanged map contents.

exchanging the correlated estimates leads to a systematic overvaluation of previous sensor measurements in the resulting grid cell state. Due to this imbalance, the final resulting state estimate deteriorates by additionally incorporating information from overlapping cell states.

Nevertheless, the proposed concept of aggregating multiple ego-local and global map instances offers several opportunities to exchange information between map cells without common measurement histories. If we consider an ego-local GM instance as illustrated in figure 5.9, the movement of the ego vehicle causes a repeated reinitialization of those cells, which are appended in front of the data structure. As the recursive state estimation equation A.68 suggests, these cells are initialized with the a priori existence probability of the considered environment feature, which is typically chosen to be 0.5. In this situation, the state of an overlapping map cell can be incorporated without causing the problem of fusing estimates with common measurement histories. As can be seen in figure 5.9, the overlapping cells' content from the previous system cycle can serve as previous knowledge for all new cells. If this information exchange is limited to the initialization of a new cell, it can be guaranteed that all sensor measurements are only incorporated once.

In the same way, this principle can also be used to generate previous knowledge for newly appended intervals in ego-local 2DIMs or newly created stationary map instances. More

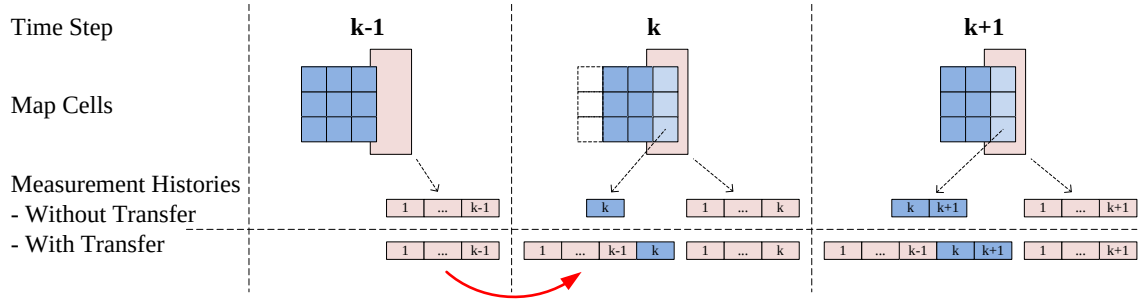


Figure 5.9.: Impacts of using overlapping map content a priori knowledge for newly initialized map cells.

generally, this method can always be applied when a map instance requires a priori knowledge in areas where another map instance exists. The beneficial effects of replacing the a priori probability by previous knowledge from overlapping map instances will be evaluated in section 5.4.4.

Virtual Interval Sensor Model for Grid-Based Maps

The task of the virtual interval sensor model is to transform the content of a 2DIM interval into a number of newly initialized square cells of a GM. As the example in figure 5.10 illustrates, the eligible cells can be located either completely or partially within the interval borders. In order to avoid complex intersection calculations, the developed approach unambiguously assigns the grid cells to single 2DIM intervals depending on their center position. Due to the comparably small grid cell sizes, this approach turned out to be sufficiently accurate. The center and corner coordinates of each so determined grid cell can be transformed into a 2DIM interval coordinate system by using the following poses:

$$^{I^{(n)}}\boldsymbol{p} = {}^{I^{(n)}}\mathcal{P}_{USK} \cdot {}^{USK}\mathcal{P}_{Grid} \cdot {}^{Grid}\boldsymbol{p} \tag{5.5}$$

The entire computational effort resulting from these coordinate transformations can be reduced by making use of the constant distances between the nodes of a rotated regular grid. Due to this characteristic, it is sufficient to transform the coordinates of a single cell and to determine the translation between the cells in the target coordinate system. The coordinates of the remaining cells can then be inferred by combining the transformed cell and the resulting transformation vector, as also illustrated in the left part of figure 5.10.

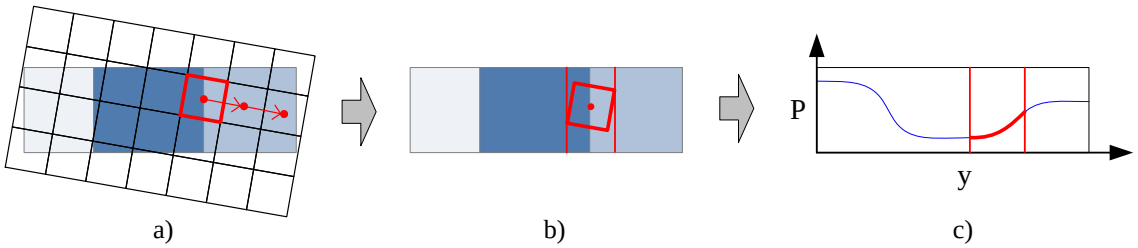


Figure 5.10.: Virtual sensor model for transferring information from interval- to grid-based representations.

Having identified the coordinates of the cell corners in a 2DIM interval coordinate system, a minimum and maximum lateral position of the cell can be determined. Based on these borders, the required probability value can be deduced from the represented interval cells. In doing so, the different representations of occupancy state uncertainties have to be considered: While the 2DIM cells explicitly describe an uncertainty about their lateral borders by quantifying a variance, the GM implicitly represents this information by gradations of the cell probabilities. Therefore, the previously introduced probability profile along the

interval's y-axis (see section 3.2.2, equation (3.12)), which incorporates both occupancy probabilities and border variances, should be used for this purpose. An exact calculation of the average probability value in the lateral range of the grid cell would require to determine the integral of the probability profile, as illustrated in the right part of figure 5.10. In view of the complexity of the probability profile equation (3.12), the developed implementation uses a simplification to estimate the required probability. The evaluation of the probability profile is restricted to the lateral start and end position, the required probability results from averaging both values. Furthermore, only directly adjoining border variances are considered for the calculation of the probability profile equation (3.12). Due to the low influence of more distant borders and the overall limited grid cell sizes, this approximation provides sufficiently accurate results.

Virtual Grid Sensor Model for Interval-based Maps

In contrast to the previous case, the target data structure does not provide predefined cell sizes and equidistant sampling locations when transferring information from grid-based representations to a newly initialized 2DIM interval. The introduction of an artificial lateral sampling rate would eliminate the major advantage of the continuous cell border positions in interval-based representations. Consequently, the task of the virtual grid sensor model is to determine both the existence probabilities and lateral extents of interval cells from overlapping grid cells. In addition, the variances of the detected borders have to be specified by analyzing the gradation of the grid cell probabilities.

The developed transfer algorithm is based on a repeated application of *Bresenham's line algorithm*, a well-known line rasterization technique from the area of computer graphics, which is e.g. in [74] described in detail. This approach is particularly suitable for an

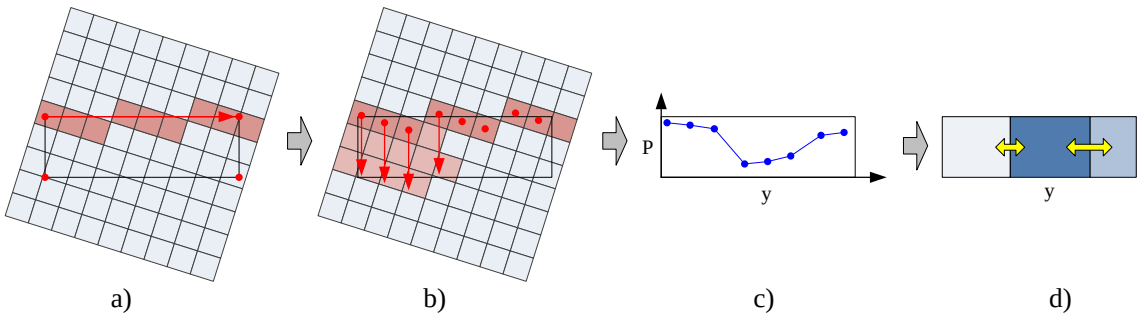


Figure 5.11.: Virtual sensor model for transferring information from grid- to interval-based representations.

efficient analysis of grid-based environment representations, as it is restricted to integer additions, subtractions and bit shift operations. In order to extract the grid map content according to the alignment of a 2DIM interval, the first step of the developed algorithm is to determine a Bresenham line between the upper corners of the interval, as also visualized

in figure 5.11 a). Similar to the previous case, the coordinates of these corners can be transformed by combining the USK poses of both maps. The resulting cells can then be used to analyze the grid cell probabilities within a certain range of the interval's y-axis. For that purpose, these cells serve as start coordinate for another Bresenham line parallel to the intervals x-axis, which is shown in figure 5.11 part b). As this procedure might leave gaps between the parallel vertical Bresenham lines, the performance of the algorithm can be optimized by using the *extended Bresenham algorithm*, which has been proposed in [16]. In this way, it can be guaranteed that every cell within the interval borders is considered for the transfer of information between both representations.

Due to the introduced semantics of the 2DIM occupancy interval cells, which requires that a cell represents the most critical occupancy probability within the enclosed area, the maximum probability values along the vertical Bresenham lines have to be determined for the further processing. Thus, the final result of the rasterization process is a polygonal chain describing the probability profile along the interval's y-axis, as illustrated in part c) of figure 5.11. The lateral distances between the chain's nodes depend on the discretization of the grid and the alignment between both representations. In case of equally aligned grid and interval geometries, the resulting distances are constant and correspond to the grid's cell sizes.

In the last step, the resulting profile has to be converted into multiple interval cells including occupancy probabilities, border positions and variances. In doing so, it is important to limit the number of emerging interval cells, as it significantly influences the computational and memory requirements of the interval-based representation. Therefore, the determined

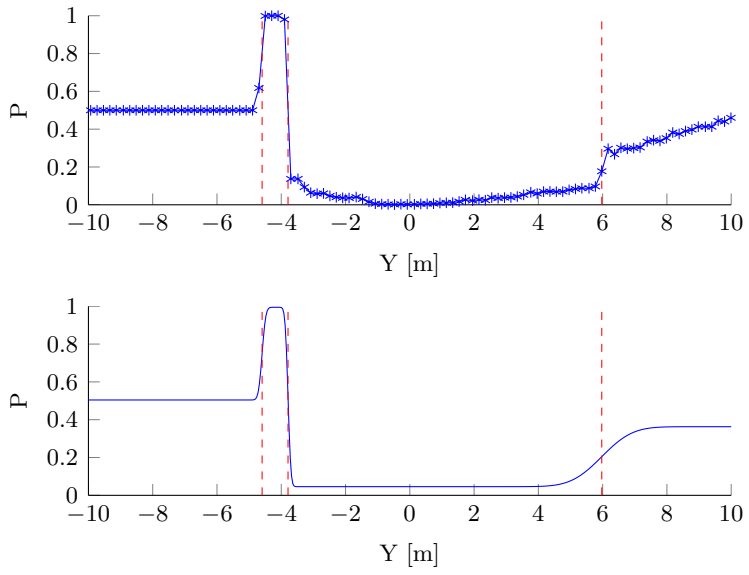


Figure 5.12.: Example of an extracted probability profile polygonal line and the determined cell-based probability profile.

probability value transitions should be quantified by cell border variances to the greatest possible extent. The developed conversion strategy is twofold, figure 5.12 illustrates an example of an extracted polygonal chain and the determined interval cell probability profile. First, the transitions between homogeneous probability value ranges are identified by approximating the polygonal chain's first derivative by the central difference quotient. The extreme values of the resulting derivative indicate the 2DIM interval cell borders, which are illustrated by vertical red dotted lines in the example, and can e.g. be determined by a simple threshold comparison. The determination of the transitions' variances in the second phase corresponds to an optimization problem, whose target function is given by the sum of differences between the extracted probability values and the finally resulting probability profile function. Once more, the illustrated example emphasizes the advantages of applying the 2DIM for this purpose: Due to the representation of coherent areas and their transitions, the occupancy states can be described significantly more compact than in the polygonal chain resulting from the grid-based map.

5.4.3. Development of Mechanisms for Spatio-Temporal Consistency

The newly created possibility to transfer contents between several map instances raises the question of how to integrate this step into the existing update procedures of the available occupancy maps. Furthermore, it has to be investigated how the spatio-temporal consistency of multiple map instances within an aggregated environment representation can be assured without significantly increasing the computational requirements, especially in comparison to an individual processing of the involved map instances.

Figure 5.13 shows a generic update cycle for occupancy maps that has been extended by the optional transfer of overlapping map contents. For both interval- and grid-based rep-

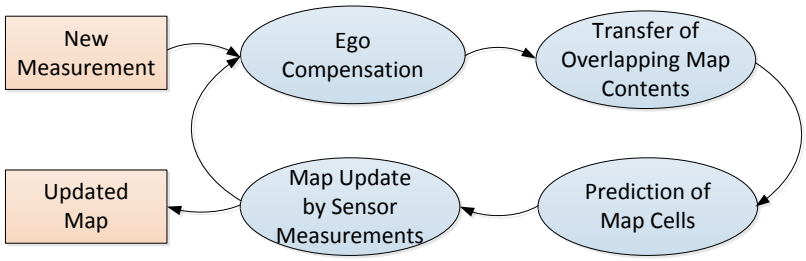


Figure 5.13.: Extended recursive update cycle of map instance in a combined map-based environment representation.

resentations, this cycle begins with the compensation of the ego motion. Strictly speaking, this step also applies to stationary map instances, where the pose between the map and the vehicle has to be adapted according to the previously derived equations. During the compensation, several cells or intervals need to be reinitialized as the map's circular buffer is shifted. For each of these cells, it can then be investigated, whether the required a priori

probability can be improved by the content of an overlapping map instance. Whenever the involved map instances share a common moving object list, the transfer of map contents may also include the information of the cells' dynamic states. For this reason, the prediction of the map cell states (see section 3.4.3) should be performed after the transfer took place, in order to create the possibility to additionally consider the transferred dynamic cells. Finally, the entire representation is updated with the new sensor measurements and can be used for display or extraction purposes.

Still, the proposed update procedure does not answer the question of how to deal with multiple map instances within an aggregated representation. As derived in the previous section, the transfer of overlapping contents for the improvement of a priori probabilities requires at least two map instances:

- One map instance, whose ego compensation has led to newly initialized cells that have not been updated in the current system cycle yet
- At least one overlapping map instance that has not been updated in the current cycle so that the accumulated knowledge from the last time step can be transferred

This formulation already indicates that the processing order of the individual map instances has a significant impact on the possibilities to exchange map contents. Figure 5.14 shows an example of four aggregated map instances including their positions at two consecutive time steps. If we consider $M^{(a)}$ and $M^{(b)}$, the map content can only be exchanged after the compensation of $M^{(a)}$ has been finished and before the update of $M^{(a)}$ and the compensation of $M^{(b)}$ takes place. The same applies to the improvement of the a priori knowledge in map instance $M^{(b)}$ by using the accumulated contents from $M^{(c)}$ and $M^{(d)}$. A simple approach to guarantee the availability of non-updated overlapping map information would be to buffer all represented contents before the map instance update cycle is processed. However, the illustrated example already indicates that this additional effort can be avoided by conveniently choosing the processing order of the individual map instances.

In order to autonomously derive this processing order, the expansion areas of all participating ego-local map instances need to be determined without executing the actual ego compensation algorithms. In figure 5.14, the predicted expansion areas are framed by solid red lines. By intersecting these areas with all overlapping representations, a binary relation between maps with transfer possibilities can be established. Based on this relation, we can build a *directed graph*, whose nodes represent the aggregated map instances, as illustrated in the lower part of figure 5.14. Whenever the resulting graph does not contain directed cycles, a topological ordering of the nodes and hence a compensation order of the map instances can be directly inferred. The determination of graph cycles and the topological ordering can be solved in linear time by using well-known graph theory algorithms, which are e.g. described in [157]. In case this analysis reveals the existence of a directed cycle, the exchange of map contents requires to buffer the content of at least one participating map instance.

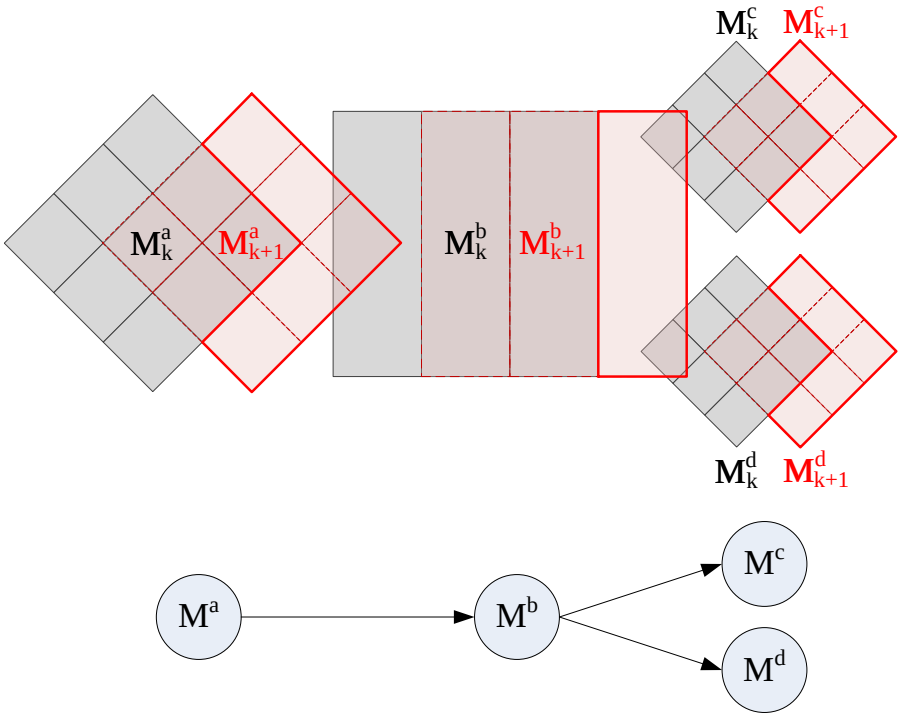


Figure 5.14.: Influence of the map compensation order onto the possibility to exchange map contents.

Having determined a final processing order, each substep of the update cycle from figure 5.13 has to be processed for all map instances until the subsequent steps takes place. Only in this way, it can be assured that overlapping map instances contain the required information for improving the a priori knowledge of an ego-local representation.

5.4.4. Evaluation

Similar to the evaluation of the newly developed interval-based representation in chapter 3, the beneficial effects of applying an aggregated map-based representation in ADASs need to be analyzed and quantified. Using the example of occupancy information, this section will systematically compare selected combined representations to conventional grid- and interval-based representations. Besides the previously identified goals to evaluate the quality, resource requirements and the interpretability of the compared maps, the presented experiments will investigate the following questions:

- Does a combined map-based representation equally satisfy the requirements of an ADAS function as conventional representations, while simultaneously reducing the resource requirements?

- What is the impact of transferring contents between aggregated map instances, on quality, and on the resource requirements of the entire representation?

In order to determine the quality of combined map-based representations, the previously proposed map-based evaluation scheme will be reused. As already introduced in figure 3.21, this approach consists of three sub-steps: the reference map creation, the establishment of comparability and the calculation of metric values. Similar to section 3.5, a high-resolution occupancy grid, obtained by measurements of the reference system, will be used as an ideal map for the quality quantification. In contrast to the experiments in chapter 3, the compared probability values will be extracted with a beam-based evaluation feature generation technique (see figure 3.22), where the beams have an angular distance of 1° and sample the maps at a distance of 0.2 m. By this strategy, the calculation of the required probability values in overlapping areas of the aggregated maps simplifies. The determined probability differences between the reference and evaluated map will then be rated by the previously introduced metrics *map score* and *weighted sum of squared errors*.

Experiment 1: Evaluation with Ideal Sensor Data

According to the approach from chapter 3, the first part of the evaluation analyzes the limitations of combined grid- and interval-based occupancy representations independent from the effects of real sensor measurements. The results in this section were obtained by using ideal reference measurements in the occupancy test scenarios S1a and S1b (see appendix D.1).

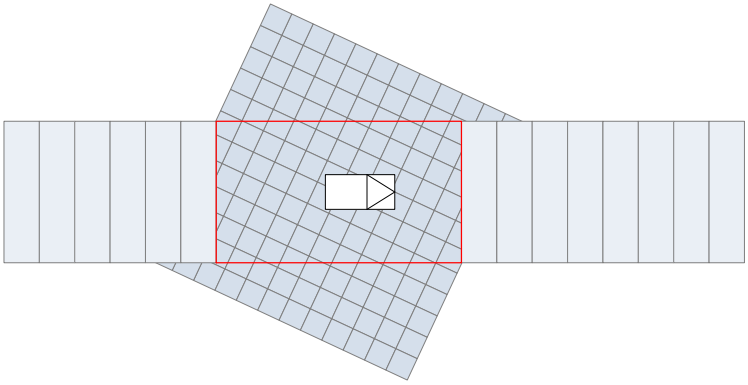


Figure 5.15.: Configuration of evaluated combined occupancy maps. The red box illustrates the guaranteed coverage of the grid-based representation.

For the evaluations of the combined occupancy maps, we will assume a required detection area of 20 m width from 50 m behind to 70 m in front of the vehicle. In the following comparisons, this area will be covered by a purely grid-based representation, a purely interval-based representation and three different combinations of both approaches, which will be denoted as A,B and C. The basic idea of the combined representations is illustrated

in figure 5.15: Whereas the direct environment of the vehicle is mapped by an ego-local high-resolution GM, the more distant areas behind and in front of the vehicle are covered by two separated 2DIMs. In this way, a high accuracy at close range, a large coverage of the entire map and reduced resource requirements can be achieved at the same time. The difference between the three options considered lies in the preview area of the central GM: Configuration A guarantees a grid-based representation within a square around the vehicle with an edge length of 20 m, whereas configuration B and C provide edge lengths of 30 respectively 40 m. Due to the rotation behavior of the grid-based representation, the dimensions of the entire GM have to be larger than the guaranteed preview areas, as also illustrated in figure 5.15. For detailed derivations of the required map dimensions the interested reader is referred to [Sei13].

Scenario	Map (Dimensions in [m])	Discr. [m]	MS	WSE	CT [ms]	MR [MByte]
S1a	GM	0.2	0.62	0.0060	0.8	2.5
	2DIM	1.0	0.60	0.019	0.5	0.05
	Comb. A (20×20 GM)	1.0 - 0.2 - 1.0	0.61	0.010	0.8	0.14
	Comb. B (30×30 GM)	1.0 - 0.2 - 1.0	0.61	0.0092	0.8	0.27
	Comb. C (40×40 GM)	1.0 - 0.2 - 1.0	0.61	0.0088	0.9	0.44
S1b	GM	0.2	0.67	0.0067	0.8	2.5
	2DIM	1.0	0.55	0.052	0.5	0.05
	Comb. A (20×20 GM)	1.0 - 0.2 - 1.0	0.57	0.039	0.8	0.16
	Comb. B (30×30 GM)	1.0 - 0.2 - 1.0	0.59	0.031	0.9	0.28
	Comb. C (40×40 GM)	1.0 - 0.2 - 1.0	0.61	0.026	0.9	0.45

Table 5.1.: Comparison of ideal combined occupancy map evaluation metrics in scenarios S1a and S1b.
(Discr. = Discretization size, cell sizes of GM / interval heights of 2DIM, MS = Averaged Map Score - Higher values indicate better map qualities, WSE = Averaged Weighted Squared Errors - Lower values indicate better map qualities, CT = Computation Time [1], MR = Memory Requirement)

Table 5.1 lists the averaged quality metrics and resource requirements obtained by using ideal reference measurements in the scenarios S1a and S2b. As expected, the ideal quality values of the combined map configurations range between those of the purely grid- and the purely interval-based representation, the larger the central GM, the better the inferred quality assessments. Except the standard 2DIM, all map configurations provided almost identical computation times for the mapping process, which indicates that the developed map transfer algorithms do not generate a considerable additional effort. Concerning

[1] All computation times in this thesis were measured on a laptop with Intel Core i7-3840QM CPU with 16 GB RAM.

the memory requirements, it appears that the incorporation of any interval-based map significantly reduces the resulting resource demands.

Experiment 2: Evaluation with Real Sensor Measurements

Similar to chapter 3.5, the aim of the second part of this evaluation is to quantify the quality of combined occupancy maps obtained by using real laser range measurements and the standard ego motion estimation module (see section 2.3.1). Unlike in the previous experiment with ideal measurements, the accumulation of the range measurements has an important role in this case, which especially allows to investigate the impact of the transfer of information between the aggregated representations.

Figure 5.16 illustrates the differences between two obtained combined representations with and without exchanging map contents. By replacing the a priori knowledge of the central GM by the contents of the front 2DIM, the accumulation of free and occupied areas in the vehicle's direct environment is significantly accelerated. For the application of a rear 2DIM as in the proposed configurations, the exchange of estimated map contents is even an compulsory requirement, as the map is outside the laser sensor's measurement range.

As the applied 2DIMs use a significantly higher discretization size than the central GM, the transfer from interval- to grid-based representation has been parametrized with different weighting factors w, which attenuate the extracted probabilities p_{2DIM} towards the a priori probability 0.5:

$$p_{GM} = 0.5 + (p_{2DIM} - 0.5) \cdot w \tag{5.6}$$

Figure 5.17 compares the map score developments of map configuration A with different map exchange parametrizations in scenario S1b. First of all, the measured values indicate a substantial improvement of the map qualities by the developed approach of transferring map contents between the combined representations. Overall, a direct adoption of the extracted probability values from the front 2DIM to the central GM provided the best quality results. The disadvantages of transferring attenuated probability values are hence more serious than the negative impacts of directly adopting coarsely discretized 2DIM contents. Consequently, this parametrization will be used for all combined representations in the further course of this evaluation section.

Table 5.2 summarizes the averaged metric values, computation times and maximum memory consumptions of all evaluated occupancy maps in scenarios S1a and S1b. The resulting quality values show a similar picture as in the previous experiment with ideal sensor data: The assessments of the combined map configurations are between those of the purely grid- and interval-based maps, but the quality gaps between the combined and grid-based representations decrease by the application of real sensor measurements. The measured computation times for the mapping process are overall very similar, except those of a purely interval-based representation. However, the final evaluation of the representations'

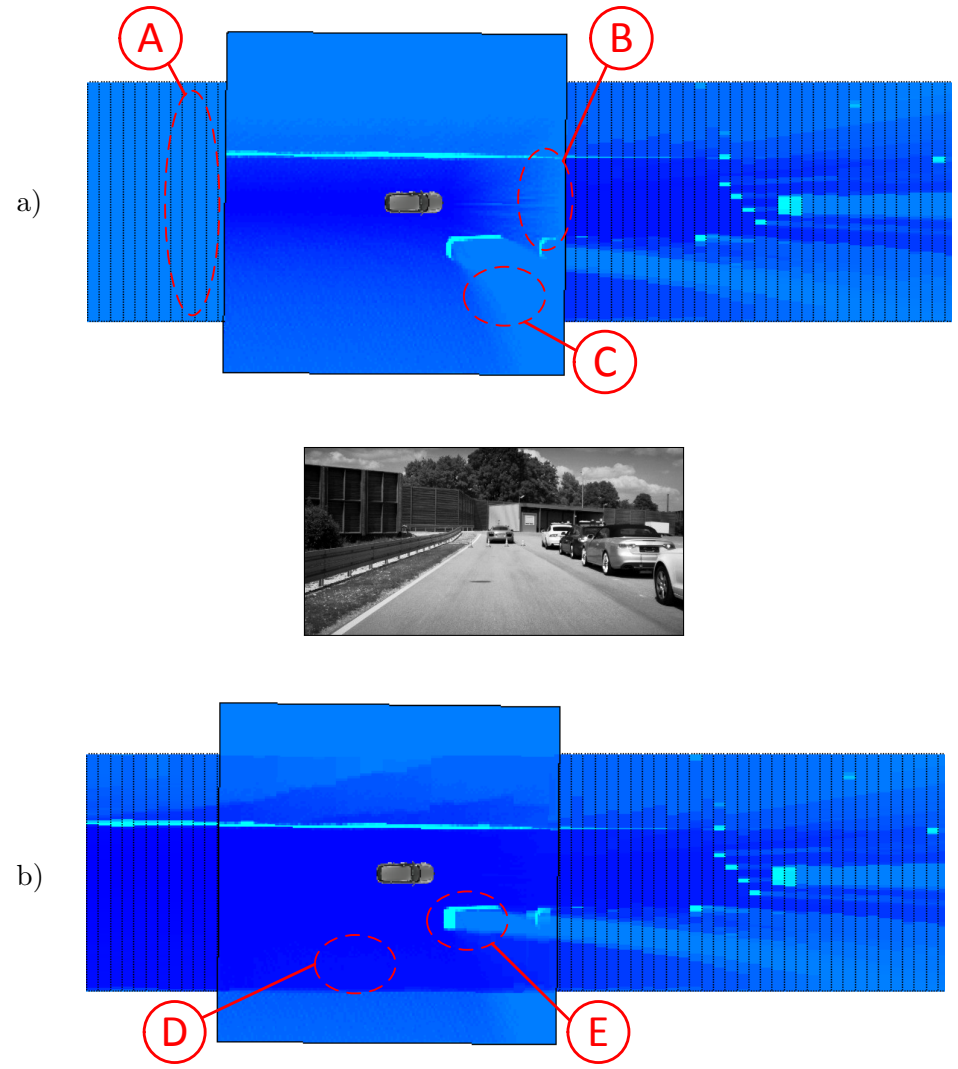


Figure 5.16.: Comparison of combined occupancy maps a) without and b) with exchanging map contents.

- A: Missing free space information due to the lack of transfer into the rear 2DIM.
- B: Slow accumulation of free space information due to the lack of transfer into the central GM.
- C: Missing free space information due to temporary occlusion.
- D: Improved accumulation of free space information due to the transfer of a priori knowledge.
- E: Coarse representation of the vehicle's contour due to the transfer of a priori knowledge.

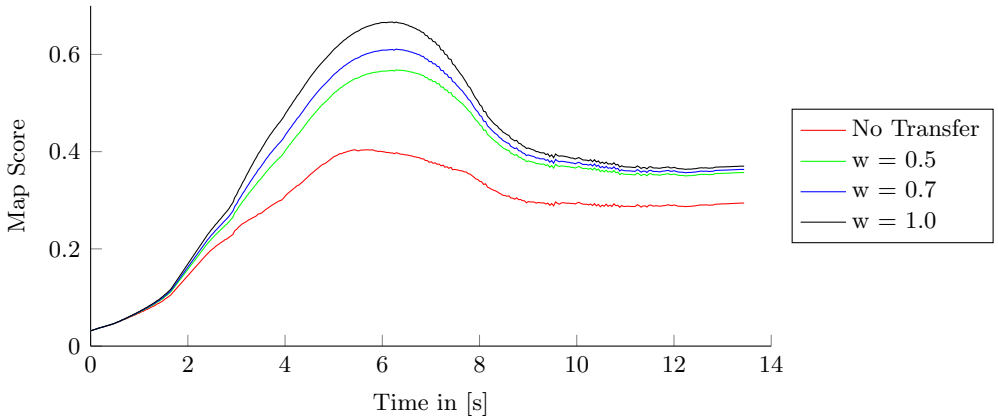


Figure 5.17.: Development of measured map scores in scenario S1b with different map exchange parameters.
w denotes the weighting factor, which attenuates the probabilities during the transfer from interval- to grid-based representations.

computational effort requires to also consider their interpretation effort, which will be analyzed in detail at the end of this section. Concerning the memory consumptions listed in table 5.2, the application of combined representations leads to a substantial improvement compared to GMs.

One of the main motivations for the development of combined map-based representations was to fulfill varying accuracy requirements in different subregions of the vehicle's environment. This can e.g. result from the requirements of several ADAS functions that utilize different preview areas and use a common environment perception. By using the beam-based probability value extraction scheme, the previously applied evaluation approach implicitly assigns a higher weighting factor to map errors in the direct vicinity of the vehicle, as the sampling accumulates at the center of the extraction. In order to simulate differing requirements of one or more ADAS functions, the resulting metric values can additionally be multiplied by a weighting mask that distinguishes between the errors in the close range, which is e.g. used for an accurate evasion trajectory evaluation, and those in far range, which can e.g. be required for a course road boundary detection. When applying this mask, combined representations achieve a similar quality evaluation as purely grid-based evaluations, although the size of the computationally intensive GM is decreased.

Concerning the measured computation times, it is again important to incorporate the extraction effort of the occupancy representations. If we consider the sole computation times of the mapping process in a file including a more unstructured urban environment, the combined map configurations even slightly increase the computational effort in comparison to a GM. However, the evaluation section in chapter 3 already indicated that a representation's interpretability can be substantially improved by applying the newly

Scenario	Map (Dimensions in [m])	Discr. [m]	MS	WSE	CT [ms]	MR [MByte]
S1a	GM	0.2	0.50	0.11	1.2	2.5
	2DIM	1.0	0.46	0.13	0.8	0.12
	Comb. A (20×20 GM)	1.0 - 0.2 - 1.0	0.49	0.12	1.2	0.33
	Comb. B (30×30 GM)	1.0 - 0.2 - 1.0	0.50	0.12	1.2	0.45
	Comb. C (40×40 GM)	1.0 - 0.2 - 1.0	0.50	0.11	1.3	0.61
S1b	GM	0.2	0.51	0.14	1.2	2.5
	2DIM	1.0	0.37	0.22	0.8	0.11
	Comb. A (20×20 GM)	1.0 - 0.2 - 1.0	0.46	0.16	1.2	0.32
	Comb. B (30×30 GM)	1.0 - 0.2 - 1.0	0.48	0.15	1.2	0.42
	Comb. C (40×40 GM)	1.0 - 0.2 - 1.0	0.49	0.15	1.3	0.57

Table 5.2.: Comparison of combined occupancy map evaluation metrics in scenarios S1a and S1b.

developed interval-based approach. In order to quantify and compare the data structures' interpretabilities, the previously introduced simple obstacle extraction (see section 3.5.5) has been applied to all tested map configurations, figure 5.18 summarizes the resulting measurements. In the overall view, the computation times of the combined representations range between those of the purely grid- and interval-based representations. The additional consideration of the extraction effort clearly shows the negative impact of GMs, even when combined with other 2DIMs. In comparison to a pure GM, the resource requirements can

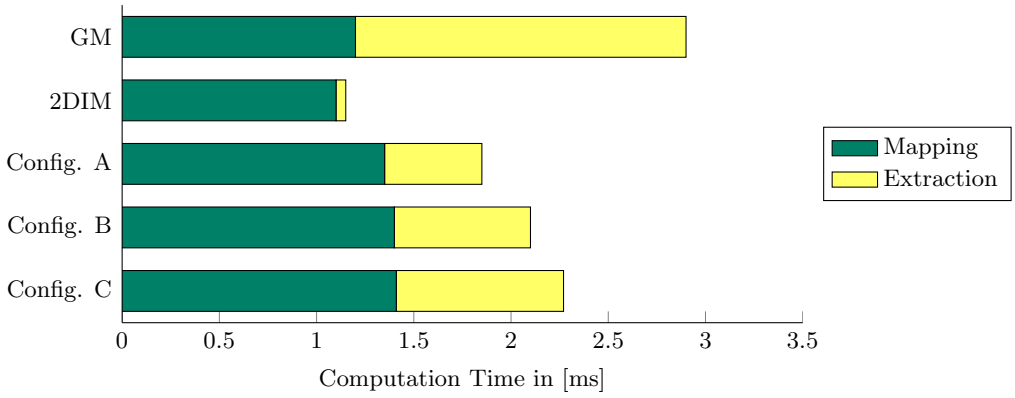


Figure 5.18.: Comparison of mapping and extraction computation times of combined occupancy maps.

be reduced by applying a combined map configuration, at which the size of the configurations' central GM can be used to control both the resulting accuracies and computational effort.

Summary

The potential of combining different map-based environment representations has been examined by evaluating and comparing three exemplary occupancy map configurations against purely grid- and interval-based representations. For that purpose, the previously developed map-based evaluation scheme and the occupancy reference scenarios have been reused. As the tested map configurations consist of a close range GM and two far range 2DIMs, the measured map-qualities range between those of a purely grid- and a purely interval-based representation, for both using ideal and real sensor measurements. If we simulate differing accuracy requirements of one or more ADAS functions by an additional weighting mask, a combined map configuration is able to provide similar map qualities as a GM, while simultaneously reducing the memory requirements and computation times, especially when incorporating the extraction process. The developed mechanisms to transfer accumulated environment features between grid- and interval-based representations turned out to have a beneficial impact on the resulting map qualities and had little effect on the measured computation times.

Overall, the proposed concept of combining grid- and interval-based representations also offers the possibility to dynamically instantiate high resolution maps in dedicated regions of interest around the vehicle, which has not been evaluated and compared against existing approaches in-depth. Nevertheless, the static combination of close range GMs and far range 2DIMs already enables a substantially improved scalability in comparison to state-of-the-art grid-based representations.

6. Conclusion and Future Works

6.1. Conclusion

A further increase of current ADASs' robustness and degree of automation requires an environment perception which provides information about all relevant environment features in sufficient accuracy while simultaneously satisfying the requirements concerning the computational effort and the interpretability of the obtained results. Due to their weak model assumptions, map-based environment representations are especially suitable for describing complex and unstructured environment features at a high level of detail. However, state-of-the-art environment perception approaches mainly use map-based representations for the description of occupancy information and are restricted to the application of grid-based representations, which suffer from substantial disadvantages with respect to their computational effort, scalability and interpretability. Concerning these scientific gaps, this thesis provides several contributions.

In chapter 3, two different versions of a novel and generic environment representation were proposed: the *straight* and *curved 2D Interval Map*. It was shown, how these representations can be used to accumulate *occupancy information* in the vehicle's surroundings and how the *uncertainties* of the applied laser sensor can be considered in the applied EKF-based estimation scheme. After that, different *ego compensation mechanisms*, which enable the instantiation of ego-local map instances, have been developed. By enhancing the developed approach concerning the additional *consideration of moving objects*, a crucial requirement for the application of the interval-based occupancy map in highly dynamic traffic scenarios has been fulfilled. Finally, a new *map-based evaluation scheme*, which allows to quantifiably compare different representations of occupancy information, has been developed. In comparison to a reference GM implementation, the 2DIM achieved similar quality assessments, while the memory demands and the computational effort including the data structure's interpretation were significantly reduced.

The simplified map-based environment representation has then been used to detect a newly identified abstract environment feature: the *common motion behavior* of surrounding traffic participants. After identifying and defining the relevant *convoy track* information, a system architecture to infer the required information has been proposed and the accumulation of the object measurements in 2DIM *motion cells* has been described. For the development of the necessary sensor models, characteristic laser and *radar measurement effects* have been analyzed in typical dense traffic occlusion scenarios. In order to correctly incorporate filtered object hypotheses in the 2DIM estimation process, a concept

of decorrelating object tracks by calculating *equivalent measurements* has been presented. To determine the *observability* of the estimated motion cell state variables, an occlusion calculation has been proposed. The chapter has been concluded with the development of a *map-based evaluation scheme*, which allows to compare the map qualities when using different algorithm parametrizations or switching between curved and straight mapping approaches. Moreover, the beneficial impacts of using convoy track information during the behavior generation of highly automated vehicles have been outlined.

The developments of chapters 3 and 4 raise the question of how to combine different map-based environment representations with varying contents in a common environment perception architecture. For that purpose, chapter 5 proposes a concept that consists of two key ideas. Concerning the combination of different contents, a *layer concept* was proposed, which enables to realize *cooperative sensor models* that simultaneously use the information of several map layers to improve the estimation of represented environment features. Concerning the aggregation of representations, a concept to *flexibly position* and deploy multiple representations in the vehicle's surroundings has been presented. For that purpose, *virtual sensor models*, which transfer a priori knowledge between grid- and interval-based maps under consideration of the represented uncertainties, have been developed. The conducted quality evaluations showed that a combination of multiple representations is able to specifically fulfill the requirements of ADAS functions, while the computational requirements can be decreased in comparison to a purely grid-based map. By combining both ideas, a further step towards scalable multi-purpose map-based environment representations can be taken.

6.2. Future Works

The contributions of this thesis open up a variety of perspectives for future works. First of all, the developed framework for map-based representations, including the 2DIM and the concept of combining multiple maps, can be used to represent further environment features that can only be insufficiently described by using model-based object approaches. A promising example is the map-based accumulation of perspectively mapped image features, that can be used to improve the detection of the road infrastructure in complex urban environments.

Within the scope of work, an alternative way to represent dense environment features (e.g. occupancy states) within a 2DIM interval has been outlined and shall therefore be mentioned here as an inspiration for subsequent investigations. The key idea of this approach is to directly model the existence probability profile along the lateral axis of an interval. In that case, the content of an interval would be defined by a number of point cells $\boldsymbol{v}_n$ which represent the vertices of a polygonal chain that approximates the probability profile, as also illustrated in figure 6.1. In this model, both the lateral uncertainty about the positions and the uncertainty about the existence of an environment feature are represented by the polygonal chain. This model could help to realize a compact and

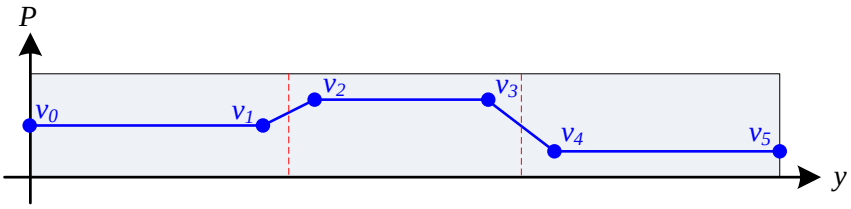


Figure 6.1.: Exemplary representation of occupancy information by polygonal chain. The blue dots illustrate the nodes of the polygonal line, whereas the red dotted lines indicate the occupancy state transitions.

straightforward update procedure, as the association problem between existing and new information significantly simplifies.

In the field of mapping common object motions and detecting convoy tracks, it would be interesting to investigate the results obtained by using unprocessed input data. By mapping raw laser and radar measurements, the described violations of the applied estimation principles concerning temporally correlated input data can be avoided. Within the 2DIM-framework, this information could further be fused with detections of other sensors, e.g. optical flow from image sequences.

Finally, also the presented concept of combining different map-based environment representations creates further research opportunities. As stated before, this concept would also allow for dynamically instantiating high-resolution occupancy maps at dedicated regions of interest. In order to take these decisions, a control mechanism that simultaneously considers the requirements of the ADAS functions, the structure of the surrounding environment and the available resources has to be developed. In this way, the proposed concept of combining grid- and interval-based representations can be used to optimally exploit the available computational resources.

A. Mathematical Appendix

This section lists fundamental mathematical principles and derivations that are needed for the development of the representations, sensor models, compensation mechanisms and extraction algorithms in this thesis.

A.1. Homogeneous Coordinates

Throughout the thesis, the homogeneous coordinates principle is used to describe coordinate transformations, e.g. between the coordinate system of a sensor and a representation. The incorporated transformation matrices are listed in the following.

A.1.1. 3D Homogeneous Coordinates

The transformation pose for three-dimensional homogeneous coordinates from a coordinate system A to the coordinate system B is defined by the following 4×4 matrix:

$$ {}^{B}\mathcal{P}_{A} = \left[\begin{array}{ccc|c} & {}^{B}\boldsymbol{R}_{A} & & {}^{B}\boldsymbol{p}_{A} \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \tag{A.1} $$

Here, the vector ${}^{B}\boldsymbol{p}_{A}$ denotes the origin of the coordinate system A expressed in coordinates of the system B. Similarly, the rotation matrix ${}^{B}\boldsymbol{R}_{A}$ describes the orientation of the system A in system B. According to the norm DIN 70000, the yaw-pitch-roll order convention will be used for the creation of the rotation matrix in this work. Consequently, the rotation matrix is defined by:

$$ {}^{B}\boldsymbol{R}_{A} = \mathcal{R}_{Z}\left(\psi\right) \cdot \mathcal{R}_{Y}\left(\theta\right) \cdot \mathcal{R}_{X}\left(\gamma\right) \tag{A.2} $$

with ψ being the yaw angle, θ the pitch angle and γ the role angle. Considering ${}^{B}\boldsymbol{p}_{A}$ as a translation vector $[x, y, z]^{T}$, we can also express the pose ${}^{B}\mathcal{P}_{A}$ as:

$$ {}^{B}\mathcal{P}_{A} = \mathcal{T}\left(x, y, z\right) \cdot \mathcal{R}_{Z}\left(\psi\right) \cdot \mathcal{R}_{Y}\left(\theta\right) \cdot \mathcal{R}_{X}\left(\gamma\right) \tag{A.3} $$

The rotation matrix around the z-axis with yaw angle ψ is defined as:

$$\mathcal{R}_Z(\psi) = \left[\begin{array}{ccc|c} \cos\psi & -\sin\psi & 0 & 0 \\ \sin\psi & \cos\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \tag{A.4}$$

The rotation matrix around the y-axis with pitch angle θ is defined as:

$$\mathcal{R}_Y(\theta) = \left[\begin{array}{ccc|c} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \tag{A.5}$$

The rotation matrix around the x-axis with roll angle γ is defined as:

$$\mathcal{R}_X(\gamma) = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma & 0 \\ 0 & \sin\gamma & \cos\gamma & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \tag{A.6}$$

The translation by a vector $\boldsymbol{t} = [x, y, z]^T$ can be realized by the matrix:

$$\mathcal{T}(x, y, z) = \left[\begin{array}{ccc|c} 1 & 0 & 0 & x \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & z \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \tag{A.7}$$

A.1.2. 2D Homogeneous Coordinates

In case of two-dimensional homogeneous coordinates, the pose ${}^B\mathcal{P}_A$ simplifies to a 3×3 matrix:

$${}^B\mathcal{P}_A = \left[\begin{array}{cc|c} & {}^B\boldsymbol{R}_A & {}^B\boldsymbol{p}_A \\ \hline 0 & 0 & 1 \end{array}\right] \tag{A.8}$$

In contrast to the three-dimensional case, the matrix ${}^B\boldsymbol{R}_A$ only has to express the rotation around the yaw angle ψ:

$${}^B\boldsymbol{R}_A = \mathcal{R}_Z(\psi) \tag{A.9}$$

In summary, the application of the pose reduces to a two-dimensional translation and a single rotation:

$$^{B}\mathcal{P}_{A} = \mathcal{T}(x, y) \cdot \mathcal{R}_{Z}(\psi) \tag{A.10}$$

The rotation matrix around the z-axis with yaw angle ψ is defined as:

$$\mathcal{R}_{Z}(\psi) = \left[\begin{array}{cc|c} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ \hline 0 & 0 & 1 \end{array}\right] \tag{A.11}$$

The translation by a vector $\boldsymbol{t} = [x, y]^{T}$ can be realized by the matrix:

$$\mathcal{T}(x, y) = \left[\begin{array}{cc|c} 1 & 0 & x \\ 0 & 1 & y \\ \hline 0 & 0 & 1 \end{array}\right] \tag{A.12}$$

A.2. Recursive State Estimation

A key concept of dealing with uncertain sensor measurements in the perception of the environment is the principle of recursive state estimation. The most important formulas and notations will be summarized in the following.

A.2.1. Model

The basic task of the approaches explained in the following is to estimate a system's state $\boldsymbol{x}$, which cannot be observed directly, but derived from unsure sensor measurements $\boldsymbol{z}$. In general, this state can be either static or dynamic and is optionally influenced by a control vector $\boldsymbol{u}$. Assuming a discrete time system, the evolution of the system state at time index k can be modeled by a *state transition function* $\boldsymbol{f}$:

$$\boldsymbol{x}_{k} = \boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k}, \boldsymbol{w}_{k-1}) \tag{A.13}$$

with $\boldsymbol{w}_{k-1}$ being the system noise, which describes an uncertainty about the state transition model. This model is based on the *Markov assumption*, which states that $\boldsymbol{x}_{k}$ incorporates the entire knowledge from previous states (also known as complete state assumption [165]). Consequently, the state $\boldsymbol{x}_{k}$ only depends on the previous state $\boldsymbol{x}_{k-1}$ and the control vector $\boldsymbol{u}_{k}$.

Similarly, the observation process can be described by a *measurement function* $\boldsymbol{h}$:

$$\boldsymbol{z}_{k} = \boldsymbol{h}(\boldsymbol{x}_{k}, \boldsymbol{v}_{k}) \tag{A.14}$$

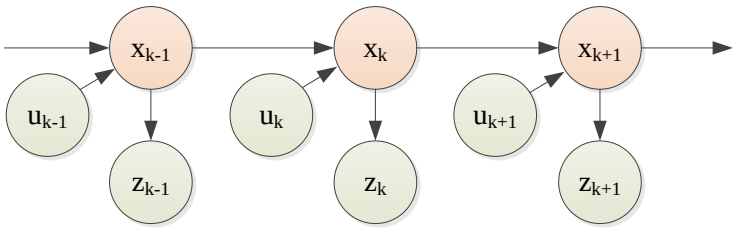


Figure A.1.: Dynamic Bayesian network illustrating the state transition and measurement process.

with $\boldsymbol{v}_k$ denoting the measurement noise. Figure A.1 illustrates both models in a dynamic Bayesian network.

Due to several conditional independencies in these relationships, we can simplify the probability distribution over a state $\boldsymbol{x}_k$:

$$p\left(\boldsymbol{x}_k|\boldsymbol{x}_{0:k-1}, \boldsymbol{u}_{1:k}, \boldsymbol{z}_{1:k-1}\right) = p\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \boldsymbol{u}_k\right) \tag{A.15}$$

Similarly, the probability distribution over the measurement $\boldsymbol{z}_k$ simplifies to:

$$p\left(\boldsymbol{z}_k|\boldsymbol{x}_{0:k}, \boldsymbol{u}_{1:k}, \boldsymbol{z}_{1:k-1}\right) = p\left(\boldsymbol{z}_k|\boldsymbol{x}_k\right) \tag{A.16}$$

A.2.2. Bayes Filter

The final goal of state estimation is to determine an unbiased estimate $\hat{\boldsymbol{x}}_k$ of the state $\boldsymbol{x}_k$ in each system cycle:

$$E\left(\hat{\boldsymbol{x}}_k\right) = E\left(\boldsymbol{x}_k\right) \tag{A.17}$$

Therefore, the *a posteriori* probability density function over all possible states given the observable measurements and control vectors has to be calculated:

$$p\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) \tag{A.18}$$

The most general solution to this problem is the recursive *Bayes Filter* formulation [165]. In this algorithm, the determination of the desired probability is subdivided into two essential steps. In the first step, the so called *a priori* or *predicted* probability distribution is calculated from all possible system states at time index $k-1$ and the most recent control input:

$$p\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right) = \int p\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \boldsymbol{u}_k\right) p\left(\boldsymbol{x}_{k-1}|\boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k-1}\right) d\boldsymbol{x}_{k-1} \tag{A.19}$$

In the second step, the a priori probability and the observed measurements are used to derive the a posteriori probability. This calculation is described as *update*, *correction* or *innovation* in the literature:

$$p\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}\right) = \eta\, p\left(\boldsymbol{z}_k|\boldsymbol{x}_k\right) p\left(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}\right) \tag{A.20}$$

with η denoting a normalization constant which assures that the resulting probabilities integrate to 1.

By using these equations and an initial a priori probability at time index 0, the a posteriori probability distribution over all possible states can be calculated recursively. For a detailed derivation of the Bayes Filter, the interested reader may e.g. refer to [165]. The deduction is based on repeated applications of the aforementioned Markov assumption and Bayes' theorem, which allows the fundamental conversion of conditional probabilities.

A.2.3. Kalman Filter

In 1960, Kalman presented his realization of a Bayes Filter for *continuous state* estimation in *discrete time linear Gaussian* systems [75]. Kalman and Bucy also adapted the filter to time continuous systems in [76]. Due to the mostly constant cycle times of the applied sensors in this thesis, the following derivations restrict to the discrete *KF*.

First of all, the KF is a *Gaussian Filter*, in which all represented probability density functions are given by multivariate normal distributions:

$$p\left(\boldsymbol{x}\right) = \mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) = \det\left(2\pi\boldsymbol{\Sigma}\right)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\left(\boldsymbol{x}-\boldsymbol{\mu}\right)^T \boldsymbol{\Sigma}^{-1}\left(\boldsymbol{x}-\boldsymbol{\mu}\right)\right\} \tag{A.21}$$

Gaussian probability distributions over the state space can be compactly described in closed form by their mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, which is also known as *moments parametrization*. As the covariance matrix $\boldsymbol{\Sigma}$ is symmetric per definition, there also exist optimizations which are restricted to the representation of an upper triangular covariance matrix [15]. Another advantage is given by the *unimodality* of the multivariate Gaussian distribution. The guaranteed existence of a single maximum value significantly simplifies the interpretation of the estimation results. Finally, Gaussian distributions also provide the essential characteristic that a linear transformation of the distribution results in another Gaussian distribution.

Due to this characteristic, the KF assumes linear relationships in the measurement (A.14) and state transition (A.13) functions. As a consequence, both equations can be formulated in the *state space*:

$$\boldsymbol{x}_k = \boldsymbol{\Phi}_{k-1}\boldsymbol{x}_{k-1} + \boldsymbol{G}_{k-1}\boldsymbol{u}_k + \boldsymbol{w}_{k-1} \tag{A.22}$$

$$\boldsymbol{z}_k = \boldsymbol{C}_k\boldsymbol{x}_k + \boldsymbol{v}_k \tag{A.23}$$

where system noise $\boldsymbol{w}_k$ and measurement noise $\boldsymbol{v}_k$ are assumed to be uncorrelated zero mean Gaussian white noise with covariance matrices $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$:

$$\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_k) \tag{A.24}$$

$$\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_k) \tag{A.25}$$

$$E\left(\boldsymbol{w}_k \boldsymbol{v}_k^T\right) = \boldsymbol{0} \tag{A.26}$$

Given this model, the KF allows to recursively calculate the posterior state estimate and corresponding error covariance. According to the Bayes Filter principle, the prediction step derives an a priori estimate of the state $\overset{*}{\boldsymbol{x}}_k$ and error covariance $\overset{*}{\boldsymbol{P}}_k$ on the basis of the state transition model:

$$\overset{*}{\boldsymbol{x}}_k = \boldsymbol{\Phi}_{k-1} \hat{\boldsymbol{x}}_{k-1} + \boldsymbol{G}_{k-1} \boldsymbol{u}_k \tag{A.27}$$

$$\overset{*}{\boldsymbol{P}}_k = \boldsymbol{\Phi}_{k-1} \hat{\boldsymbol{P}}_{k-1} \boldsymbol{\Phi}_{k-1}^T + \boldsymbol{Q}_{k-1} \tag{A.28}$$

In the innovation step, both a priori estimates are corrected by the current observation $\boldsymbol{z}_k$. The influence of the *residuum* between predicted and actual measurement to the new estimate is determined by the so called *Kalman gain* $\boldsymbol{K}_k$:

$$\hat{\boldsymbol{x}}_k = \overset{*}{\boldsymbol{x}}_k + \boldsymbol{K}_k \left(\boldsymbol{z}_k - \boldsymbol{C}_k \overset{*}{\boldsymbol{x}}_k\right) \tag{A.29}$$

$$\hat{\boldsymbol{P}}_k = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{C}_k) \overset{*}{\boldsymbol{P}}_k \tag{A.30}$$

$$\text{with } \boldsymbol{K}_k = \overset{*}{\boldsymbol{P}}_k \boldsymbol{C}_k^T \left(\boldsymbol{C}_k \overset{*}{\boldsymbol{P}}_k \boldsymbol{C}_k^T + \boldsymbol{R}_k\right)^{-1} \tag{A.31}$$

Given a linear Gaussian system with exactly known covariances, the Kalman filter is guaranteed to provide an optimal state estimate. For a detailed derivation of the presented equations, the interested reader is referred to [165].

A.2.4. Extended Kalman Filter

The EKF extends the principles of the KF to systems with nonlinear state transition functions $\boldsymbol{f}$ (A.13) and measurement functions $\boldsymbol{h}$ (A.14) [183]. Similar to the previous case, these functions can be used to initially calculate an a priori state estimate $\overset{*}{\boldsymbol{x}}_k$ and an expected measurement $\overset{*}{\boldsymbol{z}}_k$:

$$\overset{*}{\boldsymbol{x}}_k = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_k, \boldsymbol{0}) \tag{A.32}$$

$$\overset{*}{\boldsymbol{z}}_k = \boldsymbol{h}(\overset{*}{\boldsymbol{x}}_k, \boldsymbol{0}) \tag{A.33}$$

The key concept of the EKF is a linearization of both nonlinear functions around the predicted state $\overset{*}{\boldsymbol{x}}_k$ and the predicted measurement $\overset{*}{\boldsymbol{z}}_k$:

$$\boldsymbol{x}_k \approx \overset{*}{\boldsymbol{x}}_k + \boldsymbol{\Phi}_{k-1}(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1}) + \boldsymbol{\Gamma}_{k-1} \boldsymbol{w}_{k-1} \tag{A.34}$$

$$\boldsymbol{z}_k \approx \overset{*}{\boldsymbol{z}}_k + \boldsymbol{C}_k \quad (\boldsymbol{x}_k \quad - \overset{*}{\boldsymbol{x}}_k \quad) + \boldsymbol{\Lambda}_k \quad \boldsymbol{v}_k \tag{A.35}$$

In these equations, $\boldsymbol{\Phi}_{k-1}$ and $\boldsymbol{C}_k$ denote the Jacobians of the functions $\boldsymbol{f}$ and $\boldsymbol{h}$ with respect to the state vector $\boldsymbol{x}$. Similarly, $\boldsymbol{\Gamma}_{k-1}$ and $\boldsymbol{\Lambda}_k$ are the Jacobians of the related noise terms:

$$\boldsymbol{\Phi}_{k-1} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_k, \boldsymbol{0}) \qquad \boldsymbol{C}_k = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}(\overset{*}{\boldsymbol{x}}_k, \boldsymbol{0}) \tag{A.36}$$

$$\boldsymbol{\Gamma}_{k-1} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_k, \boldsymbol{0}) \qquad \boldsymbol{\Lambda}_k = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{v}}(\overset{*}{\boldsymbol{x}}_k, \boldsymbol{0}) \tag{A.37}$$

For the sake of simplicity, $\boldsymbol{\Gamma}_{k-1}$ and $\boldsymbol{\Lambda}_k$ are assumed to be the identity $\boldsymbol{I}$ in most derivations. In this case, the prediction and correction equations are similar to the KF. In the prediction step, the state transition matrix is replaced by the corresponding Jacobian, whereas the state estimate is calculated by using the nonlinear transition function:

$$\overset{*}{\boldsymbol{x}}_k = \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_k, \boldsymbol{0}\right) \tag{A.38}$$

$$\overset{*}{\boldsymbol{P}}_k = \boldsymbol{\Phi}_{k-1} \hat{\boldsymbol{P}}_{k-1} \boldsymbol{\Phi}_{k-1}^T + \boldsymbol{Q}_{k-1} \tag{A.39}$$

Similarly, the measurement matrix is replaced by the corresponding Jacobian in the correction step. The nonlinear measurement function is used to calculate the residuum between expected and new measurement:

$$\hat{\boldsymbol{x}}_k = \overset{*}{\boldsymbol{x}}_k + \boldsymbol{K}_k \left(\boldsymbol{z}_k - \boldsymbol{h}(\overset{*}{\boldsymbol{x}}_k, \boldsymbol{0}) \right) \tag{A.40}$$

$$\hat{\boldsymbol{P}}_k = \left(\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{C}_k\right) \overset{*}{\boldsymbol{P}}_k \tag{A.41}$$

$$\text{with } \boldsymbol{K}_k = \overset{*}{\boldsymbol{P}}_k \boldsymbol{C}_k^T \left(\boldsymbol{C}_k \overset{*}{\boldsymbol{P}}_k \boldsymbol{C}_k^T + \boldsymbol{R}_k \right)^{-1} \tag{A.42}$$

Due to the linearizations, the EKF does not fulfill the requirements of solely normally distributed random variables. In contrast to the KF, the obtained results are therefore not guaranteed to be optimal [112].

A.2.5. Implicit Measurement Equation

However, it is not always possible to assign the states $\boldsymbol{x}$ of a system to exactly one measurement $\boldsymbol{z}$, which prevents the formulation of a function $\boldsymbol{z} = f(\boldsymbol{x})$. In order to solve this problem, the EKF can be extended to incorporate an *implicit measurement equation*, as introduced in [6, 152]. The following derivations are adapted from [112]. The key idea of this approach is given by the description of the relation between state and measurement in a *nonlinear implicit measurement equation*:

$$\boldsymbol{g}\left(\tilde{\boldsymbol{z}}_k, \boldsymbol{x}_k\right) = \boldsymbol{0} \tag{A.43}$$

The current measurement $\boldsymbol{z}_k$ results from $\tilde{\boldsymbol{z}}_k$ and a zero mean Gaussian white noise:

$$\boldsymbol{z}_k = \tilde{\boldsymbol{z}}_k + \boldsymbol{v}_k \tag{A.44}$$

$$\boldsymbol{v}_k \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{R}_k\right) \tag{A.45}$$

A linearization of equation (A.43) at the current measurement $\boldsymbol{z}_k$ and the predicted state $\overset{*}{\boldsymbol{x}}_k$ yields:

$$\boldsymbol{g}(\check{\boldsymbol{z}}_k, \boldsymbol{x}_k) \approx \boldsymbol{g}(\boldsymbol{z}_k, \overset{*}{\boldsymbol{x}}_k) + \boldsymbol{C}_{\boldsymbol{z}_k}(\check{\boldsymbol{z}}_k - \boldsymbol{z}_k) + \boldsymbol{C}_{\boldsymbol{x}_k}(\boldsymbol{x}_k - \overset{*}{\boldsymbol{x}}_k) \approx \boldsymbol{0} \tag{A.46}$$

with the Jacobians $\boldsymbol{C}_{\boldsymbol{z}_k}$ and $\boldsymbol{C}_{\boldsymbol{x}_k}$ being defined as:

$$\boldsymbol{C}_{\boldsymbol{z}_k} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{z}}(\boldsymbol{z}_k, \overset{*}{\boldsymbol{x}}_k) \tag{A.47}$$

$$\boldsymbol{C}_{\boldsymbol{x}_k} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}(\boldsymbol{z}_k, \overset{*}{\boldsymbol{x}}_k) \tag{A.48}$$

The rearranged equation (A.46) can be interpreted as a measurement equation with a newly defined measurement $\boldsymbol{y}_k$ and the corresponding measurement noise $\boldsymbol{s}_k$:

$$\underbrace{-\boldsymbol{g}(\boldsymbol{z}_k, \overset{*}{\boldsymbol{x}}_k) + \boldsymbol{C}_{\boldsymbol{x}_k}\overset{*}{\boldsymbol{x}}_k}_{\boldsymbol{y}_k} = \boldsymbol{C}_{\boldsymbol{x}_k}\boldsymbol{x}_k + \underbrace{\boldsymbol{C}_{\boldsymbol{z}_k}(\check{\boldsymbol{z}}_k - \boldsymbol{z}_k)}_{\boldsymbol{s}_k} \tag{A.49}$$

Due to the general covariance calculation rule

$$\mathrm{Cov}(\boldsymbol{A}\boldsymbol{X}) = \boldsymbol{A}\mathrm{Cov}(\boldsymbol{X})\boldsymbol{A}^T \tag{A.50}$$

and equation (A.45), the measurement error covariance of $\boldsymbol{s}_k$ results as:

$$\mathrm{Cov}(\boldsymbol{s}_k) = \boldsymbol{C}_{\boldsymbol{z}_k}\boldsymbol{R}_k\boldsymbol{C}_{\boldsymbol{z}_k}^T \tag{A.51}$$

Assuming the measurement $\boldsymbol{y}_k$ from (A.49), the residuum between predicted and real measurement simplifies to:

$$\boldsymbol{y}_k - \boldsymbol{C}_{\boldsymbol{x}_k}\overset{*}{\boldsymbol{x}}_k = -\boldsymbol{g}(\boldsymbol{z}_k, \overset{*}{\boldsymbol{x}}_k) \tag{A.52}$$

Based on these derivations, the update equations of an EKF with implicit measurement equation can be formulated as:

$$\hat{\boldsymbol{x}}_k = \overset{*}{\boldsymbol{x}}_k + \boldsymbol{K}_k\left(-\boldsymbol{g}(\boldsymbol{z}_k, \overset{*}{\boldsymbol{x}}_k))\right) \tag{A.53}$$

$$\hat{\boldsymbol{P}}_k = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C}_{\boldsymbol{x}_k})\,\overset{*}{\boldsymbol{P}}_k \tag{A.54}$$

$$\text{with } \boldsymbol{K}_k = \overset{*}{\boldsymbol{P}}_k\boldsymbol{C}_{\boldsymbol{x}_k}^T\left(\boldsymbol{C}_{\boldsymbol{x}_k}\overset{*}{\boldsymbol{P}}_k\boldsymbol{C}_{\boldsymbol{x}_k}^T + \boldsymbol{C}_{\boldsymbol{z}_k}\boldsymbol{R}_k\boldsymbol{C}_{\boldsymbol{z}_k}^T\right)^{-1} \tag{A.55}$$

A.2.6. Static Binary State

Another well-known application of the Bayes Filter deals with the estimation of a binary static state. These models are typically used to deduce unchangeable environment features from a set of erroneous measurements $\boldsymbol{z}_{1:k}$, e.g. during the estimation of a grid cell's

occupancy state. The desired probability distribution can be reformulated by using Bayes' theorem and the Markov assumption:

$$p\left(x \mid \boldsymbol{z}_{1:k}\right) = \frac{p\left(\boldsymbol{z}_k \mid x, \boldsymbol{z}_{1:k-1}\right) \cdot p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{p\left(\boldsymbol{z}_k \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.56}$$

$$= \frac{p\left(\boldsymbol{z}_k \mid x\right) \cdot p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{p\left(\boldsymbol{z}_k \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.57}$$

Due to the unchangeable state, control operations $\boldsymbol{u}_k$ and the time index of the state variable x are neglected. As there are only two possible state values, the following equation applies:

$$p\left(x \mid \boldsymbol{z}_{1:k}\right) = 1 - p\left(\neg x \mid \boldsymbol{z}_{1:k}\right) \tag{A.58}$$

Based on these relationships, two different recursive estimation algorithms can be formulated.

The first approach is based on the forward sensor model $p\left(\boldsymbol{z}_k|x\right)$, which quantifies the probability of all possible measurements given a state:

$$p\left(x \mid \boldsymbol{z}_{1:k}\right) = \frac{p\left(\boldsymbol{z}_k \mid x\right) \cdot p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{p\left(\boldsymbol{z}_k \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.59}$$

$$= \frac{p\left(\boldsymbol{z}_k \mid x\right) \cdot p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{\sum_{x'} p\left(\boldsymbol{z}_k \mid x'\right) \cdot p\left(x' \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.60}$$

$$= \frac{p\left(\boldsymbol{z}_k \mid x\right) \cdot p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{p\left(\boldsymbol{z}_k \mid x\right) \cdot p\left(x \mid \boldsymbol{z}_{1:k-1}\right) + p\left(\boldsymbol{z}_k \mid \neg x\right) \cdot p\left(\neg x \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.61}$$

Given the forward sensor models for both possible states, the desired state probability distribution can be calculated recursively.

By contrast, the second approach is based on the inverse sensor model $p\left(x|\boldsymbol{z}_k\right)$, which specifies the distribution over possible states depending on the current measurement:

$$p\left(x \mid \boldsymbol{z}_{1:k}\right) = \frac{p\left(\boldsymbol{z}_k \mid x\right) \cdot p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{p\left(\boldsymbol{z}_k \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.62}$$

$$= \frac{p\left(x \mid \boldsymbol{z}_k\right) \cdot p\left(\boldsymbol{z}_k\right) \cdot p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{p\left(x\right) \cdot p\left(\boldsymbol{z}_k \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.63}$$

Still, this equation includes probabilities, which are difficult to determine. By using the corresponding equation for the opposite state $\neg x$

$$p\left(\neg x \mid \boldsymbol{z}_{1:k}\right) = \frac{p\left(\neg x \mid \boldsymbol{z}_k\right) \cdot p\left(\boldsymbol{z}_k\right) \cdot p\left(\neg x \mid \boldsymbol{z}_{1:k-1}\right)}{p\left(\neg x\right) \cdot p\left(\boldsymbol{z}_k \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.64}$$

we can further simplify the probability update. The most popular approach is based on the so called *log odds* formulation $l_k(x)$, which can be derived by dividing (A.63) by (A.64):

$$\frac{p\left(x \mid \boldsymbol{z}_{1:k}\right)}{p\left(\neg x \mid \boldsymbol{z}_{1:k}\right)} = \frac{p\left(x \mid \boldsymbol{z}_k\right)}{p\left(\neg x \mid \boldsymbol{z}_k\right)} \cdot \frac{p\left(\neg x\right)}{p\left(x\right)} \cdot \frac{p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{p\left(\neg x \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.65}$$

$$\frac{p\left(x \mid \boldsymbol{z}_{1:k}\right)}{1-p\left(x \mid \boldsymbol{z}_{1:k}\right)} = \frac{p\left(x \mid \boldsymbol{z}_k\right)}{1-p\left(x \mid \boldsymbol{z}_k\right)} \cdot \frac{1-p\left(x\right)}{p\left(x\right)} \cdot \frac{p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{1-p\left(x \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.66}$$

$$log\frac{p\left(x \mid \boldsymbol{z}_{1:k}\right)}{1-p\left(x \mid \boldsymbol{z}_{1:k}\right)} = log\frac{p\left(x \mid \boldsymbol{z}_k\right)}{1-p\left(x \mid \boldsymbol{z}_k\right)} + log\frac{1-p\left(x\right)}{p\left(x\right)} + log\frac{p\left(x \mid \boldsymbol{z}_{1:k-1}\right)}{1-p\left(x \mid \boldsymbol{z}_{1:k-1}\right)} \tag{A.67}$$

$$l_k\left(x\right) = log\frac{p\left(x \mid \boldsymbol{z}_k\right)}{1-p\left(x \mid \boldsymbol{z}_k\right)} + log\frac{1-p\left(x\right)}{p\left(x\right)} + l_{k-1}\left(x\right) \tag{A.68}$$

Assuming that $l_0\left(x\right) = 0.5$, we can simplify the recursive state estimation equation in this formulation to:

$$l_k\left(x\right) = log\frac{p\left(x \mid \boldsymbol{z}_k\right)}{1-p\left(x \mid \boldsymbol{z}_k\right)} + l_{k-1}\left(x\right) \tag{A.69}$$

By using this formulation, the computational effort of updating the a-posteriori state estimate is reduced to a single addition.

Besides that, the desired probability $p\left(x \mid \boldsymbol{z}_{1:k}\right)$ can also be inferred from equation (A.66) without using log odds ratios:

$$p\left(x|\boldsymbol{z}_{1:k}\right) = \frac{p\left(x|\boldsymbol{z}_k\right)}{1-p\left(x|\boldsymbol{z}_k\right)} \cdot \frac{1-p\left(x\right)}{p\left(x\right)} \cdot \frac{p\left(x|\boldsymbol{z}_{1:k-1}\right)}{1-p\left(x|\boldsymbol{z}_{1:k-1}\right)} \cdot \left(1-p\left(x|\boldsymbol{z}_{1:k}\right)\right) \tag{A.70}$$

$$\Rightarrow p\left(x|\boldsymbol{z}_{1:k}\right) = \frac{\frac{p(x|\boldsymbol{z}_k)}{1-p(x|\boldsymbol{z}_k)} \cdot \frac{1-p(x)}{p(x)} \cdot \frac{p(x|\boldsymbol{z}_{1:k-1})}{1-p(x|\boldsymbol{z}_{1:k-1})}}{1+\frac{p(x|\boldsymbol{z}_k)}{1-p(x|\boldsymbol{z}_k)} \cdot \frac{1-p(x)}{p(x)} \cdot \frac{p(x|\boldsymbol{z}_{1:k-1})}{1-p(x|\boldsymbol{z}_{1:k-1})}} \tag{A.71}$$

$$= \frac{\frac{p(x|\boldsymbol{z}_k)}{1-p(x|\boldsymbol{z}_k)} \cdot \frac{1-p(x)}{p(x)} \cdot \frac{p(x|\boldsymbol{z}_{1:k-1})}{1-p(x|\boldsymbol{z}_{1:k-1})}}{\frac{(1-p(x|\boldsymbol{z}_k))\cdot p(x)\cdot(1-p(x|\boldsymbol{z}_{1:k-1}))}{(1-p(x|\boldsymbol{z}_k))\cdot p(x)\cdot(1-p(x|\boldsymbol{z}_{1:k-1}))} + \frac{p(x|\boldsymbol{z}_k)\cdot(1-p(x))\cdot p(x|\boldsymbol{z}_{1:k-1})}{(1-p(x|\boldsymbol{z}_k))\cdot p(x)\cdot(1-p(x|\boldsymbol{z}_{1:k-1}))}} \tag{A.72}$$

$$= \frac{p\left(x|\boldsymbol{z}_k\right)\cdot\left(1-p\left(x\right)\right)\cdot p\left(x|\boldsymbol{z}_{1:k-1}\right)}{\left(1-p\left(x|\boldsymbol{z}_k\right)\right)\cdot p\left(x\right)\cdot\left(1-p\left(x|\boldsymbol{z}_{1:k-1}\right)\right) + p\left(x|\boldsymbol{z}_k\right)\cdot\left(1-p\left(x\right)\right)\cdot p\left(x|\boldsymbol{z}_{1:k-1}\right)} \tag{A.73}$$

Under the assumption of a uniformly distributed a priori state probability, this approach results in a slightly more complex recursive update rule:

$$p\left(x|\boldsymbol{z}_{1:k}\right) = \frac{p\left(x|\boldsymbol{z}_k\right)\cdot p\left(x|\boldsymbol{z}_{1:k-1}\right)}{\left(1-p\left(x|\boldsymbol{z}_k\right)\right)\cdot\left(1-p\left(x|\boldsymbol{z}_{1:k-1}\right)\right) + p\left(x|\boldsymbol{z}_k\right)\cdot p\left(x|\boldsymbol{z}_{1:k-1}\right)} \tag{A.74}$$

For the sake of readability, the recursively estimated probabilities can be redefined:

$$p_k = \frac{p\left(x|\boldsymbol{z}_k\right)\cdot p_{k-1}}{\left(1-p\left(x|\boldsymbol{z}_k\right)\right)\cdot\left(1-p_{k-1}\right) + p\left(x|\boldsymbol{z}_k\right)\cdot p_{k-1}} \tag{A.75}$$

A.3. Dempster-Shafer Theory

As stated in section 3.1, the key principle of the DST is to assign *belief masses* to the power set of possible states Ω. Based on these values, the *belief* and *plausibility* of an element A of Ω can be derived. The belief $bel\,(A)$ is given by the sum of all masses m of subsets of A. The plausibility $pl\,(A)$ is given by the sum of all masses that intersect A:

$$bel\,(A) = \sum_{B|B\subseteq A} m\,(B) \qquad\qquad pl\,(A) = \sum_{B|B\cap A\neq\emptyset} m\,(B) \tag{A.76}$$

In case of an occupancy grid cell, the power set of possible states Ω results as:

$$2^{\Omega} = \begin{cases} \emptyset \\ O:\ occupied \\ F:\ free \\ U:\ unknown = \{occupied, free\} \end{cases} \tag{A.77}$$

from which the mass of the empty set is zero by definition. Based on *Dempster's rule of combination*, these belief masses can be updated by using the following equations [42]:

$$m_{1:k+1}(O) = \frac{m_{1:k}(O)\cdot m_{k+1}(O) + m_{1:k}(U)\cdot m_{k+1}(O) + m_{1:k}(O)\cdot m_{k+1}(U)}{1 - m_{1:k}(O)\cdot m_{k+1}(F) - m_{1:k}(F)\cdot m_{k+1}(O)} \tag{A.78}$$

$$m_{1:k+1}(F) = \frac{m_{1:k}(F)\cdot m_{k+1}(F) + m_{1:k}(U)\cdot m_{k+1}(F) + m_{1:k}(F)\cdot m_{k+1}(U)}{1 - m_{1:k}(O)\cdot m_{k+1}(F) - m_{1:k}(F)\cdot m_{k+1}(O)} \tag{A.79}$$

$$m_{1:k+1}(U) = 1 - m_{1:k+1}(O) - m_{1:k+1}(F) \tag{A.80}$$

A.4. Transformation of Measurement Uncertainties

The developed sensor models in this thesis transform the uncertainties of the input data into the obtained representations. The derived formulas are summarized in the following two sections.

A.4.1. Occupancy Map Sensor Model

Let $\boldsymbol{C_z}$ be the partial derivative of the implicit measurement equation with respect to the current measurement from (3.22), $\boldsymbol{R}$ be the measurement error from (3.15) and ${}^{I^{(n)}}\mathcal{P}_{Sensor}$ be the following pose between sensor and interval coordinate system:

$${}^{I^{(n)}}\mathcal{P}_{Sensor} = \mathcal{T}\,(x, y, z)\cdot\mathcal{R}_Z\,(\psi)\cdot\mathcal{R}_Y\,(\theta) \tag{A.81}$$

Then, the transformation of the measurement error into the state space results as [1]:

$$\begin{aligned}\boldsymbol{C_z R C_z^T} = \\ &\mathrm{Var}(\varphi)\left(r\cos(\varphi+\alpha/2)\cos(\psi) - r\sin(\varphi+\alpha/2)\cos(\theta)\sin(\psi)\right)^2 \\ +&\mathrm{Var}(r)\left(\sin(\varphi+\alpha/2)\cos(\psi) + \sin(\theta)\sin(\vartheta)\sin(\psi) + \cos(\varphi+\alpha/2)\cos(\theta)\sin(\psi)\right)^2 \\ +&r^2\mathrm{Var}(\vartheta)\cos(\vartheta)^2\sin(\theta)^2\sin(\psi)^2 \end{aligned} \tag{A.82}$$

If we further neglect the pitch angle in the pose between sensor and interval coordinate system:

$${}^{I^{(n)}}\mathcal{P}_{Sensor} = \mathcal{T}(x,y,z)\cdot\mathcal{R}_Z(\psi) \tag{A.83}$$

the transformed error simplifies to:

$$\begin{aligned}\boldsymbol{C_z R C_z^T} = \\ &\mathrm{Var}(\varphi)(r\cos(\varphi+\alpha/2)\cos(\psi) - r\sin(\varphi+\alpha/2)\sin(\psi))^2 \\ +&\mathrm{Var}(r)\,(\ \cos(\varphi+\alpha/2)\sin(\psi) + \ \sin(\varphi+\alpha/2)\cos(\psi))^2 \end{aligned} \tag{A.84}$$

A.4.2. Sensor Model for Motion Cells

Given the implicit measurement equation (4.8), the partial derivative $\boldsymbol{C_z}$ with respect to the measurement vector formulated in 4.7 results as:

$$\boldsymbol{C_z} = \begin{bmatrix} -\sin(\psi^{(n)}) & -\cos(\psi^{(n)}) & 0 & 0 & \lambda\sin(\psi^{(n)})\cos(\delta) - \lambda\cos(\psi^{(n)})\cos(\delta) \\ 0 & 0 & -\cos(\delta) & \sin(\delta) & v_y\cos(\delta) + v_x\sin(\psi^{(n)}) \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{A.85}$$

with ϕ, v_x, v_y denoting the measured object's angle and velocity components and $\psi^{(n)}$ representing the angle of the updated interval in the USK.

A.5. Velocities and Accelerations in Rotating Coordinate Systems

When converting the motion of a model-based object hypothesis into the static cells of a map-based environment representation, the rotational behavior of an object has to be considered. Formally, this problem corresponds to a transformation of velocities and accelerations between rotating coordinate systems.

[1] Time index k has been left out for the sake of clarity

Let $\boldsymbol{r}'$ be a coordinate in a reference frame S' that rotates around its origin with angular velocity ω. The velocity and acceleration within the rotating reference frame is given by the first and second time derivative, which will be denoted as:

$$\left\langle \frac{d\boldsymbol{r}'}{dt} \right\rangle = \left[v_{x'}, v_{y'}, v_{z'}\right]^T \tag{A.86}$$

$$\left\langle \frac{d^2\boldsymbol{r}'}{dt^2} \right\rangle = \left[a_{x'}, a_{y'}, a_{z'}\right]^T \tag{A.87}$$

The corresponding coordinate $\boldsymbol{r}$ in a stationary frame S can be calculated by the inverse rotation matrix:

$$\boldsymbol{r} = \left(\mathcal{R}_Z\left(\omega t\right)\right)^{-1} \boldsymbol{r}' \tag{A.88}$$

Due to the angular velocity ω, the velocity of $\boldsymbol{r}$ in the stationary frame changes to [60]:

$$\frac{d\boldsymbol{r}}{dt} = \left\langle \frac{d\boldsymbol{r}'}{dt} \right\rangle + \omega \times \boldsymbol{r} \tag{A.89}$$

with $\times$ denoting the vector cross product. Based on this transformation, the acceleration at $\boldsymbol{r}$ in the stationary frame can be derived as [60]:

$$\begin{aligned}
\frac{d^2\boldsymbol{r}}{dt^2} &= \frac{d}{dt}\left\langle \frac{d\boldsymbol{r}'}{dt} \right\rangle + \frac{d}{dt}\left(\omega \times \boldsymbol{r}\right) && \text{(A.90)}\\
&= \frac{d}{dt}\left\langle \frac{d\boldsymbol{r}'}{dt} \right\rangle + \frac{d\omega}{dt} \times \boldsymbol{r} + \omega \times \frac{d\boldsymbol{r}}{dt} && \text{(A.91)}\\
&\overset{\text{(A.89)}}{=} \frac{d}{dt}\left\langle \frac{d\boldsymbol{r}'}{dt} \right\rangle + \frac{d\omega}{dt} \times \boldsymbol{r} + \omega \times \left\langle \frac{d\boldsymbol{r}'}{dt} \right\rangle + \omega \times \left(\omega \times \boldsymbol{r}\right) && \text{(A.92)}\\
&\overset{\text{(A.89)}}{=} \left\langle \frac{d^2\boldsymbol{r}'}{dt^2} \right\rangle + \underbrace{\frac{d\omega}{dt} \times \boldsymbol{r}}_{\substack{0 \\ \text{if } \omega \text{ const.}}} + \underbrace{2\left(\omega \times \left\langle \frac{d\boldsymbol{r}'}{dt} \right\rangle\right)}_{\substack{\text{Coriolis} \\ \text{Acceleration}}} + \underbrace{\omega \times \left(\omega \times \boldsymbol{r}\right)}_{\substack{\text{Centrifugal} \\ \text{Acceleration}}} && \text{(A.93)}
\end{aligned}$$

If we apply these results to a coordinate in a system that rotates with constant yaw rate $\dot{\psi}$ around the z-axis of a stationary frame, the velocity in the stationary frame results as:

$$\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} v_{x'} \\ v_{y'} \\ v_{z'} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 0 \end{bmatrix} = \begin{bmatrix} v_{x'} - \dot{\psi} y \\ v_{y'} + \dot{\psi} x \\ v_{z'} \end{bmatrix} \tag{A.94}$$

Similarly, the acceleration in the stationary frame results as:

$$\begin{aligned}
\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} &= \begin{bmatrix} a_{x'} \\ a_{y'} \\ a_{z'} \end{bmatrix} + 2\left(\begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} v_{x'} \\ v_{y'} \\ v_{z'} \end{bmatrix}\right) + \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \times \left(\begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}\right) && \text{(A.95)}\\
&= \begin{bmatrix} a_{x'} - 2\dot{\psi} v_{y'} - \dot{\psi}^2 x \\ a_{y'} + 2\dot{\psi} v_{x'} - \dot{\psi}^2 y \\ a_{z'} \end{bmatrix} && \text{(A.96)}
\end{aligned}$$

If we further apply these results to the prediction of a 2DIM cell border position by the lateral velocity and acceleration of an associated object, the new lateral border position can be derived as:

$$\overset{*}{y}_{k+1} = \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{\boldsymbol{k}}\right) = \begin{bmatrix} 1 & T & T^2/2 \end{bmatrix} \begin{bmatrix} \hat{y}_k \\ v_{y'} + \dot{\psi} x \\ a_{y'} + 2\dot{\psi} v_{x'} - \dot{\psi}^2 y \end{bmatrix} \tag{A.97}$$

$$\text{with} \quad \hat{\boldsymbol{x}}_k = \begin{bmatrix} \hat{y}_k & x' & y' & v_{x'} & v_{y'} & a_{x'} & a_{y'} & \dot{\psi} \end{bmatrix}^T \tag{A.98}$$

with x', y' being the object's position, $v_{x'}, v_{y'}$ being the object velocity in the object coordinate system, $a_{x'}, a_{y'}$ being the object acceleration in the object coordinate system and $\dot{\psi}$ denoting the object's yaw rate. Assuming a diagonal error covariance matrix $\mathrm{Cov}(\hat{\boldsymbol{x}}_k)$, whose entries are given by the estimation variances of the introduced state vector elements, the application of this function modifies the variance of a border position as follows:

$$\mathrm{Var}(\overset{*}{y}_{k+1}) = \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\hat{\boldsymbol{x}}_k)\right) \mathrm{Cov}(\hat{\boldsymbol{x}}_k) \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\hat{\boldsymbol{x}}_k)\right)^T \tag{A.99}$$

$$\begin{aligned} =&\mathrm{Var}(\hat{y}_k) + \\ &Var(x')\dot{\psi}^2\ T^2 + \mathrm{Var}(y')\dot{\psi}^4 T^4/4 + \mathrm{Var}(v_{x'})\dot{\psi}^2 T^4 + \\ &\mathrm{Var}(v_{y'})T^2 + \mathrm{Var}(a_{y'})T^4/4 + \mathrm{Var}(\dot{\psi})(Tx' + (T^2(2\dot{v}_{x'} - 2\dot{\psi}y'))/2)^2 \end{aligned} \tag{A.100}$$

A.6. Clothoid Approximations

For a compact representation of extracted convoy tracks from a 2DIM with motion cells the application of clothoid approximations is proposed in this thesis. Formally, a clothoid is defined as a curve, whose radius of curvature r is inversely proportional to the curve length l:

$$r = \frac{a^2}{l} \tag{A.101}$$

with a being a fixed positive constant. Given this prerequisite, the complete track can be exactly described by:

$$\begin{bmatrix} x \\ y \end{bmatrix} = a\sqrt{\pi} \int_0^t \begin{bmatrix} \cos\frac{\pi\xi^2}{2} \\ \sin\frac{\pi\xi^2}{2} \end{bmatrix} d\xi \qquad \text{with} \quad t = \frac{l}{a\sqrt{\pi}} \tag{A.102}$$

The curvature of the resulting path is composed of an initial curvature c_0 and a change of curvature c_1:

$$c(l) = c_0 + c_1 \cdot l \tag{A.103}$$

In order to infer clothoids from a set of $(x,\ y)$-pairs, a method presented by Reichel in [133, appx. E] was applied within the scope of this thesis. In this approach, a Taylor approximation of the equation A.102 is used to formulate equations of the form:

$$y = \begin{bmatrix} \frac{1}{2}x^2 & \frac{1}{6}x^3 & x & 1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \tan(\zeta) \\ dy \end{bmatrix} \tag{A.104}$$

with ζ and dy being the direction and lateral offset of the clothoid. Given a set of more than four $(x,\ y)$-pairs of an extracted polygonal chain, an overdetermined equation system can be established to infer an optimal clothoid approximation. This equation system can then be solved by using common methods of linear algebra, e.g. calculating the *pseudoinverse* matrix.

B. Vehicles and Sensors

The following sections briefly describe the sensor configuration of the test vehicle that was used to obtain the measurement data for the developments in this thesis. After that, the key principles of the applied radar and laser sensors will be described.

B.1. Test Vehicle Sensor Configuration

For a prototypical implementation and validation of the developed approaches, real sensor data from an available test vehicle were used. Besides state-of-the-art ADAS-sensors, the vehicle was equipped with a close-to-production automotive laser scanner and additional laser respectively video sensors for the areas right and left to the vehicle. Figure B.1 illustrates the entire sensor configuration. The conventional sensor set consists of

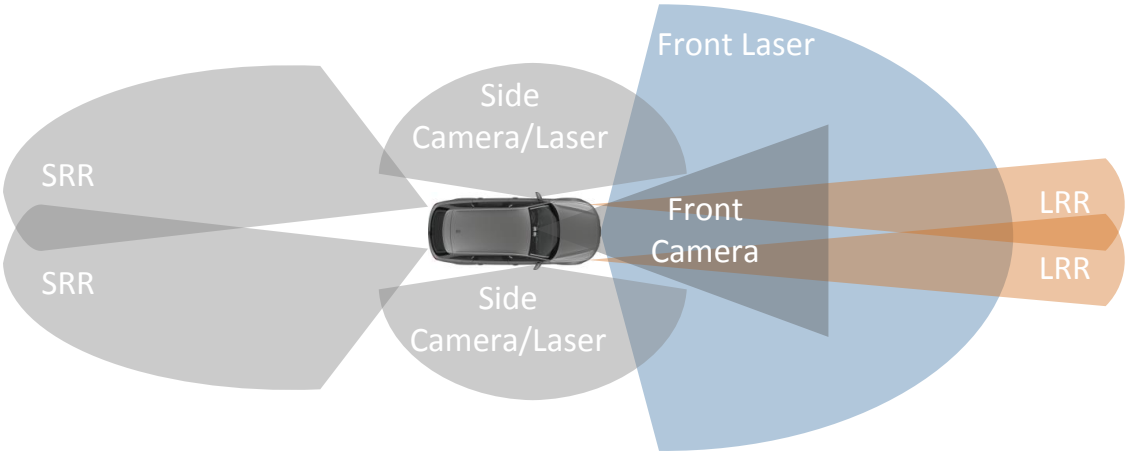


Figure B.1.: Illustration of the test vehicle sensor configuration.

- Two LRR3 sensors in the vehicle's front, which are used for an ACC system
- A video camera, which is able to detect road markings, traffic signs and other vehicles
- Two Short Range Radar (SRR) sensors in the vehicle's rear, which are part of a blind spot detection system

The developments in this thesis focused on the utilization of the laser- and radar sensors in the vehicle's front area. The corresponding measurement principles will be described in the following.

B.2. Radar

Since the market launch of first ACC systems in 1998, automotive Radio Detection and Ranging (RADAR) sensors have gained an enormous popularity. Today, there exists a wide range of mass production radar sensors, which are mainly categorized according to the applied modulation methods and receiving antenna technologies [186]. Concerning the signal modulation, a distinction is made between continuous wave and impulse RADAR systems as well as between amplitude and frequency modulated signals. Regarding the receiving antennas, there exist mechanically or electronically scanning systems and conventional fixed beam sensor arrays. All RADAR measurements presented in this work were obtained by using a Bosch double LRR3 system [95]. Therefore, the following section is restricted to a brief outline of the applied Frequency Modulated Continous Wave (FMCW) measurement principle.

Continuous wave radar systems transmit harmonic wave functions, which are defined as:

$$u_{\mathrm{t}}(t) = A_t \cdot \cos\left(2\pi f_0 t + \varphi_0\right) \tag{B.1}$$

with amplitude A_t, frequency f_0 and phase φ_0. In general, all three parameters can be used for modulations. The received signal has a different amplitude A_r and a phase shift φ_r. Furthermore, it also includes the Doppler frequency, which depends on the change of the measured object's radial distance $\dot{r}$:

$$u_{\mathrm{r}}(t) = A_r \cdot \cos\left(2\pi(f_0 + f_{\mathrm{Doppler}})t + \varphi_r\right) \tag{B.2}$$

$$\text{with } f_{\mathrm{Doppler}} = -2\dot{r} f_0 / c \tag{B.3}$$

with c denoting the speed of light.

FMCW radar systems continuously change the currently transmitted frequency ω in linear ramp waveform:

$$\omega(t) = \omega_0 + m_\omega (t - t_0) \tag{B.4}$$

with ω_0 being the initial frequency and m_ω denoting the gradient of the ramp. An example of a transmitted positive ramp signal is shown in a time-frequency-diagram in figure B.2 a). The received signal is shifted along the time axis due to the distance of the measured object and along the frequency axis due to the Doppler effect. The mixing of the transmitted and received signals results in the following low frequency signal component [186]:

$$u(t)_{\overline{\mathrm{lt,r}}} = \frac{1}{2} A_r A_t \cos\left(\left(\frac{2\omega_0}{c}\dot{r} + \frac{2m_\omega}{c} r\right) t + \frac{2r}{c}\omega_0 + \left(\frac{2r}{c}\right)^2 m_\omega\right) \tag{B.5}$$

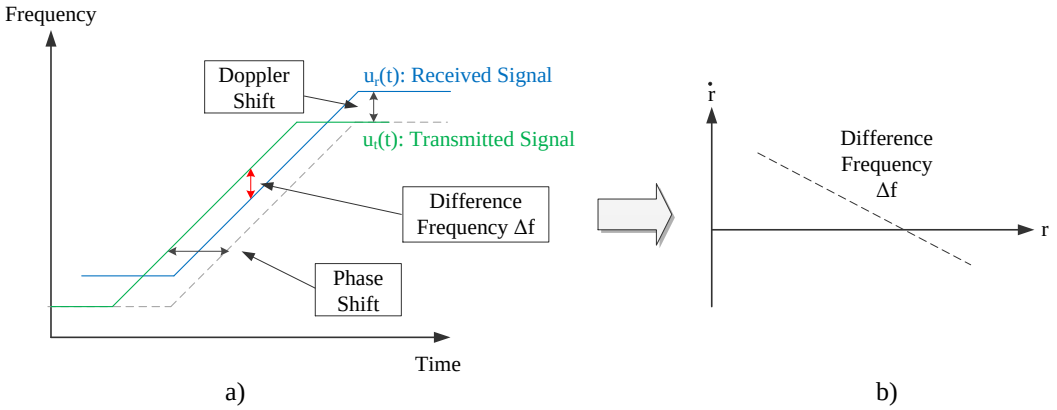


Figure B.2.: FMCW radar signal characteristics of an approaching object, adapted from [186].

The so called *difference frequency* of this signal can be processed to infer a linear relationship between the radial distance r and the radial velocity $\dot{r}$ of a measured object, as illustrated in figure B.2 b). By using another, preferably negative ramp signal, the relative distance and velocity of a single object can be identified unambiguously. In order to measure multiple objects, several different ramp signals have to be used [186].

The LRR3 uses this modulation technique in the frequency band between 76 and 77 Ghz to measure objects in a distance range between 0.5 and 250 m at an opening angle of 30° [186]. The angle of the identified objects is determined by comparing the obtained measurements to the sensor specific characteristics of the integrated four beam antenna. In the applied double LRR3 setup, both sensors have asymmetric sensor characteristics in order to improve the close range detection [95]. After having extracted the radial distance, velocity and azimuth angle, all echos that belong to a common object are clustered. These object hypotheses are tracked over time to improve the quality of the finally resulting object list. For the experiments in this thesis, only temporally filtered radar object lists were available.

The major advantages of radar sensors are given by their high detection range, precise velocity measurements and robustness against weather conditions. On the downside, the applied LRR3 sensor is not able to provide exact contours of surrounding objects or information about free spaces. The provided object list includes echos from static targets as well as visible and occluded dynamic objects. Due to the combination of two sensors, also rough estimates about the dimensions of the preceding vehicle can be inferred. In state-of-the-art driver assistance sensor setups, radar measurements are also fused with video information to improve the detection accuracy. A detailed analysis of the radar object list in dense traffic scenarios is presented in section 4.3.1.

B.3. Lidar

Instead of electromagnetic waves, Light Detection and Ranging (LIDAR) sensors use light waves to infer positions of surrounding objects. Automotive sensors typically restrict to infrared light with wavelengths between 850 nm and 1000 nm [58]. Overall, there exist two major time of flight measurement principles: The application of modulated light, e.g. in Photonic Mixer Device (PMD) sensors, and the utilization of pulsed laser light sources [154]. Due to the available sensors in the scope of this thesis, the following explanations will be restricted to the second approach, which is also illustrated in figure B.3.

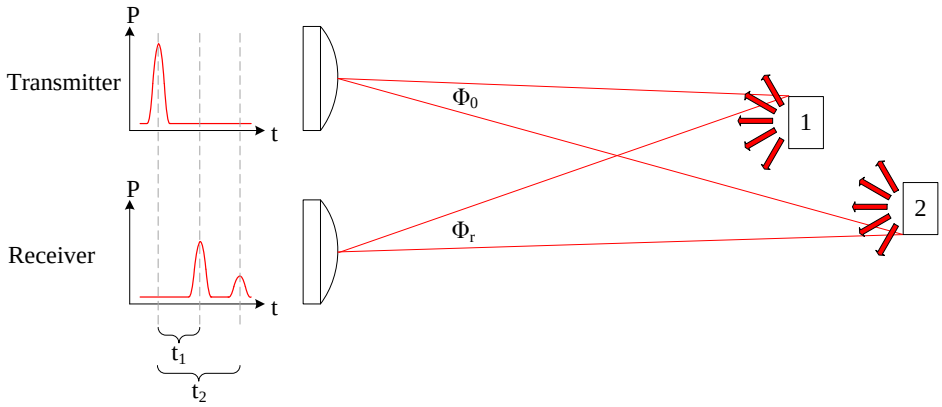


Figure B.3.: Laser pulse time of flight measurement, adapted from [154].

In this case, the measurement process starts with the transmission of a short light impulse. The emitted light output Φ_0 is either absorbed (Φ_a), diffusely scattered (Φ_s) or reflected to the measurement unit (Φ_r), which is typically realized by an APD[1] or PIN[2] diode [58]:

$$\phi_0 = \Phi_s + \Phi_a + \Phi_r \tag{B.6}$$

The fraction of the reflected light Φ_r depends on the radial distance r as well as the angle, size and reflectivity of the measured object and the surrounding atmosphere [54]. As a rule of thumb, the received light energy scales with $1/r^2$ if the illuminated area is larger than the entire measured object and with $1/r^4$ otherwise [154].

As illustrated in figure B.3, the distances of possibly multiple measured targets can be inferred from the received light signal curve. An analysis of the incoming echo pulse widths can help to recognize invalid targets, e.g. due to fog, rain or dirt on the sensor. In laser scanner systems, the emitted light beam is additionally deflected in order to enlarge the overall detection area. As a consequence, the measurements of the different channels are not exactly taken at the same time. Besides that, there also exist approaches, which

[1] Avalanche PhotoDiode
[2] Positive Intrinsic Negative

utilize several light sources or uniformly illuminate the detection area by a flash light, e.g. described in [16].

For the experiments in this thesis, a close-to-production automotive laser scanner was applied. In order to provide a high-resolution scan of the surroundings, the sensor takes multiple measurements in several vertical *layers* l and horizontal *channels* c. The entire scan at time index k is composed of $L \times C$ individual range measurements:

$$\boldsymbol{z}_k = \left[z_k^{(0,0)}, \cdots, z_k^{(l,c)}, \cdots, z_k^{(L,C)} \right] \tag{B.7}$$

$$\text{with} \quad z_k^{(l,c)} = \left[r_k^{(l,c)}, \vartheta_k^{(l)}, \varphi_k^{(c)} \right] \tag{B.8}$$

with $\vartheta_k^{(l)}$ denoting the vertical angle and $\varphi_k^{(c)}$ being the horizontal angle of the range measurement. Due to the divergence of the emitted laser beam and the receiving unit's detection zone, each measurement covers a specific horizontal and vertical angular range, which is denoted as α and β in this thesis. Consequently, the reflecting obstacle can be located within a spherical quadrilateral. By an analysis of the detected echo pulse characteristics and the distribution of the neighboring measurements, the sensor already provides a classification of the reflection into the categories *valid*, *ground* and *clutter*.

In contrast to the previously introduced radar sensors, high resolution laser scanners are able to provide detailed information about contours of surrounding objects and free spaces, which is an essential requirement for highly automated vehicles. On the downside, the measurements do not include velocity information and deteriorate with bad visibility conditions. The overall detection range depends on the intensity of the emitted light pulse, the sensitivity of the receiving unit, the characteristics of the measured object and the surrounding atmosphere.

C. Software Tooling Environment

All software modules which were created in the scope of this work were implemented by using ADTF, a framework, which allows an improved development of software prototypes for driver assistance systems. The basic idea of ADTF is to implement modules, called *filters*, in a predefined C++ structure, which can then be combined and parametrized in a graphical user interface. An example of this interface can be seen in figure C.1. These

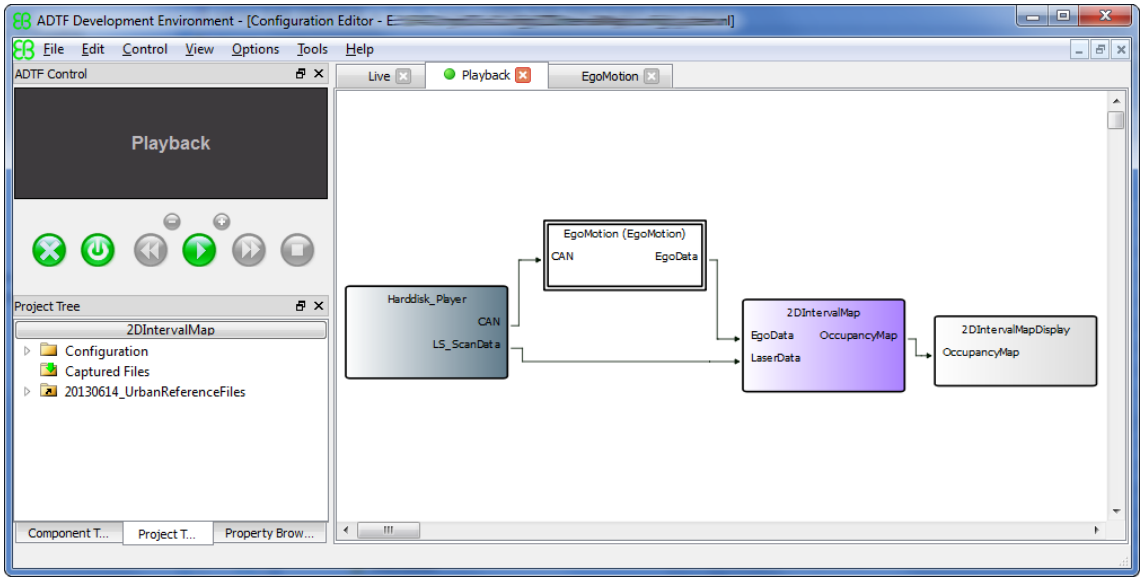


Figure C.1.: Illustration of filters and pin connections in ADTF graphical user interface.

filters can exchange information by transmitting and receiving data packets via pins, as also illustrated in the screenshot of the development environment in figure C.1. Besides that, filters can also be triggered by timers and may additionally provide interfaces for procedure calls.

One major advantage of using ADTF for ADAS software development is the simplified transition between online and offline operation. For the online mode, ADTF offers a wide range of I/O devices, which allow for accessing sensors and vehicle bus systems. The provided data can either be used to record a file or to directly operate the developed filters. Consequently, software modules can be implemented offline and tested online in a vehicle without any further migration effort. For more details about the applied framework, the interested reader is referred to [172].

D. Additional Experimental Results

The last section of the appendix offers descriptions of the used reference test scenarios and additional experimental results.

D.1. Occupancy Map Test Scenarios

The reference scenarios that have been used to evaluate and compare new and state-of-the-art occupancy representations are shown in figure D.1. Scenario S1a aims to simulate an ending lane due to a scene of accident in a dense traffic highway scenario. In order to evaluate the ability of the developed representations to deal with ego vehicle rotations, scenario S1b extends scenario S1a by a curved approach to the traffic lanes. Scenario S1c additionally includes an evasive maneuver when the vehicle reaches the scene of accident, which allows to test representations' behavior during dynamic ego movements.

Figure D.1.: Test scenarios for occupancy map evaluations.

D.2. Convoy Track Test Scenarios

The reference scenarios which have been used to evaluate the 2DIM with collective motion information are illustrated in figure D.2 and additionally described in the following listing:

- S2a: Three vehicles in front of the ego vehicle are driving in a straight convoy without lateral offset.
- S2b: Three vehicles in front of the ego vehicle are driving in a convoy and equally evade an obstacle.
- S2c: Three vehicles in front of the ego vehicle are driving in a convoy with a small lateral offset in order to simulate realistic dense traffic scenario.
- S2d: Two vehicles in front of the ego vehicle are driving in a convoy without lateral offset, another vehicle is driving in a neighboring lane.
- S2e: Two vehicles in front of the vehicle are permanently driving in a convoy, whereas another vehicle continuously changes the lane in order to enter and leave the convoy.
- S2f: Three vehicles in front of the ego vehicle are driving in a convoy without lateral offset on a circular path.

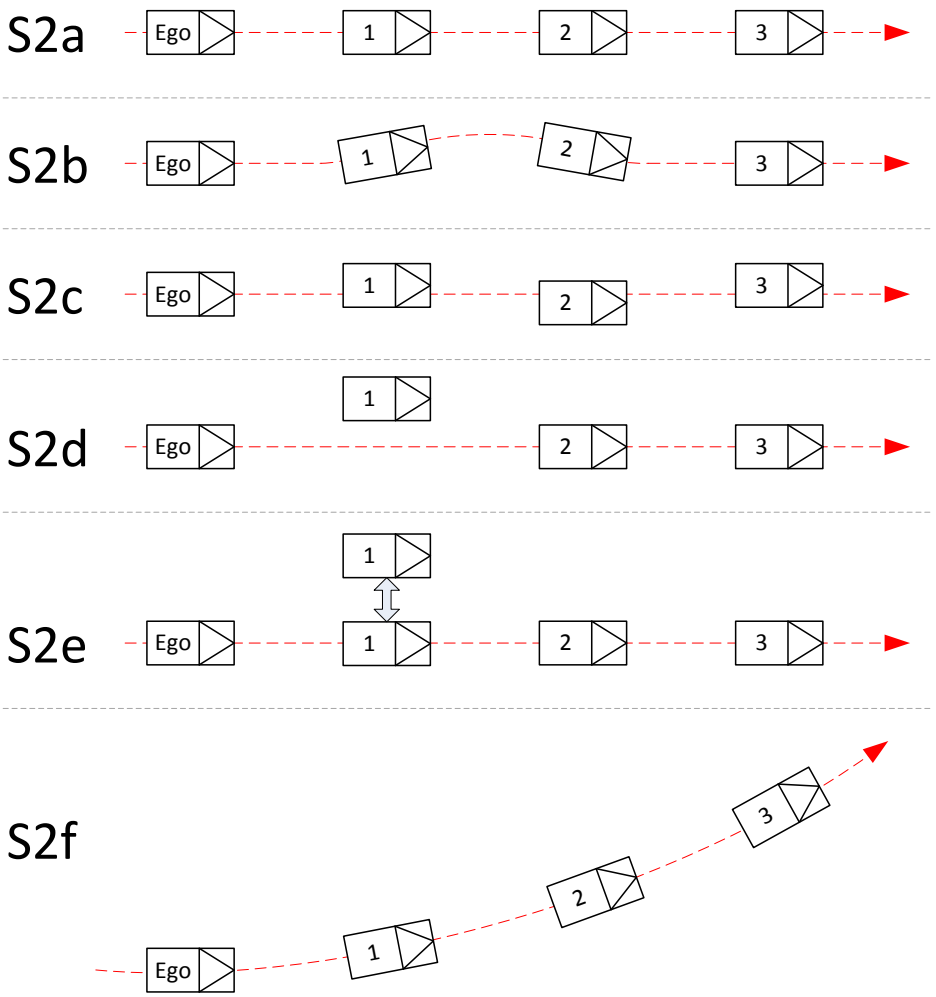


Figure D.2.: Test scenarios for convoy track evaluations.

D.3. Additional Convoy Track Evaluation Results

This section provides further sensor data analysis plots and evaluation metric tables concerning the mapping of common object motions.

D.3.1. Additional Evaluation Metric Tables

Table D.1 summarizes the obtained metric value differences between using the reference and standard ego motion module in the test scenarios S2a and S2f.

Scenario	Ego Motion Estimation	F1	Prec.	Rec.	CE [m]	BE [m]	AE [°]	VE [m/s]
S2a	Standard	0.91	0.99	0.84	0.28	0.40	2.32	0.47
S2a	Reference	0.91	0.99	0.83	0.28	0.41	2.25	0.49
S2f	Standard	0.79	0.76	0.81	0.22	0.46	2.58	0.36
S2f	Reference	0.75	0.73	0.77	0.65	0.78	2.84	0.43

Table D.1.: Comparison of metric values resulting from different ego motion estimation approaches.
(F1 = F1-Score, Prec. = Precision, Rec. = Recall, CE = Cell Center Error, BE = Cell Border Error, AE = Cell Angle Error, VE = Cell Velocity Error, CT = Computation Time, MR = Memory Requirement)

Table D.2 illustrates the measured metric values concerning the application of correlated and decorrelated input data.

Input Data	F1	Prec.	Rec.	CE [m]	BE [m]	AE [°]	VE [m/s]	CT [ms]
Tracked Objects	0.91	0.99	0.84	0.28	0.41	2.31	0.49	0.12
Equivalent Measurements	0.89	0.96	0.82	0.26	0.39	2.55	0.49	0.14

Table D.2.: Comparison of average metric values with different input data in scenario S2e.

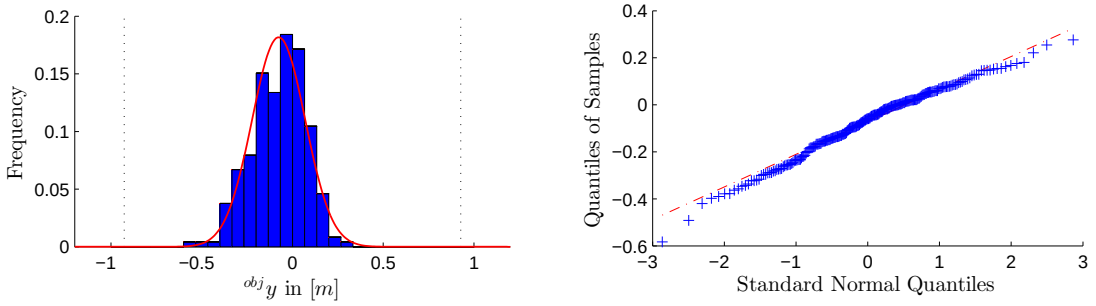
Finally, table D.3 shows the differences between using the point and line approach for the modeling of the motion cells' longitudinal extent.

Compensation Model	F1	Prec.	Rec.	CE [m]	AE [°]	VE [m/s]	CT [ms]	MR [kByte]
Point	0.78	0.82	0.74	0.67	3.16	0.49	0.06	2.1
Line	0.81	0.86	0.77	0.64	3.11	0.46	0.08	2.6

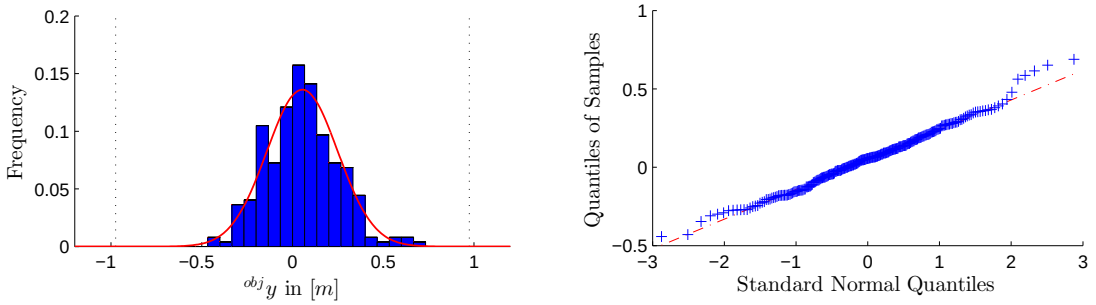
Table D.3.: Comparison of average metric values with different compensation strategies in scenario S2f.

D.3.2. Sensor Data Analysis

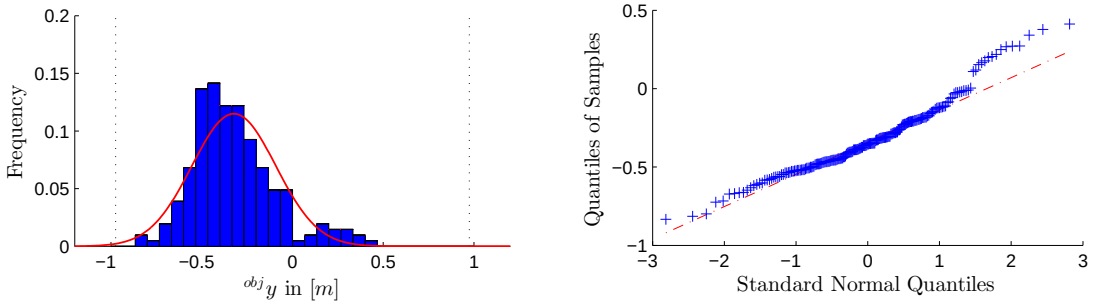
In addition to the graphs presented in section 4.3.1, figure D.3 provides histograms and Quantile-Quantile-Plots for the lateral distribution of radar reflection point locations in different occlusion scenarios. The Quantile-Quantile-Plots illustrate that the assumption of a normally distributed lateral reflection point location cannot be maintained in partial occlusion scenarios.



(a) Unoccluded vehicle



(b) Completely occluded vehicle



(c) Partially occluded vehicle with direct line of sight

Figure D.3.: Histograms and Quantile-Quantile-Plot of lateral radar reflection point locations on a vehicle's rear in different occlusion scenarios. The dotted lines in the histograms indicate the lateral borders of the vehicle.

Bibliography

[1] Aeberhard, M., Rauch, A., Rabiega, M., Kaempchen, N., and Bertram, T. Track-to-track fusion with asynchronous sensors and out-of-sequence tracks using information matrix fusion for advanced driver assistance systems. In *2012 IEEE Intelligent Vehicles Symposium (IV)* (Alcala de Henares, Madrid, Spain, 2012), pp. 1–6.

[2] America's Independent Electric Light and Power Companies. Power companies build for your new electric living. *LIFE Magazine 40*, 5 (1956), 8.

[3] Andert, F., and Goormann, L. Combined grid and feature-based occupancy map building in large outdoor environments. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (San Diego, CA, USA, 2007), pp. 2065–2070.

[4] Andrade-cetto, J., and Sanfeliu, A. Concurrent map building and localization on indoor dynamic environments. *International Journal on Pattern Recognition and Artificial Intelligence 16*, 3 (2002), 2002.

[5] Aue, J., Schmid, M. R., Graf, T., and Effertz, J. Improved Object Tracking from Detailed Shape Estimation Using Object Local Grid Maps with Stereo. In *2013 16th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (The Hague, The Netherlands, 2013), pp. 330–335.

[6] Ayache, N., and Faugeras, O. D. Maintaining representations of the environment of a mobile robot. *IEEE Transactions on Robotics and Automation 5*, 6 (1989), 804–819.

[7] Badino, H., Franke, U., and Pfeiffer, D. The Stixel World - A Compact Medium Level Representation of the 3D-World. In *Pattern Recognition*, D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, J. Denzler, G. Notni, and H. Süße, Eds., vol. 5748 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, Berlin and Heidelberg, 2009, pp. 51–60.

[8] Baer, M., Bouzouraa, M. E., Demiral, C., Hofmann, U., Gies, S., and Diepold, K. EgoMaster: A central ego motion estimation for driver assist systems. In *2009 IEEE International Conference on Control and Automation (ICCA)* (2009), pp. 1708–1715.

[9] Baig, Q., Aycard, O., Vu, T. D., and Fraichard, T. Fusion between laser and stereo vision data for moving objects tracking in intersection like scenario. In *2011 IEEE Intelligent Vehicles Symposium (IV)* (Baden-Baden, Germany, 2011), pp. 362–367.

[10] Balaguer, B., Balakirsky, S., Carpin, S., and Visser, A. Evaluating maps produced by urban search and rescue robots: lessons learned from RoboCup. *Autonomous Robots 27*, 4 (2009), 449–464.

[11] Balaguer, B., Carpin, S., Balakirsky, S., and Visser, A. Evaluation of RoboCup maps. In *Proceedings of the 9th Workshop on Performance Metrics for Intelligent Systems (PerMIS '09)* (Gaithersburg, Maryland, 2009), pp. 217–222.

[12] Bashir, F., and Porikli, F. Performance Evaluation of Object Detection and Tracking Systems. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS2006)* (2006).

[13] Bernardin, K., and Stiefelhagen, R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. *EURASIP Journal on Image and Video Processing 2008* (2008), 1–10.

[14] Beutelspacher, A., and Rosenbaum, U. *Projektive Geometrie*, 2. ed. Vieweg-Studium. Vieweg, Wiesbaden, 2004.

[15] Bierman, G. J. *Factorization methods for discrete sequential estimation.* Dover Publications, Mineola, 2006.

[16] Bouzouraa, M. E. *Belegungskartenbasierte Umfeldwahrnehmung in Kombination mit objektbasierten Ansätzen für Fahrerassistenzsysteme.* Doctoral thesis, Technische Universität München, München, 2012.

[17] Bouzouraa, M. E. Modellbasierte Freiraumverfolgung für Fahrerassistenzsysteme. In *8. Workshop Fahrerassistenzsysteme* (Walting, 2012), pp. 67–75.

[18] Bouzouraa, M. E., and Hofmann, U. Fusion of occupancy grid mapping and model based object tracking for driver assistance systems using laser and radar sensors. In *2010 IEEE Intelligent Vehicles Symposium (IV)* (San Diego, CA, USA, 2010), pp. 294–300.

[19] Brahmi, M. *Bewertung der FAS-Umfeldwahrnehmung mit Hilfe von Referenzsystemen (To be published).* Doctoral thesis, Technische Universität Carolo-Wilhelmina zu Braunschweig, Braunschweig, 2015.

[20] Bungartz, H.-J., Griebel, M., and Zenger, C. *Einführung in die Computergraphik: Grundlagen, geometrische Modellierung, Algorithmen*, 2. ed. Vieweg, Braunschweig, 2002.

[21] Burgard, W., Brock, O., and Stachniss, C., Eds. *Robotics: Science and systems III.* MIT Press, Cambridge, 2008.

[22] Chang, K. C., Saha, R. K., and Bar-Shalom, Y. On optimal track-to-track fusion. *IEEE Transactions on Aerospace and Electronic Systems 33*, 4 (1997), 1271–1276.

[23] Chin, W. C., Lee, H. B., Xiao, X. H., Ng, G. W., How, K. Y., and Dasarathy, B. V. A group tracking algorithm for ground vehicle convoys. In *Defense and Security* (Orlando, FL, 2005), SPIE Proceedings, SPIE, pp. 150–161.

[24] Chong, C.-Y., Mori, S., Barker, W. H., and Chang, K.-C. Architectures and algorithms for track association and fusion. *IEEE Aerospace and Electronic Systems Magazine 15*, 1 (2000), 5–13.

[25] Christensen, H. I., and Hager, G. D. Sensing and Estimation. In *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer Berlin Heidelberg, Berlin and Heidelberg, 2008, pp. 87–107.

[26] Cohen, O., Edan, Y., and Schechtman, E. Statistical Evaluation Method for Comparing Grid Map Based Sensor Fusion Algorithms. *The International Journal of Robotics Research 25*, 2 (2006), 117–133.

[27] Collins, T., Collins, J. J., and Ryan, C. Occupancy grid mapping: An empirical evaluation. In *2007 Mediterranean Conference on Control & Automation (MED)* (Athens, Greece, 2007), pp. 1–6.

[28] Coué, C., Pradalier, C., Laugier, C., Fraichard, T., and Bessiere, P. Bayesian Occupancy Filtering for Multitarget Tracking: an Automotive Application. *International Journal of Robotics Research 25*, 1 (2006), 19–30.

[29] Daimler AG. Pionierleistung: Autonome Langstreckenfahrt im Überland- und Stadtverkehr : Mercedes-Benz S-Klasse „INTELLIGENT DRIVE“ fährt autonom auf den Spuren von Bertha Benz, 10.09.2013. Press Release. Online available under URL `http://media.daimler.com/Projects/c2c/channel/documents/2370749_PM_MB___smart_Media_Night_IAA2013_de.pdf`. Last visit: Feb 2, 2014.

[30] Darms, M., Komar, M., and Lueke, S. Map based road boundary estimation. In *2010 IEEE Intelligent Vehicles Symposium (IV)* (San Diego, CA, USA, 2010), pp. 609–614.

[31] Dempster, A. P. Upper and Lower Probabilities Induced by a Multivalued Mapping. *The Annals of Mathematical Statistics 38*, 2 (1967), 325–339.

[32] Denzler, J., and Niemann, H. Active Rays: Polar-transformed Active Contours for Real-Time Contour Tracking. *Real-Time Imaging 5* (1997), 203–213.

[33] Dickmanns, E. D. Vision: Von Assistenz zum Autonomen Fahren. In *Fahrerassistenzsysteme mit maschineller Wahrnehmung*, M. Maurer and C. Stiller, Eds. Springer-Verlag, Berlin and Heidelberg, 2005, pp. 203–237.

[34] Dickmanns, E. D. *Dynamic vision for perception and control of motion.* Springer, London, 2007.

[35] Dietmayer, K., Kirchner, A., and Kämpchen, N. Fusionsarchitekturen zur Umfeldwahrnehmung für zukünftige Fahrerassistenzsysteme. In *Fahrerassistenzsysteme mit maschineller Wahrnehmung*, M. Maurer and C. Stiller, Eds. Springer-Verlag, Berlin and Heidelberg, 2005, pp. 59–88.

[36] Donner, E., Winkle, T., Walz, R., and Schwarz, J. RESPONSE3 - Code of Practice für die Entwicklung, Validierung und Markteinführung von künftigen Fahrerassistenzsystemen (ADAS). Technischer Kongress, VDA, Sindelfingen, 2007.

[37] Drummond, O. E. Feedback in track fusion without process noise. In *Signal and Data Processing of Small Targets 1995* (San Diego, CA, USA, 1995), SPIE Proceedings 2561, SPIE, pp. 369–383.

[38] Drummond, O. E. Track fusion with feedback. In *Signal and Data Processing of Small Targets 1996* (Orlando, FL, 1996), SPIE Proceedings 2759, SPIE, pp. 342–360.

[39] Drummond, O. E. Tracklets and a hybrid fusion with process noise. In *Signal and Data Processing of Small Targets 1997* (San Diego, CA, USA, 1997), SPIE Proceedings 3163, SPIE, pp. 512–524.

[40] Drummond, O. E., and Dana-Bashian, D. Comparison of tracklet filter methods for non-maneuvering targets. In *Signal and Data Processing of Small Targets 2005* (San Diego, CA, USA, 2005), SPIE Proceedings 5913, SPIE, pp. 59131B–59131B–13.

[41] Dryanovski, I., Morris, W., and Jizhong Xiao. Multi-volume occupancy grids: An efficient probabilistic 3D mapping model for micro aerial vehicles. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Taipei, 2010), pp. 1553–1559.

[42] Effertz, J. *Autonome Fahrzeugführung in urbaner Umgebung durch Kombination objekt- und kartenbasierter Umfeldmodelle.* Doctoral thesis, Technische Universität Carolo-Wilhelmina zu Braunschweig, Braunschweig, 2009.

[43] Einhorn, E., Schröter, C., and Gross, H.-M. Building 2D and 3D Adaptive-Resolution Occupancy Maps using ND-Trees. In *Crossing borders within the ABC* (2010), P. Scharff, Ed., Verl. ISLE.

[44] Einhorn, E., Schroter, C., and Gross, H.-M. Finding the adequate resolution for grid mapping - Cell sizes locally adapting on-the-fly. In *2011 IEEE International Conference on Robotics and Automation (ICRA)* (Shanghai, China, 2011), pp. 1843–1848.

[45] Elfes, A. A sonar-based mapping and navigation system. In *1986 IEEE International Conference on Robotics and Automation (ICRA)* (San Francisco, CA, USA, 1986), pp. 1151–1156.

[46] Elfes, A., and Matthies, L. Sensor integration for robot navigation: Combining sonar and stereo range data in a grid-based representataion. In *26th IEEE Conference on Decision and Control* (1987), pp. 1802–1807.

[47] Farrell, J., and Barth, M. *The global positioning system and inertial navigation.* McGraw-Hill, New York, 1999.

[48] Feldmann, M., and Koch, W. Road-map assisted convoy track maintenance using random matrices. In *2008 11th International Conference on Information Fusion* (2008), pp. 1–8.

[49] Ferguson, D., Darms, M., Urmson, C., and Kolski, S. Detection, prediction, and avoidance of dynamic obstacles in urban environments. In *2008 IEEE Intelligent Vehicles Symposium (IV)* (Eindhoven, Netherlands, 2008), pp. 1149–1154.

[50] Fisher, A. Inside Google's Quest To Popularize Self-Driving Cars. *Popular Science*, October (2013).

[51] Foessel, A., Bares, J., and William L. Whittaker. Three-Dimensional Map Building with MMW Radar. In *Proceedings of the 3rd International Conference on Field and Service Robotics* (Helsinki, Finland, 2001), A. Halme, R. Chatila and E. Prassler, Eds.

[52] Frenkel, G. Flexible Architecture for Sensor Fusion in Theater Missile Defense, 1994. Technical Report IDA P-2935, Institute for Defense Analysis, Alexandria, Virginia.

[53] Frenkel, G., and Drummond, O. E. Multisensor tracking of ballistic targets. In *Signal and Data Processing of Small Targets 1995* (San Diego, CA, USA, 1995), SPIE Proceedings 2561, SPIE, pp. 337–346.

[54] Fürstenberg, K. C. *Fahrzeugumfelderfassung und Fußgängerschutz unter Nutzung mehrzeiliger Laserscanner.* Doctoral thesis, Universität Ulm. Fakultät für Ingenieurwissenschaften und Informatik, 2009.

[55] Gackstatter, C., Heinemann, P., Thomas, S., Rosenhahn, B., and Klinker, G. Fusion of clothoid segments for a more accurate and updated prediction of the road geometry. In *2010 13th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (Funchal, Madeira Island, Portugal, 2010), pp. 1691–1696.

[56] Gasser, T. M. *Rechtsfolgen zunehmender Fahrzeugautomatisierung: Gemeinsamer Schlussbericht der Projektgruppe, Bericht zum Forschungsprojekt F 1100.5409013.01*, vol. 83 of *Berichte der Bundesanstalt für Straßenwesen Fahrzeugtechnik (F).* Wirtschaftsverl. NW, Bremerhaven, 2012.

[57] Gavrila, D. M., Franke, U., Wohler, C., and Gorzig, S. Real time vision for intelligent vehicles. *IEEE Instrumentation & Measurement Magazine 4*, 2 (2001), 22–27.

[58] Geduld, G. Lidarsensorik. In *Handbuch Fahrerassistenzsysteme*, H. Winner, S. Hakuli, and G. Wolf, Eds. Vieweg+Teubner Verlag, Wiesbaden, 2012, pp. 172–185.

[59] Giesler, B., Müller, T., Hofmann, U., and Siedersberger, K. Opportunities and Challenges on the Route to Piloted Driving. In *chassis.tech plus 2013 - 4. Internationales Münchner Fahrwerk-Symposium* (München, 2013).

[60] Goldstein, H., Poole, C. P., and Safko, J. L. *Classical mechanics*, 3rd ed. Addison Wesley, San Francisco, 2002.

[61] Grewe, R., Hohm, A., Hegemann, S., Lueke, S., and Winner, H. Towards a generic and efficient environment model for ADAS. In *2012 IEEE Intelligent Vehicles Symposium (IV)* (Alcala de Henares, Madrid, Spain, 2012), pp. 316–321.

[62] Grewe, R., Hohm, A., Lüke, S., and Winner, H. Umfeldmodelle - Standardisierte Schnittstellen für Assistenzsysteme. *ATZelektronik 7*, 5 (2012), 334–339.

[63] Grewe, R., Komar, M., Hohm, A., Lueke, S., and Winner, H. Evaluation method and results for the accuracy of an automotive occupancy grid. In *2012 IEEE International Conference on Vehicular Electronics and Safety (ICVES)* (Istanbul, Turkey, 2012), pp. 19–24.

[64] Heinrichs-Bartscher, S. System zur Beeinflussung der Geschwindigkeit eines Kraftfahrzeugs, 2004. Deutsches Patent- und Markenamt, DE10254402B4.

[65] Himmelsbach, M., Mueller, A., Luettel, T., and Wuensche, H.-J. LIDAR-based 3D Object Perception. In *Proceedings of 1st International Workshop on Cognition for Technical Systems* (2008).

[66] Hofmann, U. *Zur visuellen Umfeldwahrnehmung autonomer Fahrzeuge*. Doctoral thesis, Universität der Bundeswehr München, München, 2004.

[67] Hofmann, U., and Bouzouraa, M. E. Verfahren zur Auswertung von die Umgebung eines Kraftfahrzeugs betreffenden Sensordaten wenigstens eines Umfeldsensors und Kraftfahrzeug, 2011. Deutsches Patent- und Markenamt, DE102010005501A1.

[68] Hohm, A., Wojek, C., Schiele, B., and Winner, H. Multi-Level Sensorfusion and Computer-Vision Algorithms within a Driver Assistance System for Avoiding Overtaking-Accidents. In *FISITA 2008 World Automotive Congress* (München, 2008), VDI-FVT, Ed., pp. 1–14.

[69] Homeier, K., and Wolf, L. RoadGraph: High level sensor data fusion between objects and street network. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (Washington, DC, USA, 2011), pp. 1380–1385.

[70] Homm, F., Kaempchen, N., and Burschka, D. Fusion of laserscannner and video based lanemarking detection for robust lateral vehicle control and lane change

maneuvers. In *2011 IEEE Intelligent Vehicles Symposium (IV)* (Baden-Baden, Germany, 2011), pp. 969–974.

[71] Homm, F., Kaempchen, N., Ota, J., and Burschka, D. Efficient occupancy grid computation on the GPU with lidar and radar for road boundary detection. In *2010 IEEE Intelligent Vehicles Symposium (IV)* (San Diego, CA, 2010), pp. 1006–1013.

[72] Hong, T. H., Abrams, M., Chang, T., and Shneier, M. An Intelligent World Model for Autonomous Off-Road Driving. *Computer Vision and Image Understanding, Hook, Aster Spectral Library* (2000).

[73] Hundelshausen, F. v., Himmelsbach, M., Hecker, F., Mueller, A., and Wuensche, H.-J. Driving with Tentacles - Integral Structures for Sensing and Motion. In *The DARPA Urban Challenge*, B. Siciliano, O. Khatib, F. Groen, M. Buehler, K. Iagnemma, and S. Singh, Eds., vol. 56 of *Springer Tracts in Advanced Robotics*. Springer, Berlin and Heidelberg, 2009, pp. 393–440.

[74] ISRD Group. *Computer Graphics*. Tata McGraw-Hill Education, Noida, 2005.

[75] Kalman, R. E. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME–Journal of Basic Engineering 82*, Series D (1960), 35–45.

[76] Kalman, R. E., and Bucy, R. S. New results in linear filtering and prediction theory. *Transactions of the ASME. Series D, Journal of Basic Engineering 83* (1961), 95–107.

[77] Kammel, S. Autonomes Fahren. In *Handbuch Fahrerassistenzsysteme*, H. Winner, S. Hakuli, and G. Wolf, Eds. Vieweg+Teubner Verlag, Wiesbaden, 2012, pp. 651–657.

[78] Kämpchen, N. *Feature-level fusion of laser scanner and video data for advanced driver assistance systems*. Doctoral thesis, Universität Ulm. Fakultät für Ingenieurwissenschaften und Informatik, 2007.

[79] Kang, J., and Chung, M. J. Stereo-vision based free space and obstacle detection with structural and traversability analysis using probabilistic volume polar grid map. In *2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM)* (Qingdao, China, 2011), pp. 245–251.

[80] Kapach, K., and Edan, Y. Evaluation of grid-map sensor fusion mapping algorithms. In *2007 IEEE International Conference on Systems, Man and Cybernetics (ISIC)* (Montreal, QC, Canada, 2007), pp. 829–834.

[81] Kasturi, R., Goldgof, D., Soundararajan, P., Manohar, V., Garofolo, J., Bowers, R., Boonstra, M., Korzhova, V., and Jing Zhang. Framework for Performance Evaluation of Face, Text, and Vehicle Detection and Tracking in

Video: Data, Metrics, and Protocol. *IEEE Transactions on Pattern Analysis and Machine Intelligence 31*, 2 (2009), 319–336.

[82] Kellner, M., Bouzouraa, M. E., and Hofmann, U. Road Curb Detection Based on Different Elevation Mapping Techniques. In *2014 IEEE Intelligent Vehicles Symposium (IV)* (Dearborn, Michigan, USA, 2014), pp. 1217 – 1224.

[83] Kellner, M., Hofmann, U., Bouzouraa, M. E., Kasper, H., and Neumaier, S. Laserscanner based road curb feature detection and efficient mapping using local curb descriptions. In *2014 17th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (Qingdao, China, 2014), pp. 2602 – 2609.

[84] Knaup, J., and Homeier, K. RoadGraph - Graph based environmental modelling and function independent situation analysis for driver assistance systems. In *2010 13th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (Funchal, Madeira Island, Portugal, 2010), pp. 428–432.

[85] Knuth, D. E. *The Art of Computer Programming: Volume 1: Fundamental Algorithms*, 3rd ed. Addison-Wesley, Boston, 1997.

[86] Kompass, K., Gruber, C., and Domsch, C. Der Beitrag von Fahrerassistenzsystemen zur Aktiven und Passiven Sicherheit – die Integrale Sicherheit als Antwort auf die wachsenden Anforderungen an die Fahrzeugsicherheit. In *4. Tagung Fahrerassistenz* (München, 2010).

[87] Konrad, M., and Dietmayer, K. Occupancy Grid Mapping using the Dempster-Shafer-Theory. In *Proceedings of the International Workshop on Intelligent Transportation (WIT)* (2011), pp. 167–172.

[88] Konrad, M., Nuss, D., and Dietmayer, K. Localization in digital maps for road course estimation using grid maps. In *2012 IEEE Intelligent Vehicles Symposium (IV)* (Alcala de Henares, Madrid, Spain, 2012), pp. 87–92.

[89] Konrad, M., Szczot, M., Schule, F., and Dietmayer, K. Generic grid mapping for road course estimation. In *2011 IEEE Intelligent Vehicles Symposium (IV)* (Baden-Baden, Germany, 2011), pp. 851–856.

[90] Kraetzschmar, G. K., Gassull, G. P., and Uhl, K. Probabilistic Quadtrees for Variable-Resolution Mapping of Large Environments. In *Proceedings of the 5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles* (Lisbon, Portugal, 2004), M. I. Ribeiro and S. J. Victor, Eds., Elsevier Science.

[91] Kühlmeyer, M. *Statistische Auswertungsmethoden für Ingenieure: Mit Praxisbeispielen.* VDI-Buch. Springer, Berlin, 2001.

[92] Kuhn, H. W., and Heynen-Imhof, L. *The Hungarian method for the assignment problem*, vol. 2. Naval Research Logistics, 1955.

[93] Kühnl, T., Kummert, F., and Fritsch, J. Monocular road segmentation using slow feature analysis. In *2011 IEEE Intelligent Vehicles Symposium (IV)* (Baden-Baden, Germany, 2011), pp. 800–806.

[94] Leal, J., Scheding, S., and Dissanayake, G. Probabilistic 2D mapping in unstructured environments. In *Proceedings of the Australian Conference on Robotics and Automation* (1998), Kluwer Academic Publishers, pp. 29–53.

[95] Lucas, B., Held, R., Freundt, D., Klar, M., and Maurer, M. Frontsensorsystem mit Doppel Long Range Radar. In *5. Workshop Fahrerassistenzsysteme 2008* (Walting, 2008).

[96] Lundquist, C., and Schön, T. B. Estimation of the Free Space in Front of a Moving Vehicle, 2009. Technical Report LiTH-ISY-R-289, Division of Automatic Control, Linköpings Universitet, Sweden.

[97] Lutz, L. S., Tang, T., and Lienkamp, M. Die rechtliche Situation von teleoperierten und autonomen Fahrzeugen. *Neue Zeitschrift für Verkehrsrecht NZV 26*, 2 (2013), 57–63.

[98] Mählisch, M. *Filtersynthese zur simultanen Minimierung von Existenz-, Assoziations- und Zustandsunsicherheiten in der Fahrzeugumfelderfassung mit heterogenen Sensordaten.* Doctoral thesis, Universität Ulm. Fakultät für Ingenieurwissenschaften und Informatik, 2009.

[99] Martin C. Martin, and Moravec, H. P. Robot evidence Grids, 1996. Technical Report CMU-RI-TR-96-06, Carnegie Mellon University, Robotics Institute, Pittsburgh.

[100] Mason, J., Ricco, S., and Parr, R. Textured Occupancy Grids for Monocular Localization Without Features. In *2011 IEEE International Conference on Robotics and Automation (ICRA)* (Shanghai, China, 2011).

[101] Matthaei, R., Dyckmanns, H., Maurer, M., and Lichte, B. Consistency-based motion classification for laser sensors dealing with cross traffic in urban environments. In *2011 IEEE Intelligent Vehicles Symposium (IV)* (Baden-Baden, Germany, 2011), pp. 1136–1141.

[102] Maurer, M. *Flexible Automatisierung von Straßenfahrzeugen mit Rechnersehen.* Doctoral thesis, Universität der Bundeswehr München, München, 2000.

[103] Maurer, M. Entwurf und Test von Fahrerassistenzsystemen. In *Handbuch Fahrerassistenzsysteme*, H. Winner, S. Hakuli, and G. Wolf, Eds. Vieweg+Teubner Verlag, Wiesbaden, 2012, pp. 43–54.

[104] Mesow, L. *Multisensorielle Datensimulation im Fahrzeugumfeld für die Bewertung von Sensorfusionsalgorithmen.* Doctoral thesis, Technische Universität Chemnitz, Chemnitz, 2007.

[105] Mitschke, M. *Dynamik von Kraftfahrzeugen*, 4th ed. Springer, Berlin, 2003.

[106] Mogelmose, A., Trivedi, M. M., and Moeslund, T. B. Vision-Based Traffic Sign Detection and Analysis for Intelligent Driver Assistance Systems: Perspectives and Survey. *IEEE Transactions on Intelligent Transportation Systems 13*, 4 (2012), 1484–1497.

[107] Montemerlo, M., and Thrun, S. A multi-resolution pyramid for outdoor robot terrain perception. In *Proceedings of the 19th national conference on Artifical intelligence* (2004), AAAI Press, pp. 464–469.

[108] Moravec, H. P. Robot Spatial Perception by Stereoscopic Vision and 3D Evidence Grids, 1996. Technical Report CMU-RI-TR-96-34, Carnegie Mellon University, Robotics Institute, Pittsburgh.

[109] Moravec, H. P., and Elfes, A. High resolution maps from wide angle sonar. In *1985 IEEE International Conference on Robotics and Automation (ICRA)* (St. Louis, MO, USA, 1985), pp. 116–121.

[110] Mullane, J., Adams, M. D., and Wijesoma, W. S. Evidential versus Bayesian Estimation for Radar Map Building. In *2006 9th International Conference on Control, Automation, Robotics and Vision* (Singapore, 2006), pp. 1–8.

[111] Naab, K., and Reichart, G. Grundlagen der Fahrerassistenzsysteme und Anforderungen aus Nutzersicht. In *Seminar Fahrerassistenzsysteme* (Haus der Technik, Essen, 2008).

[112] Neumaier, S. *Räumlich-zeitliche Szeneninterpretation einer videobasierten Umfelderfassung zur Fahrerassistenz*. Doctoral thesis, Technische Universität München, München, 2013.

[113] Nguyen, T.-N., Meinecke, M.-M., Tornow, M., and Michaelis, B. Optimized grid-based environment perception in advanced driver assistance systems. In *2009 IEEE Intelligent Vehicles Symposium (IV)* (Xi'an, China, 2009), pp. 425–430.

[114] Nuss, D., Reuter, S., Konrad, M., Munz, M., and Dietmayer, K. Using grid maps to reduce the number of false positive measurements in advanced driver assistance systems. In *2012 15th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (Anchorage, AK, USA, 2012), pp. 1509–1514.

[115] Nuss, D., Wilking, B., Wiest, J., Deusch, H., Reuter, S., and Dietmayer, K. Decision-Free True Positive Estimation with Grid Maps for Multi-Object Tracking. In *2013 16th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (The Hague, The Netherlands, 2013).

[116] Oniga, F., Nedevschi, S., Meinecke, M. M., and To, T. B. Road Surface and Obstacle Detection Based on Elevation Maps from Dense Stereo. In *2007 IEEE Intelligent Transportation Systems Conference (ITSC)* (Bellevue, WA, USA, 2007), pp. 859–865.

[117] Oriolo, G., Ulivi, G., and Vendittelli, M. Fuzzy Maps: A New Tool for Mobile Robot Perception and Planning. *Journal of Robotic Systems 14* (1997), 179–197.

[118] Pancham, A., Tlale, N., and Bright, G. Literature review of SLAM and DATMO. In *4th Robotics and Mechatronics Conference of South Africa, RobMech 2011* (Pretoria, 2011), vol. 7484.

[119] Pandey, A. K., Krishna, K. M., and Nath, M. Feature based occupancy grid maps for sonar based safe-mapping. In *Proceedings of the 20th international joint conference on Artifical intelligence* (San Francisco, CA, USA, 2007), IJCAI'07, Morgan Kaufmann Publishers Inc, pp. 2172–2177.

[120] Paskin, M., and Thrun, S. Robotic Mapping with Polygonal Random Fields. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence (UAI)* (2005).

[121] Pathak, K., Birk, A., Poppinga, J., and Schwertfeger, S. 3D forward sensor modeling and application to occupancy grid based sensor fusion. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (San Diego, CA, USA, 2007), pp. 2059–2064.

[122] Payeur, P., Hebert, P., Laurendeau, D., and Gosselin, C. M. Probabilistic octree modeling of a 3D dynamic environment. In *1997 IEEE International Conference on Robotics and Automation (ICRA)* (Albuquerque, NM, USA, 1997), pp. 1289–1296.

[123] Peter, A., Bäker, W., and Ruchatz, T. Verfahren zur automatischen Abstandsregelung von Kraftfahrzeugen, 1999. Deutsches Patent- und Markenamt, DE19804944A1.

[124] Petrovskaya, A. *Towards Dependable Robotic Perception*. PhD thesis, Stanford University, Stanford, CA, 2011.

[125] Pfeiffer, D., and Franke, U. Efficient representation of traffic scenes by means of dynamic stixels. In *2010 IEEE Intelligent Vehicles Symposium (IV)* (San Diego, CA, 2010), pp. 217–224.

[126] Pfeiffer, D., and Franke, U. Towards a Global Optimal Multi-Layer Stixel Representation of Dense 3D Data. In *Proceedings of the British Machine Vision Conference* (2011), BMVA Press, pp. 51.1–51.12.

[127] Platho, M., Gross, H.-M., and Eggert, J. Traffic situation assessment by recognizing interrelated road users. In *2012 15th International IEEE Conference on Intelligent Transportation Systems ITSC* (Anchorage, AK, USA, 2012), pp. 1339–1344.

[128] Pollard, E., Pannetier, B., and Rombaut, M. Convoy detection processing by using the hybrid algorithm (GMCPHD/VS-IMMC-MHT) and Dynamic Bayesian

Networks. In *2009 12th International Conference on Information Fusion (FUSION)* (2009), pp. 907–914.

[129] Pollard, E., Pannetier, B., and Rombaut, M. Performances in multitarget tracking for convoy detection over real GMTI data. In *2010 13th Conference on Information Fusion (FUSION)* (2010), pp. 1–7.

[130] Powers, D. M. W. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness & correlation. *Journal of Machine Learning Technologies 2*, 1 (2011), 37–63.

[131] Raaijmakers, M., and Bouzouraa, M. E. Circle Detection in Single-Layer Laser Scans for Roundabout Perception. In *2014 17th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (Qingdao, China, 2014), pp. 2636 – 2643.

[132] Rasmussen, J. Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man, and Cybernetics SMC-13*, 3 (1983), 257–266.

[133] Reichel, M. *Situationsanalyse für fortschrittliche Fahrerassistenzsysteme*. Doctoral thesis, Technische Universität Carolo-Wilhelmina zu Braunschweig, Braunschweig, 2013.

[134] Reichel, M., Botsch, M., Rauschecker, R., Siedersberger, K., and Maurer, M. Situation aspect modelling and classification using the Scenario Based Random Forest algorithm for convoy merging situations. In *2010 13th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (Funchal, Madeira Island, Portugal, 2010), pp. 360–366.

[135] Reichel, M., Bouzouraa, M. E., Siegel, A., Siedersberger, K.-H., and Maurer, M. Erweiterte Umfelderkennung und Nutzung einer Ausweichanalyse als Grundlage einer aktiven Gefahrenbremsung. In *Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel (AAET)* (Braunschweig, 2010).

[136] Richter, E., Lindner, P., Wanielik, G., Takagi, K., and Isogai, A. Advanced occupancy grid techniques for lidar based object detection and tracking. In *2009 12th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (St. Louis, 2009), pp. 1–5.

[137] Runkler, T. A. *Data Mining: Methoden und Algorithmen intelligenter Datenanalyse*, 1. ed. Vieweg + Teubner, Wiesbaden, 2010.

[138] SAE International. Taxonomy and Definitions for Terms Related to On-Road Automated Motor Vehicles, j3016, 2013.

[139] Salaria, K., Darsono, W., Hinman, M., Linderman, M., Bai, L., and Dasarathy, B. V. Object aggregation using merge-at-a-point algorithm. In *Defense and Security* (Orlando, FL, 2004), SPIE Proceedings, SPIE, pp. 287–294.

[140] Scheunert, U., Mattern, N., Lindner, P., and Wanielik, G. Generalized Grid Framework for Multi Sensor Data Fusion. In *Proceedings of the International Conference on Information Fusion* (2008).

[141] Schmid, M. R. *Umgebungserfassung für Fahrerassistenzsysteme mit hierarchischen Belegungskarten.* Doctoral thesis, Universität der Bundeswehr München, München, 2012.

[142] Schmid, M. R., Ates, S., Dickmann, J., Hundelshausen, F. v., and Wuensche, H.-J. Parking space detection with hierarchical dynamic occupancy grids. In *2011 IEEE Intelligent Vehicles Symposium (IV)* (Baden-Baden, Germany, 2011), pp. 254–259.

[143] Schmid, M. R., Mählisch, M., Dickmann, J., and Wuensche, H. J. Dynamic level of detail 3D occupancy grids for automotive use. In *2010 IEEE Intelligent Vehicles Symposium (IV)* (San Diego, CA, 2010), pp. 269–274.

[144] Schreier, M., and Willert, V. Robust free space detection in occupancy grid maps by methods of image analysis and dynamic B-spline contour tracking. In *2012 15th International IEEE Conference on Intelligent Transportation Systems ITSC* (Anchorage, AK, USA, 2012), pp. 514–521.

[145] Schreier, M., Willert, V., and Adamy Jürgen. From grid maps to Parametric Free Space maps - A highly compact, generic environment representation for ADAS. In *2013 IEEE Intelligent Vehicles Symposium (IV)* (Gold Coast City, Australia, 2013), pp. 938–944.

[146] Seeliger, F., and Dietmayer, K. Fahrzeugübergreifende Informationsfusion. In *5. Tagung Fahrerassistenz* (München, 2012).

[147] Seetharaman, G., Lakhotia, A., and Blasch, E. P. Unmanned vehicles come of age: The DARPA grand challenge. *Computer 39*, 12 (2006), 26–29.

[148] Shafer, G. *A mathematical theory of evidence.* Princeton University Press, Princeton, 1976.

[149] Siedersberger, K.-H. *Komponenten zur automatischen Fahrzeugführung in sehenden (semi-)autonomen Fahrzeugen.* Doctoral thesis, Universität der Bundeswehr München, München, 2003.

[150] Smith, K., Gatica-Perez, D., Odobez, J., and Sileye Ba. Evaluating Multi-Object Tracking. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops* (San Diego, CA, USA, 2005), pp. 36 – 44.

[151] Smith, K., Schweiger, R., Ritter, W., and Kallhammer, J.-E. Development and evaluation of a performance metric for image-based driver assistance systems. In *2011 IEEE Intelligent Vehicles Symposium (IV)* (Baden-Baden, Germany, 2011), pp. 381–386.

[152] Soatto, S., Frezza, R., Perona, P., and Picci, G. Dynamic estimation of rigid motion from perspective views via recursive identification of exterior differential systems with parameters on a topological manifold, 1994. Technical Report CIT-CDS 94-004, Control and Dynamical Systems, California Institute of Technology.

[153] Spangenberg, R., and Döring, T. Evaluation of Object Tracking in Traffic Scenes. In *ISPRS Com. V Symposium 'Image Engineering and Vision Metrology'*, vol. XXXVI. IAPRS, Dresden, 2006.

[154] Spies, M., and Spies, H. Automobile Lidar Sensorik: Stand, Trends und zukünftige Herausforderungen. *Advances in Radio Science 4* (2006), 99–104.

[155] Stachniss, C., and Burgard, W. Mobile Robot Mapping and Localization in Non-Static Environments. In *Proceedings of the National Conference on Artificial Intelligence* (2005).

[156] Statistisches Bundesamt. Unfallentwicklung auf deutschen Straßen 2012, 2013/07/10. Online available under URL `https://www.destatis.de/DE/Publikationen/Thematisch/TransportVerkehr/Verkehrsunfaelle/PK_Unfallentwicklung_PDF.pdf?__blob=publicationFile`. Last visit Feb 2, 2014.

[157] Steger, A. *Diskrete Strukturen 1.* Springer-Lehrbuch. Springer-Verlag Berlin Heidelberg, New York, 2007.

[158] Stiller, C., Kammel, S., Lulcheva, I., and Ziegler, J. Probabilistic Methods for Environment Perception of Cognitive Automobiles. *at - Automatisierungstechnik 56*, 11 (2008).

[159] Strasser, B., Siegel, A., Siedersberger, K.-H., Maurer, M., and Bubb, H. Vernetzung von Test- und Simulationsmethoden für die Entwicklung von Fahrerassistenzsystemen. In *4. Tagung Fahrerassistenz* (München, 2010).

[160] Teizer, J., Bosché, F., Caldas, C. H., and Haas, C. T. Real-time Spatial Detection and Tracking of Resources in a Construction Environment. In *Proceedings of the ASCE/ISCCBE Joint International Conference on Computing and Decision Making in Civil and Building Engineering* (2006), pp. 494–502.

[161] Thrun, S. Bayesian Landmark Learning for Mobile Robot Localization. *Machine Learning 33*, 1 (1998), 41–76.

[162] Thrun, S. Robotic Mapping: A Survey. In *Exploring Artificial Intelligence in the New Millenium* (Burlington, Massachusetts, 2002), Morgan Kaufmann.

[163] Thrun, S. Learning Occupancy Grid Maps with Forward Sensor Models. *Autonomous Robots 15*, 2 (2003), 111–127.

[164] Thrun, S., Burgard, W., and Fox, D. A Real-Time Algorithm for Mobile Robot Mapping With Applications to Multi-Robot and 3D Mapping. In *2000 IEEE*

International Conference on Robotics and Automation (ICRA) (San Francisco, CA, 2000), IEEE.

[165] Thrun, S., Burgard, W., and Fox, D. *Probabilistic robotics*. Intelligent robotics and autonomous agents. MIT Press, Cambridge, 2005.

[166] TomTom International BV. TomTom European Congestion Index 2012, 2013. Online available under URL `http://www.tomtom.com/lib/doc/trafficindex/2013-0322-TomTom-CongestionIndex-2012-Annual-EUR-km.pdf`. Last Visit Feb 2, 2014.

[167] Triebel, R., Pfaff, P., and Burgard, W. Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Beijing, China, 2006), pp. 2276–2282.

[168] United Nations. Convention on Road Traffic: Chapter XI: Transport and Communications, 1968. Online available under URL `http://treaties.un.org/doc/Treaties/1977/05/19770524%2000-13%20AM/Ch_XI_B_19.pdf`. Last visit Feb 2, 2014.

[169] United Nations Economic Commission for Europe. Regulation R79 - Steering Equipment, 20.1.2006. Online available under URL `http://www.bmvi.de/SharedDocs/DE/Anlage/static/ECE/r-79-lenkanlagen-pdf.pdf?__blob=publicationFile`. Last visit: Feb 2, 2014.

[170] Urmson, C., Anhalt, J., Bae, H., J. Andrew Bagnell, Baker, C. R., Bittner, R. E., Brown, T., Clark, M. N., Darms, M., Demitrish, D., Dolan, J. M., Duggins, D., Ferguson, D., Galatali, T., Geyer, C. M., Gittleman, M., Harbaugh, S., Hebert, M., Howard, T., Kolski, S., Likhachev, M., Litkouhi, B., Kelly, A., McNaughton, M., Miller, N., Nickolaou, J., Peterson, K., Pilnick, B., Rajkumar, R., Rybski, P., Sadekar, V., Salesky, B., Seo, Y.-W., Singh, S., Snider, J. M., Struble, J. C., Stentz, A., Taylor, M., William L. Whittaker, Wolkowicki, Z., Zhang, W., Ziglar, J., and Martin Buehler, Karl Lagnemma, Sanjiv Singh. Autonomous driving in urban environments: Boss and the Urban Challenge. *Journal of Field Robotics Special Issue on the 2007 DARPA Urban Challenge, Part I 25*, 8 (2008), 425–466.

[171] Veeck, M., and Burgard, W. Learning polyline maps from range scan data acquired with mobile robots. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Sendai, Japan, 2004), pp. 1065–1070.

[172] Voigtländer, P. ADTF: Framework for Driver Assistance and Safety Systems. In *FISITA 2008 World Automotive Congress* (München, 2008), VDI-FVT, Ed.

[173] Vu, T.-D., and Aycard, O. Laser-based detection and tracking moving objects using data-driven Markov chain Monte Carlo. In *2009 IEEE International Conference on Robotics and Automation (ICRA)* (Kobe, 2009), pp. 3800–3806.

[174] Vu, T.-D., Aycard, O., and Appenrodt, N. Online Localization and Mapping with Moving Object Tracking in Dynamic Outdoor Environments. In *2007 IEEE Intelligent Vehicles Symposium (IV)* (Istanbul, Turkey, 2007), pp. 190–195.

[175] Vu, T.-D., Burlet, J., and Aycard, O. Grid-based localization and online mapping with moving objects detection and tracking: new results. In *2008 IEEE Intelligent Vehicles Symposium (IV)* (Eindhoven, Netherlands, 2008), pp. 684–689.

[176] Wang, C.-C., and Thorpe, C. A hierarchical object based representation for simultaneous localization and mapping. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Sendai, Japan, 2004), pp. 412–418.

[177] Wang, C.-C., Thorpe, C., and Thrun, S. Online simultaneous localization and mapping with detection and tracking of moving objects: theory and results from a ground vehicle in crowded urban areas. In *2003 IEEE International Conference on Robotics and Automation (ICRA)* (Taipei, Taiwan, 2003), pp. 842–849.

[178] Wang, C.-C., Thorpe, C., Thrun, S., Hebert, M., and Durrant-Whyte, H. Simultaneous Localization, Mapping and Moving Object Tracking. *The International Journal of Robotics Research 26*, 9 (2007), 889–916.

[179] Waxman, M. J., and Drummond, O. E. A bibliography of cluster (group) tracking. In *Defense and Security* (Orlando, FL, 2004), SPIE Proceedings, SPIE, pp. 551–560.

[180] Weigel, H., Cramer, H., Wanielik, G., Polychronopoulos, A., and Saroldi, A. Accurate Road Geometry Estimation for a Safe Speed Application. In *2006 IEEE Intelligent Vehicles Symposium (IV)* (Meguro-Ku, Japan, 2006), pp. 516–521.

[181] Weiss, T., Schiele, B., and Dietmayer, K. Robust Driving Path Detection in Urban and Highway Scenarios Using a Laser Scanner and Online Occupancy Grids. In *2007 IEEE Intelligent Vehicles Symposium (IV)* (Istanbul, Turkey, 2007), pp. 184–189.

[182] Weiss, T.-T. *Hochgenaue Positionierung und Kartographie mit Laserscannern für Fahrerassistenzsysteme.* Doctoral thesis, Universität Ulm. Fakultät für Ingenieurwissenschaften und Informatik, 2011.

[183] Welch, G., and Bishop, G. An Introduction to the Kalman Filter, 1995. Technical Report TR 95-041, University of North Carolina at Chapel Hill, USA.

[184] Wender, S. *Multisensorsystem zur erweiterten Fahrzeugumfelderfassung.* Doctoral thesis, Universität Ulm. Fakultät für Ingenieurwissenschaften und Informatik, 2008.

[185] Winner, H. Frontalkollisionsschutzsysteme. In *Handbuch Fahrerassistenzsysteme*, H. Winner, S. Hakuli, and G. Wolf, Eds. Vieweg+Teubner Verlag, Wiesbaden, 2012, pp. 522–542.

[186] Winner, H. Radarsensorik. In *Handbuch Fahrerassistenzsysteme*, H. Winner, S. Hakuli, and G. Wolf, Eds. Vieweg+Teubner Verlag, Wiesbaden, 2012, pp. 123–171.

[187] Winner, H., Hakuli, S., and Wolf, G., Eds. *Handbuch Fahrerassistenzsysteme*. Vieweg+Teubner Verlag, Wiesbaden, 2012.

[188] Wolf, D., and Sukhatme, G. S. Online simultaneous localization and mapping in dynamic environments. In *2004 IEEE International Conference on Robotics and Automation (ICRA)* (New Orleans, LA, USA, 2004), pp. 1301–1307 Vol.2.

[189] 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 *Proceedings of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation* (Anchorage, Alaska, 2010).

[190] Yguel, M., Aycard, O., and Laugier, C. Efficient GPU-based Construction of Occupancy Grids Using several Laser Range-finders. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Beijing, China, 2006).

[191] Yguel, M., Aycard, O., and Laugier, C. Wavelet Occupancy Grids: A Method for Compact Map Building. In *Field and Service Robotics*, P. Corke and S. Sukkariah, Eds., vol. 25 of *Springer Tracts in Advanced Robotics*. Springer Berlin Heidelberg, 2006, pp. 219–230.

[192] Yguel, M., Keat, Christopher Tay Meng, Braillon, C., Laugier, C., and Aycard, O. Dense Mapping for Range Sensors: Efficient Algorithms and Sparse Representations. In *Robotics*, W. Burgard, O. Brock, and C. Stachniss, Eds. MIT Press, Cambridge, 2008.

Publications, Patent Applications and Supervised Theses

[BHN+12] S. Bouzouraa, U. Hofmann, S. Neumaier, T. Weiherer, J. Storz, and M. Reichel. Methoden zur Erfassung der Fahrbahninfrastruktur in komplexen Fahrsituationen. In *5. Tagung Fahrerassistenz*, München, 2012.

[Dü15] D.-A. Dücker. Erkennung von Fahrstreifenverläufen in urbanen Szenarien als Grundlage für hochautomatisiertes Fahren. Master's thesis, Technische Universität Hamburg-Harburg, 2015.

[HBW14a] Ulrich Hofmann, Mohamed Essayed Bouzouraa, and Tobias Weiherer. Verfahren zum Betrieb eines Fahrerassistenzsystems für Überholvorgänge und Kraftfahrzeug, 2014. Deutsches Patent- und Markenamt, DE102014002116.5.

[HBW14b] Ulrich Hofmann, Mohamed Essayed Bouzouraa, and Tobias Weiherer. Verfahren zum Betrieb eines Fahrerassistenzsystems zur Unterstützung bei der Wahl einer Fahrspur, 2014. Deutsches Patent- und Markenamt, DE102014002115.7.

[HBW14c] Ulrich Hofmann, Mohamed Essayed Bouzouraa, and Tobias Weiherer. Verfahren zum Betrieb eines Sicherheitssystems eines Kraftfahrzeugs, 2014. Deutsches Patent- und Markenamt, DE102014002113.0.

[HBW14d] Ulrich Hofmann, Mohamed Essayed Bouzouraa, and Tobias Weiherer. Verfahren zum Betrieb eines zur wenigstens teilweise automatischen Fahrzeugführung ausgebildeten Fahrzeugsystems und Kraftfahrzeug, 2014. Deutsches Patent- und Markenamt, DE102014002114.9.

[Prö13] A. Pröbstl. Comparison of map matching approaches in vehicle to vehicle communication applications. Diploma thesis, Technische Universität München, 2013.

[Sei13] D. Seibert. Entwicklung von kombinierten Belegungskarten für Fahrerassistenzsysteme. Master's thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2013.

[SWBH12] K. Schueler, T. Weiherer, S. Bouzouraa, and U. Hofmann. 360 Degree multi sensor fusion for static and dynamic obstacles. In *2012 IEEE Intelligent Vehicles Symposium (IV)*, pages 692–697, Alcala de Henares, Madrid, Spain, 2012.

[WBH12] T. Weiherer, S. Bouzouraa, and U. Hofmann. A generic map based environment representation for driver assistance systems applied to detect convoy tracks. In *2012 15th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 691–696, Anchorage, AK, Sept 2012.

[WBH13] T. Weiherer, S. Bouzouraa, and U. Hofmann. An interval based representation of occupancy information for driver assistance systems. In *2013 16th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 21–27, The Hague, The Netherlands, Oct 2013.