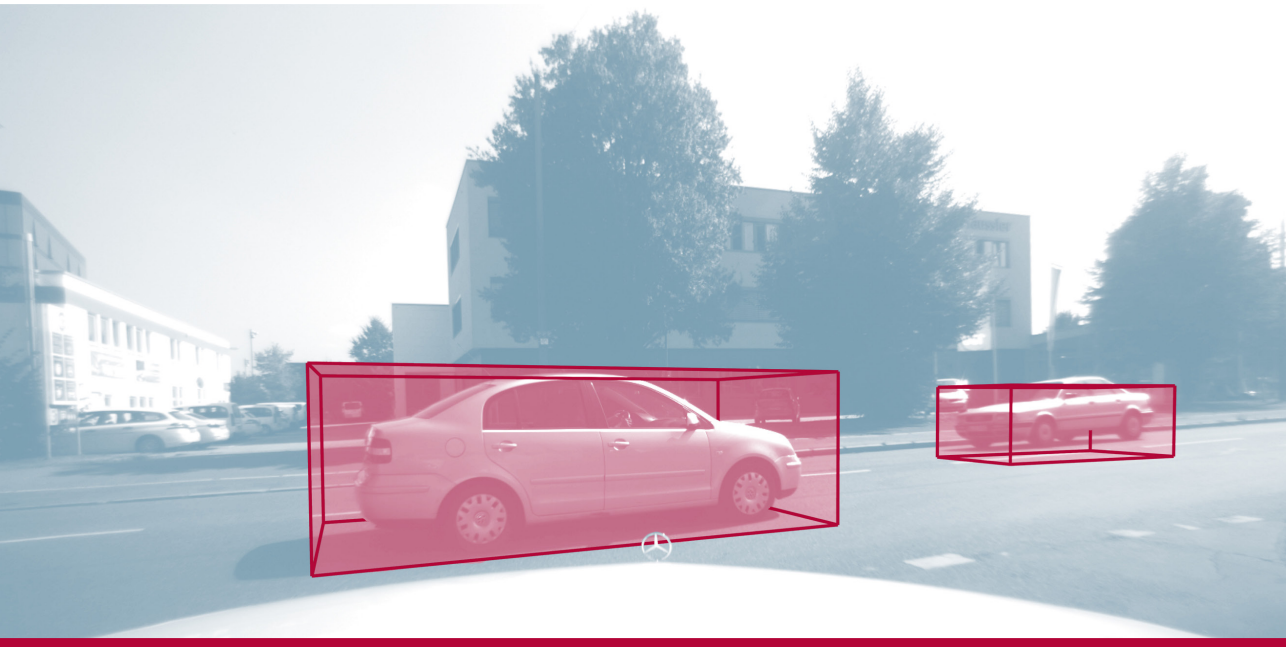
ulm university universität ulm



Fully Bayesian Vehicle Tracking Using Extended Object Models

Alexander Matthias Scheel

Band 30

Schriftenreihe des Instituts für Mess-, Regel- und Mikrotechnik

Alexander Matthias Scheel

Fully Bayesian Vehicle Tracking Using Extended Object Models

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

Herausgeber:
Prof. Dr.-Ing. Klaus Dietmayer

Band 30

Alexander Matthias Scheel

Fully Bayesian Vehicle Tracking Using Extended Object Models

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

Dissertation, Universität Ulm,
Fakultät für Ingenieurwissenschaften, Informatik und Psychologie, 2019

Impressum

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

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

Institut für Mess-, Regel- und Mikrotechnik der Universität Ulm 2019
Print on Demand

ISBN 978-3-941543-45-4 ISBN 978-3-941543-46-1 (E-Book)

ulm university universität
uulm

Fully Bayesian Vehicle Tracking Using Extended Object Models

DISSERTATION

zur Erlangung des akademischen Grades eines

DOKTOR-INGENIEURS
(Dr.-Ing.)

der Fakultät für Ingenieurwissenschaften, Informatik
und Psychologie der Universität Ulm

von

Alexander Matthias Scheel
aus Heilbronn

Gutachter:	Prof. Dr.-Ing. Klaus Dietmayer
	Prof. Dr.-Ing. Christoph Stiller
Amtierender Dekan:	Prof. Dr.-Ing. Maurits Ortmanns

Ulm, 19.07.2019

Acknowledgements

This dissertation is the result of my time at the Institute of Measurement, Control, and Microtechnology at Ulm University. Many people have contributed in different ways to compiling this book and to make these somewhat over four and half years so enjoyable.

First and foremost, I'd like to thank my advisor Prof. Dr.-Ing. Klaus Dietmayer for his trust in my abilities, his support, and for giving me the chance to work on all the topics I was interested in. I would furthermore like to thank Prof. Dr.-Ing. Christoph Stiller for his interest in my research and for being my second examiner.

Also, I'd like to express my gratitude to Stephan Reuter who, with his enormous tracking knowledge and post-doc experience, has always been a very encouraging mentor and who has contributed to this work in numerous ways and discussions.

Thanks to all the fellow PhD students at the institute who have accompanied me through the different phases of graduation and who made the time worthwhile. In particular, I'd like to thank those who made the "C++ Center of Excellence" the most awesome office, who were always open for technical and non-technical conversations, who made good sports partners, who engaged in lively discussion during lunch time, and who proofread the chapters of this thesis: Nils Rexin, Stefan Hörmann, Felix Kunz, Manuel Herzog, Manuel Stübler, Martin Bach, Andreas Danzer, Fabian Gies, Daniel Stumper, Florian Schüle, Martin Herrmann, and Johannes Müller. Also, I'd like to thank Franz Gritschneder and Markus Thom for providing the scene labeling approach that I used for fusing semantic information from cameras.

Thank you to all the students that made university life so much more thrilling through our three years of competing in the Carolo-Cup and to the students that supported my research by letting me supervise their theses. I'd like to highlight the work of Christina Knill whose thesis has kick-started the radar tracking part and has found its way into this book in form of the direct scattering model.

Furthermore, I'd like to thank the institute staff: Thank you Claudia Fricke and Michael Buchholz for your kindness and help in so many different administrative tasks. Thank you, Oliver Betz, Franz Degenhard, Uwe Kerner, Thomas Löffler, and

Martin Nieß for your work on our experimental vehicles and all the other things you did for us.

Finally, I'd like to thank my parents, my sister, and particularly my wife for their never-ending support and patience that carried me throughout those years.

Abstract

Environment perception systems for automated vehicles and driver assistance systems commonly use radar and lidar sensors as well as cameras for tracking other vehicles. While these sensors exhibit different strengths and weaknesses, they share the ability to resolve multiple measurements for a single object: lidar sensors typically provide multiple distance measurements for a vehicle, modern high-resolution radar sensors yield multiple detections, and objects cover a set of pixels in camera images.

Such dense data is valuable for obtaining a detailed and precise representation of the environment. From a tracking perspective, however, it poses three major challenges: First, tracking algorithms have to process multiple measurements for one object and thus face the extended object problem. This conflicts with the classical assumption of at most one measurement per cycle and object. Secondly, the increased amount of data further complicates multi-object problems which involve measurement-to-object associations in the presence of clutter, occlusion, and misdetections. Thirdly, the data from several heterogeneous sensors needs to be fused into a consistent estimate in order to benefit from complementary sensor strengths or to either achieve increased sensor fields of view or redundancy.

This thesis presents a tracking framework that is based on finite set statistics (FISST) and tackles these challenges for the application of vehicle tracking in a Bayesian fashion. The employed multi-object filter uses labeled multi-Bernoulli distributions and allows for combining different sensor modules with sensor-specific update routines through centralized fusion. Furthermore, extended object measurement models which work on the full raw sensor data are developed. In particular, it is demonstrated how separable likelihood approaches can be used to incorporate accurate vehicle models for lidar data and semantically labeled camera images. Additionally, two radar vehicle models are developed. They are able to process multiple radar detections and allow for tracking vehicles in arbitrary maneuvers, including turning vehicles or cross-traffic. While the first radar model is based on expert knowledge, the second uses variational Gaussian mixtures and is learned from actual measurements. The methods are evaluated on experimental data with accurate ground truth. The results demonstrate that the extended object models achieve precise estimates, that the data-driven variational radar model is able to outperform the manually designed model, and how different sensor combinations improve the performance.

Kurzfassung

Umgebungserfassungssysteme für automatisierte Fahrzeuge oder Fahrerassistenzsysteme verwenden üblicherweise Radar- oder Lidarsensoren sowie Kameras, um andere Fahrzeuge zu erfassen und zu verfolgen. Während diese Sensoren verschiedene Vor- und Nachteile aufweisen, so teilen sie die Fähigkeit, mehrere Messungen für ein einzelnes Objekt auflösen zu können: Lidarsensoren liefern üblicherweise mehrere Distanzmessungen für ein Fahrzeug, moderne hochauflösende Radarsensoren erzeugen mehrere Detektionen und Objekte erstrecken sich über mehrere Pixel eines Kamerabilds.

Diese Datendichte ist wertvoll, um ein detailliertes und präzises Abbild der Umgebung zu erhalten. Aus Objektverfolgungssicht ergeben sich daraus jedoch drei Herausforderungen: Erstens muss die Objektverfolgung mehrere Messungen für ein einzelnes Objekt verarbeiten und steht daher vor dem Problem, mit sogenannten ausgedehnten Objekten umgehen zu müssen. Dies steht im Widerspruch zur klassischen Annahme von maximal einer Messung pro Objekt und Messzyklus. Zweitens erschwert die zusätzliche Datenmenge Multiobjektproblemstellungen, welche Assoziationen zwischen Messungen und Objekten unter Berücksichtigung von Falschmessungen, Sensorrauschen, Verdeckungen und Fehldetektionen erforderlich machen. Drittens müssen Messungen von verschiedenen, heterogenen Sensoren zu einer einheitlichen Schätzung fusioniert werden, um von sich ergänzenden Sensorstärken zur profitieren oder um vergrößerte Sensorsichtbereiche oder Redundanz zu erreichen.

Diese Arbeit stellt ein Objektverfolgungssystem vor, welches auf der Methode der Finite-Set-Statistics (FISST) beruht und diese drei Herausforderungen für die Anwendung der Fahrzeugverfolgung auf Bayes'sche Weise löst. Der eingesetzte Multiobjektfilter verwendet Labeled-Multi-Bernoulli-(LMB)-Verteilungen und ermöglicht die Kombination verschiedener Sensormodule mit sensorspezifischen Update-Routinen durch eine zentrale Fusionsarchitektur. Zusätzlich werden Messmodelle für ausgedehnte Objekte entwickelt, die direkt auf den Sensorrohdaten arbeiten. Im Besonderen wird gezeigt, wie separierbare Wahrscheinlichkeitsfunktionen es ermöglichen, präzise Fahrzeugmodelle für Laserdaten sowie semantisch klassifizierte Kamerabilder zu integrieren. Außerdem werden zwei Radarmodelle entwickelt, die mehrere Radardetektionen verarbeiten können und das Verfolgen von Fahrzeugen in beliebigen Manövern erlauben, einschließlich abbiegender Fahrzeuge und Querverkehr. Während

das erste Radarmessmodell auf Expertenwissen basiert, wird das zweite mithilfe variationeller Gaußmixturen auf Basis realer Messungen gelernt. Die Ansätze werden auf experimentellen Sensordaten, für welche akkurate Referenzwerte zur Verfügung stehen, ausgewertet. Die Ergebnisse zeigen, dass die Messmodelle für ausgedehnte Objekte präzise Schätzergebnisse erreichen, dass das datengestützte variationelle Radarmodell bessere Ergebnisse erreicht als das händisch entwickelte Modell und wie die Kombination verschiedener Sensoren die Leistungsfähigkeit erhöht.

Contents

Chapter 1

Introduction

One of the basic lessons in driving school is how to correctly monitor the surrounding of one's vehicle, for example by doing the shoulder check. This ability is vital for safely navigating through dense traffic and avoiding collisions with other traffic participants. Of course, perceiving the surrounding is equally important for modern driver assistance systems or even automated vehicles since the decision to trigger an emergency braking maneuver, an evasion maneuver, a lane change or simply planning the future trajectory of the vehicle cannot be made without any knowledge about the environment.

In most vehicle environment perception systems, tracking modules are a key component to obtain an object-level representation of the surrounding. That is, they provide a list of objects which represent the other traffic participants and estimate their state, i.e. their position, orientation, motion, or size, using the measurements of the vehicle's sensors. For this purpose, today's production cars mostly use ultrasonic sensors, radar sensors, and monocular or stereo cameras. Also, lidar sensors, which measure distances using laser rays, have been becoming increasingly popular in sensor set-ups of experimental automated vehicles. Each sensor type exhibits different shortcomings and advantages. For instance, camera sensors are particularly good at providing valuable semantic information, whereas lidar sensors yield precise distance and geometric information. In contrast, radar sensors are able to directly measure the object motion by exploiting the Doppler effect and are more robust against occlusions or adverse weather or lighting conditions. The combination of multiple heterogeneous sensors is often desirable to benefit from these complementary sensor strengths. At the same time, sensor fusion allows to increase the field of view (FOV) or to achieve redundancy in cases of sensor failure.

The amount of information that is conveyed by a sensor strongly depends on its resolution in relation to the observed object's extent. The higher the resolution is, the more measurements the sensor is able to provide from the object. Cameras,

for instance, have a particularly high resolution and provide rich and detailed information about the environment using many pixels. Yet, the advent of densely scanning lidar sensors and advances in radar sensor technology have led to additional high-resolution sensors for vehicle environment perception. That is, lidar sensors yield multiple distance measurements and modern radar sensors are able to resolve multiple detections for a single vehicle. This increase in information is valuable as it allows constructing a more detailed representation of the environment, for example by being able to extract the object dimensions or contour.

For obtaining state estimates of all objects in a vehicle's surrounding, tracking algorithms face three major challenges: First, the sensor measurements need to be properly processed to extract the estimates. This is typically done by finding appropriate measurement models that relate the received measurements to the object states. Secondly, tracking other traffic participants constitutes a complex multi-object tracking problem. There may be several objects in the FOV and the sensors additionally output clutter measurements that originate from objects that are not of interest (e.g. stationary objects which are not within the scope of the tracking module such as houses) or are simply sensor noise. Tracking algorithms therefore need to determine the number of objects that are currently present and need to associate the received measurements to existing objects or to clutter. Yet, such associations may be ambiguous, e.g. in the case of objects that are close to each other, and finding the correct association is oftentimes non-trivial. As a third challenge, the measurements from different, possibly heterogeneous sensors need to be fused to consistent estimates in order to achieve the mentioned benefits of sensor fusion.

Particularly the first two challenges are further complicated by the increased resolution of modern sensors. Tracking algorithms now receive multiple measurements per object and the object extent or shape plays a crucial role in the measurement process. This is commonly referred to as the extended object problem. However, the presence of multiple measurements violates the assumption of at most one measurement per object that is made in most classical tracking algorithms. Therefore, new ways to process the additional data are necessary. Apart from multiple measurements for relevant objects, high-resolution sensors also yield more measurements from objects that are irrelevant for the tracking task and consequently an increased number of clutter measurements. The additional clutter measurements in conjunction with multiple measurements per object exacerbate the association problem in a multi-object setting.

A widespread solution to the mentioned challenges is to break the full problem down into subproblems. For example, data processing is usually outsourced to preprocessing routines that first convert the multiple measurements to single object detections or meta-measurements by, for example, computing the centroid or fitting

bounding boxes. Thus, the amount of data is reduced, the assumption of receiving at most one measurement is again satisfied, and standard tracking approaches can be applied. Also, the multi-object problem is oftentimes solved by making hard association decisions, treating the objects independently in architectures with multiple tracking filter instances, and using heuristics for track management, i.e. track initialization and termination. Such strategies simplify the problem but can lead to deteriorated tracking performance. Using one out of many possible associations, for instance, discards information and wrong associations cannot simply be corrected in subsequent time steps. On the other hand, preprocessing routines may fail if the available information from one time step is not sufficient for extracting meaningful meta-measurements or if there is at least considerable ambiguity that can only be resolved by accumulating information over several time steps. Moreover, the reduction of data to a single meta-measurement inherently leads to a loss of information for the tracking filter.

For this reason, there has been active research in the tracking community to find new methods that deal with the multi-object and extended object problems in a principled way. In particular, different multi-object filters that consider the entire multi-object problem and take the association uncertainty into account have been developed. Also, new filter variants that tackle the extended object problem have been proposed. They make use of extended object measurement models which take the object extent into account and are able to process multiple measurements per object. In the past years, efforts have been made to transfer these new approaches to automotive tracking applications. Multi-object filters and existing extended object measurement models have been applied to track traffic participants and new measurement models for different object and sensor types have been developed. Yet, some of the available extended object models, for example the random matrix model, are based on rather restrictive theoretical assumptions that are not suitable in the vehicle tracking context. Others require detailed expert knowledge in the design phase, such as manually created reflection point models, or use simplistic models of the measurement principle that do not allow for full usage of the available information, for example knowledge about the absence of objects contained in lidar scans. Also, most work focuses on particular parts of the problems, e.g. the multi-object problem or extended object measurement models for a certain sensor type. There has, however, been little work on tracking frameworks that aim at fusing data from heterogeneous sensors to achieve a continuous 360° coverage of the vehicle environment while fully considering the multi-object and extended object problems.

Therefore, the objective of this thesis is to advance the work on multiple extended object tracking for vehicle environment perception in two ways: First, the goal is to propose new extended object measurement models which not only operate directly on the raw sensor data but also naturally emulate the measurement principle to achieve accurate tracking results and the ability to cope with ambiguity. Secondly, the

thesis aims at developing a tracking framework that solves the mentioned challenges rigorously in an integral Bayesian formulation. In this regard, the thesis particularly focuses on tracking vehicles on the basis of radar, lidar, and camera data. The Bayesian end-to-end formulation is achieved by using a multi-object approach based on the theory of finite set statistics (FISST). It describes the entire multi-object state as well as the measurements in form of random finite sets (RFSs), i.e. set-valued random variables. By providing mathematical tools for estimating the multi-object posterior distribution in a Bayesian fashion, FISST allows for a probabilistic top-down description of the complete tracking problem.

The specific contributions of this thesis are threefold. First, it proposes a FISST-based tracking architecture which uses a labeled RFS formulation, allows for the combination of different sensor-specific multi-object measurement models, and contains a prediction routine that approximately avoids overlapping vehicles. Secondly, it develops two extended object models for radar-based vehicle tracking that on the one hand use expert knowledge and on the other hand a machine learning approach based on variational Bayes, a new approach for lidar tracking that allows to integrate ray-based extended object models into the FISST world, and a simple extended object measurement model for incorporating semantical information from scene-labeled monocular camera images. Thirdly, the thesis constitutes the first work that implements data fusion from three sensor types using a full extended object formulation. All proposed methods are evaluated using experimental data including accurate ground truth.

The remainder of the thesis is organized as follows: Chapter 2 first introduces the basics of Bayesian inference, which forms the foundation of all theoretical methods that are applied in the tracking framework. Subsequently, Chapter 3 reviews the fundamentals of tracking and particularly those methods that are paramount to this work in the fields of single object tracking, multi-object tracking, extended object tracking, and sensor fusion. Chapter 4 then discusses the state of the art in vehicle tracking using the three sensor types and derives the proposed tracking framework, including the problem formulation, architecture, and multi-object prediction. Afterwards, Chapter 5 describes the extended object measurement models for lidar, camera, as well as radar data, and Chapter 6 evaluates and discusses the proposed approaches. Chapter 7 concludes the thesis.

Chapter 2

Fundamentals of Bayesian Inference

The goal of statistical inference is to obtain information about unknown variables or models from measurements [BT08, p. 408]. These variables may for example represent states of a dynamic system or static quantities and one might be interested in finding the best model structure or model parameters. Two different schools of thought have emerged in statistical inference for solving such problems: the Bayesian and the frequentist approach [BT08, p. 408]. While frequentists treat the quantities of interest as unknown constants, the Bayesian approach models them as random variables and aims at inferring their distributions. Further conclusions about the values can then be drawn from these distributions.

There has been a prolonged debate about the assets and drawbacks of both approaches, see for example [HU91]. Frequentists oftentimes object to the idea of modeling an unknown but essentially deterministic quantity as random variable with a prior distribution, whereas supporters of the Bayesian approach argue that the introduction of a prior allows to systematically incorporate prior knowledge [BT08, p. 409]. Despite this debate, the Bayesian method has attracted great interest in the recent decades. It is widely used in the field of machine learning and is paramount in tracking. Also, the introduction of a prior helps to avoid singularities that sometimes arise in classical solutions; see for example [Bis13, pp. 480-481].

For these reasons, the methods in this thesis are entirely Bayesian and this chapter aims at establishing the necessary mathematical foundations. First, Section 2.1 introduces the basic mathematical concept and Bayes' theorem. Then, Sections 2.2 to 2.4 discuss different possibilities to solve the Bayes update.

2.1 Basic Principle and Bayes' Theorem

Bayesian inference can be applied to a variety of problems in terms of the number of involved variables and their nature. For the sake of simplicity, however, this section first focuses on a simple problem that comprises two continuous random variables[1]: the unknown quantity and an observation. The unknown quantity is denoted by x and is oftentimes termed latent (i.e. hidden or unobserved) variable. It takes its values from the sample space $\mathbb{X}$, that is $x \in \mathbb{X}$, which comprises all possible values. The latent variable x cannot be observed directly, e.g. due to the system structure. Instead, information can be obtained through the measurement or observed variable $z \in \mathbb{Z}$ which is connected to x and where $\mathbb{Z}$ denotes the measurement space. The goal is to find the posterior distribution $p(x|z)$ over the value of x which incorporates the information contained in the measurement[2]. In this example, both random variables are assumed to be continuous and hence, their probabilistic characteristics are described by probability densities. Yet, the concept can be easily adapted to discrete random variables by using probability mass functions and replacing the integrals that occur in the following by sums.

The basic procedure for inferring the posterior distribution consists of three steps. First, a prior density $p(x)$ over the latent variable is formulated. It contains all prior knowledge about the variable and may be chosen rather uninformative, i.e. "flat", or informative, i.e. "tight", depending on the amount or reliability of the knowledge.

In a second step, a measurement model is specified by formulating the likelihood function $g(z|x)$. It relates x to the received measurement by modeling the measurement process and incorporating measurement noise. In each experiment, a particular realization of the random variable z is observed while x remains the unknown parameter of the function. This usage as a function of x explains the term likelihood function despite $g(z|x)$ being a density of the measurement z that is defined on $\mathbb{Z}$.

The third step then computes the posterior density $p(x|z)$ from the prior density and the likelihood function. That is, the prior knowledge is refined by applying Bayes' theorem, which yields

$$p(x|z) = \frac{g(z|x)p(x)}{p(z)} = \frac{g(z|x)p(x)}{\int g(z|x)p(x)\,\mathrm{d}x}. \tag{2.1}$$

[1] The random variables may either be scalars or vectors. This thesis does not distinguish between these two types. Please refer to the list of symbols at the end of the thesis for an overview of the used notation.

[2] Note that this thesis does not make a notational difference between a random variable and its realization. Also, probability densities or probability mass functions are generally denoted by $p(\cdot)$. The functions are distinguishable by the arguments (e.g. prior $p(x)$ vs. posterior $p(x|z)$). Some special densities, such as likelihood functions, are indicated by special letters.

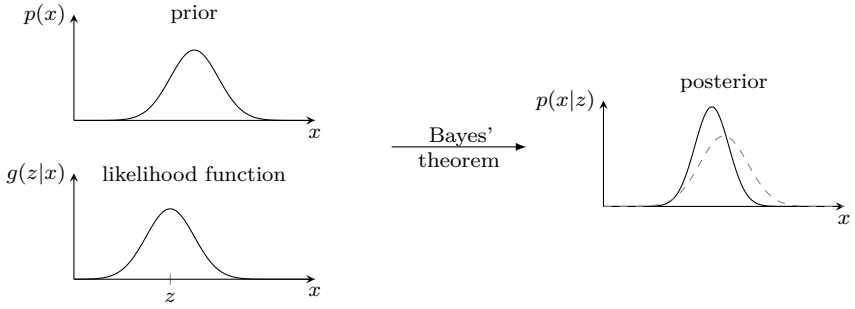


Figure 2.1: Illustration of the Bayesian inference procedure for a scalar variable x with measurement z from the same space

The factor $p(z)$ in the denominator normalizes the resulting posterior and is sometimes referred to as model evidence. It indicates how likely the measurement is when marginalizing over all possible values of the latent variable.

Figure 2.1 illustrates the basic procedure of Bayesian inference for a simple scalar example. It is assumed that $\mathbb{X}$ and $\mathbb{Z}$ are identical and it is shown how incorporating information from a measurement leads to a refined posterior density with decreased variance.

Once the posterior density is obtained, it can be used to infer different properties about the latent variable. For example, one might be interested in obtaining an estimate $\hat{x}$. In Bayesian inference there are two mainly used estimators: the maximum a posteriori (MAP) estimator and the minimum mean square error (MMSE) estimator [BLK01, pp. 92, 99]. The MAP estimator is defined as

$$\hat{x}_{\mathrm{MAP}} = \arg\max_{x} p(x|z) \tag{2.2}$$

and searches—as implied by the name—for the maximum value of the posterior distribution. In contrast, the MMSE estimator outputs the estimate that minimizes the mean square error $\mathbb{E}[(\hat{x} - x)^2|z]$ where $\mathbb{E}[\cdot]$ denotes the expectation. It can be shown [BLK01, p. 99] that this results in

$$\hat{x}_{\mathrm{MMSE}} = \mathbb{E}[x|z] = \int x p(x|z)\,\mathrm{d}x. \tag{2.3}$$

That is, the MMSE estimate is the expected value of the posterior density.

Despite the relatively simple overall procedure, performing Bayesian inference can be computationally challenging. This is due to the integral in the denominator of (2.1)

which needs to be evaluated for obtaining the posterior density. While closed-form solutions are available for some particular densities and likelihood functions, others require approximation schemes. The following sections detail different approaches for solving (2.1) that are used in this thesis.

2.2 Parametric Distributions and Conjugate Priors

The probably most widely used way to describe the characteristics of a random variable is to use parametric distributions. These distributions have in common that their shape and statistical moments are governed by a finite set of parameters. There are many different variants suitable for different applications or different types of random variables. In Bayesian inference, parametric distributions can be used as densities over the latent variable or as likelihood functions. If used for the likelihood function, one or more of the distribution parameters take up the role of the unknown latent variables which influence the shape of the likelihood function.

Conjugate priors, as first introduced in [RS70], are a particularly useful class of distributions over the latent variable as they allow for efficient analytic solutions to (2.1). Loosely speaking, a prior distribution $p(x)$ is said to be a conjugate prior to a likelihood function $g(z|x)$ if the resulting posterior distribution $p(x|z)$ has the same functional form as $p(x)$ [GCS+14, pp. 35-36]. Thus, the recursive application of Bayes' theorem is facilitated and (2.1) reduces to updating the parameters of the latent variable's distribution.

The following subsections introduce several continuous and discrete parametric distributions that are relevant for the thesis. Also, their conjugate priors are presented where necessary. Additionally, short notations for the probability mass functions and densities which emphasize the dependence on the distribution parameters are introduced.

2.2.1 Bernoulli Distribution

One of the simplest discrete distributions is the Bernoulli distribution. It is defined for the two events $x \in \{0, 1\}$ and is used to model the probability of success in trials with binary outcome, e.g. when flipping a coin. Its probability mass function is [Bis13, p. 685]

$$p(x) = \mathrm{Ber}(x|r) = r^x(1-r)^{1-x} \tag{2.4}$$

with the parameter $r \in [0, 1]$ denoting the probability of success. The conjugate prior to the Bernoulli distribution is the beta distribution [Bis13, p. 686], which is, however, not used in this thesis.

2.2.2 Multinomial and Dirichlet Distributions

The multinomial distribution is an extension of the Bernoulli distribution to experiments with multiple possible outcomes. It is thus a multivariate discrete distribution over a binary 1-of-n_x variable with n_x dimensions. That is, exactly one element of the random vector x takes the value 1 while the others are 0. The probability mass function is [Bis13, pp. 690-691]

$$p(x) = \text{Mult}(x|w) = \prod_{i=1}^{n_x} w_i^{x_i}. \tag{2.5}$$

The parameter vector w contains the probability for each possible outcome and its elements w_i thus have to sum to one.

The Dirichlet distribution is the conjugate prior to the multinomial distribution. It is a continuous distribution over a random vector with n_w elements and where the elements satisfy $0 \leq w_i \leq 1$ and $\sum_{i=1}^{n_w} w_i = 1$. Hence, it is a suitable distribution for the parameter vector w of the multinomial distribution. The probability density is [Bis13, p. 687]

$$p(w) = \text{Dir}(w|\rho) = C(\rho) \prod_{i=1}^{n_w} w_i^{\rho_i - 1}, \tag{2.6}$$

where the elements of the parameter vector ρ have to satisfy $\rho_i > 0$. When the Dirichlet distribution is used as conjugate prior to the multinomial distribution, ρ_i can be interpreted as the effective number of observations of a specific outcome w_i in a multinomial experiment [Bis13, p. 687]. The normalization factor is

$$C(\rho) = \frac{\Gamma(\tilde{\rho})}{\prod_{i=1}^{n_w} \Gamma(\rho_i)}, \tag{2.7}$$

with the Gamma function $\Gamma(\cdot)$ (see e.g. [FEHP11, pp. 56, 57]) and

$$\tilde{\rho} = \sum_{i=1}^{n_w} \rho_i. \tag{2.8}$$

2.2.3 Poisson Distribution

The Poisson distribution is a discrete distribution which is used for modeling the number of occurrences of an event. It is defined by the density [FEHP11, p. 152]

$$p(x) = \mathrm{Poi}(x|\lambda) = \frac{\lambda^x}{x!}e^{-\lambda} \tag{2.9}$$

and is defined for $x \in \mathbb{N}_0$, i.e. for all natural numbers including zero. The rate parameter $\lambda > 0$ models the expected number of occurrences ($\mathbb{E}[x] = \lambda$) and also determines the variance ($\mathbb{E}[(x - \mathbb{E}[x])^2] = \lambda$). The conjugate prior to the Poisson distribution, which is again not required in this thesis, is the gamma distribution (see e.g. [FEHP11, Chapter 22]).

2.2.4 Uniform Distribution

The uniform distribution is a continuous distribution defined over the interval $x \in [u_l, u_u] \; \forall \; u_u > u_l$. Its probability density function is [Bis13, p. 692]

$$p(x) = \mathrm{U}(x|u_l, u_u) = \frac{1}{u_u - u_l} \tag{2.10}$$

and assigns the same density value to each possible x. There exist different types of conjugate priors from the class of Pareto distributions [Fin97, p. 13], depending on which of the two interval parameters is unknown and serves as latent variable.

2.2.5 Multivariate Gaussian and Wishart Distributions

The multivariate Gaussian distribution is one of the most common continuous distributions over random vectors. Its density is given by

$$p(x) = \mathcal{N}(x|\mu, \underline{\Sigma}) = \frac{1}{\sqrt{(2\pi)^{n_x} \det(\underline{\Sigma})}} e^{-\frac{1}{2}(x-\mu)^{\mathrm{T}}\underline{\Sigma}^{-1}(x-\mu)} \tag{2.11}$$

where n_x is again the dimension of the random vector x, μ is the mean, and $\underline{\Sigma}$ is the covariance matrix. Furthermore, $\det(\cdot)$ denotes the determinant. In some cases, it is convenient to replace the covariance matrix by its inverse, the precision matrix $\underline{H}$. Then, the short notation is $\mathcal{N}(x|\mu, \underline{H}^{-1})$.

If the likelihood function is a Gaussian distribution with unknown mean μ and known

covariance matrix $\underline{\Sigma}$, the conjugate prior is again a Gaussian distribution. This fact is, for instance, used in the Kalman filter (see Section 3.1.1). In contrast, the conjugate prior of a Gaussian likelihood function with known mean μ and unknown precision matrix $\underline{H}$ is a Wishart distribution [Bis13, p. 693]. The Wishart distribution is hence a distribution over a matrix-variate random variable $\underline{H}$ under the assumption that $\underline{H}$ is positive semi-definite. Its probability density is [Bis13, p. 693]

$$p(\underline{H}) = \mathcal{W}(\underline{H}|\underline{V}, \nu) = B(\underline{V}, \nu) \det(\underline{H})^{\frac{\nu - n_h - 1}{2}} e^{-\frac{1}{2}\operatorname{Tr}(\underline{V}^{-1}\underline{H})} \tag{2.12}$$

where $\operatorname{Tr}(\cdot)$ is the trace of a matrix, $\underline{H}$ has dimensions $n_h \times n_h$, and $\underline{V}$ is the symmetric, positive definite $n_h \times n_h$ scale matrix parameter. The normalization factor is

$$B(\underline{V}, \nu) = \det(\underline{V})^{-\frac{\nu}{2}} \left(2^{\frac{\nu n_h}{2}} \pi^{\frac{n_h(n_h-1)}{4}} \prod_{i=1}^{n_h} \Gamma\left(\frac{\nu + 1 - i}{2}\right) \right)^{-1}. \tag{2.13}$$

Also, the number of degrees of freedom ν has to satisfy $\nu > n_h - 1$.

If neither the mean nor the precision matrix of a Gaussian likelihood function is known, the conjugate prior is a Gaussian-Wishart distribution [Bis13, p. 690]. Its probability density is

$$p(\mu, \underline{H}) = \mathcal{N}(\mu|\gamma, (\beta\underline{H})^{-1})\mathcal{W}(\underline{H}|\underline{V}, \nu), \tag{2.14}$$

with the mean of the prior Gaussian γ and the additional scaling parameter β.

2.2.6 Multivariate Student's t-Distribution

If a random variable x is distributed according to a normal distribution with mean μ and a precision matrix that follows a Wishart prior, the density of x can be obtained by marginalizing over the random precision matrix. This yields a multivariate Student's t-distribution of the form [AK65; Bis13, p. 692]

$$p(x) = \mathrm{St}(x|\mu, \underline{\Lambda}, \nu) = \frac{\Gamma\left(\frac{\nu}{2} + \frac{n_x}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)} \frac{\det(\underline{\Lambda})^{\frac{1}{2}}}{(\nu\pi)^{\frac{n_x}{2}}} \left(1 + \frac{(x-\mu)^{\mathrm{T}}\underline{\Lambda}(x-\mu)}{\nu}\right)^{\frac{-\nu - n_x}{2}}. \tag{2.15}$$

Here, $\underline{\Lambda}$ is the new precision matrix parameter of Student's t-distribution. For $\nu \to \infty$, the multivariate Student's t-distribution becomes a multivariate Gaussian distribution with mean μ and precision matrix $\underline{\Lambda}$ [Bis13, p. 692]. Appendix A lists marginal and conditional densities that can be derived from Student's t-distribution.

2.3 Importance Sampling

In some cases, there is no convenient analytical solution of (2.1). If, for example, there is no suitable conjugate prior for a likelihood function, or the integral in the denominator cannot be evaluated, approximation schemes are necessary. Sampling methods are one alternative that is especially suitable, if an exact analytical description of the posterior density is dispensable and the goal is rather to compute specific properties of the posterior distribution, such as the expected value. An overview of different sampling methods can be found in [Bis13, Chapter 11]. Here, the focus is set on importance sampling, which forms the basis for the particle filter approaches that are used in the subsequent chapters. The explanations closely follow [RAG04, pp. 36-37].

Before turning to the solution of Bayes' theorem, the section starts with some preliminary considerations. Suppose the goal is to compute the expected value of a random variable with density $p(x)$ with respect to some function $f(\cdot)$,

$$\mathbb{E}[f(x)] = \int f(x)p(x)\,\mathrm{d}x, \tag{2.16}$$

which, for instance, corresponds to computing the mean when setting $f(x) = x$. If an analytical solution of the integral is intractable, it is possible to instead draw samples $x^{(i)}$ from $p(x)$ and compute the expected value using Monte Carlo integration. Then, the integral reduces to a sum over the n_p samples

$$\mathbb{E}[f(x)] \approx \frac{1}{n_p}\sum_{i=1}^{n_p} f(x^{(i)}), \tag{2.17}$$

which allows for a simple numerical implementation.

Now suppose that one cannot directly sample from $p(x)$. In such cases, the technique of importance sampling can be applied. The integral from (2.16) is rewritten as

$$\mathbb{E}[f(x)] = \int f(x)\frac{p(x)}{q(x)}q(x)\,\mathrm{d}x \tag{2.18}$$

by introducing the proposal density $q(x)$. If the proposal density satisfies [RAG04, p. 36]

$$p(x) > 0 \Rightarrow q(x) > 0 \;\forall\; x \in \mathbb{X}, \tag{2.19}$$

the expected value

$$\mathbb{E}[f(x)] \approx \frac{1}{n_p} \sum_{i=1}^{n_p} f(x^{(i)}) \tilde{w}(x^{(i)}) \tag{2.20}$$

can be computed by sampling from the proposal density $q(x)$ and computing the importance weights

$$\tilde{w}(x^{(i)}) = \frac{p(x^{(i)})}{q(x^{(i)})}. \tag{2.21}$$

In cases where the density $p(x)$ or the proposal density $q(x)$ are only known up to a normalization constant, i.e. $\tilde{p}(x)$ and $\tilde{q}(x)$, the importance weights $\tilde{w}(x^{(i)}) = \tilde{p}(x^{(i)})/\tilde{q}(x^{(i)})$ have to be normalized via [RAG04, p. 37; Bis13, p. 533]

$$w(x^{(i)}) = \frac{\tilde{w}(x^{(i)})}{\sum_{j=1}^{n_p} \tilde{w}(x^{(j)})}, \tag{2.22}$$

and the expected value is then approximated by

$$\mathbb{E}[f(x)] \approx \sum_{i=1}^{n_p} f(x^{(i)}) w(x^{(i)}). \tag{2.23}$$

Note that the choice of the proposal density is of particular practical relevance in importance sampling. Proposal densities that are close to the actual densities avoid the generation of many samples that receive low weights and hardly contribute to the final result.

In Bayesian inference, one's goal is oftentimes to compute the expected value of a latent variable's posterior density $p(x|z)$, e.g. to obtain the MMSE estimate (cf. (2.3)). If the solution of (2.1) is not readily available, this density will be unknown and—as in the previous considerations—it cannot be sampled from. Yet, (2.1) can be solved approximately using importance sampling. The expected value of the posterior density is

$$\mathbb{E}[x|z] = \int x p(x|z) \, \mathrm{d}x. \tag{2.24}$$

For some proposal density $q(x)$, the importance weights can be obtained by inserting (2.1) into (2.21)

$$w(x^{(i)}) = \frac{g(z|x)p(x)}{p(z)q(x)} \propto \frac{g(z|x)p(x)}{q(x)} \tag{2.25}$$

and noting that omitting the constant normalizing factor $p(z) = \int g(z|x)p(x) \, \mathrm{d}x$, which is assumed intractable, leads to importance weights that need to be normalized. If, for instance, the prior density $p(x)$ is chosen as proposal density, the importance

weights further reduce to $w(x^{(i)}) \propto g(z|x) = \tilde{w}(x^{(i)})$. Hence, an approximation of (2.24) can be obtained by drawing samples from the prior density $p(x)$, evaluating the likelihood function point-wise for the samples, normalizing the importance weights through (2.22), and computing

$$\mathbb{E}[x|z] \approx \sum_{i=1}^{n_p} w(x^{(i)}) x^{(i)}. \tag{2.26}$$

2.4 Variational Bayes

Another alternative to solving (2.1) in cases where simple analytical solutions are not available is the use of the variational Bayes framework. In this method, approximate densities are obtained by maximizing lower bounds on the true densities using calculus of variations. In the following, the basic principle of the approach is first explained in Section 2.4.1 before Section 2.4.2 introduces the application of the method to density estimation using Gaussian mixtures. The explanations are mostly based on [Bis13, Chapters 9 and 10].

2.4.1 Basic Principle

So far, the previous sections dealt with inferring the posterior density of a single latent variable x given a single measurement z, whereas variational Bayes is typically applied to larger models with multiple latent variables. Also, multiple measurements are typically received over several experiments. Therefore, the estimation problem is generalized in this section. All n unknown variables of the model are combined in the set $X = \{x^{(1)}, \dots, x^{(n)}\}$ and the m measurements in the set $Z = \{z^{(1)}, \dots, z^{(m)}\}$. Again, the goal is to find the posterior distribution $p(X|Z)$.

The cornerstone of variational Bayes is the insight, that one can construct a lower bound on the logarithm of the model evidence $p(Z)$, also called the log marginal distribution,

$$\ln p(Z) = \ln \left(\int q(X) \frac{p(X,Z)}{q(X)} \, \mathrm{d}X \right) \geq \int q(X) \ln \left(\frac{p(X,Z)}{q(X)} \right) \mathrm{d}X = \mathcal{L}(q(X)), \tag{2.27}$$

where the relationship follows from Jensen's inequality [Jen06] for concave functions (cf. [Bea03, p. 47]). To construct the lower bound $\mathcal{L}(q(X))$, an auxiliary density over the latent variables $q(X)$ was introduced. Note that the lower bound is a functional

(i.e. a function of functions) as it depends on the auxiliary density. It can be shown that the difference between the lower bound and log marginal distribution is given by the Kullback-Leibler divergence [NH98; Bis13, pp. 450-451, p. 463][3]

$$\mathrm{KL}(q||p) = -\int q(X) \ln\left(\frac{p(X|Z)}{q(X)}\right) \mathrm{d}X, \tag{2.28}$$

and hence

$$\ln p(Z) = \mathcal{L}(q(X)) + \mathrm{KL}(q||p). \tag{2.29}$$

The Kullback-Leibler divergence satisfies $\mathrm{KL}(q||p) \geq 0$ and can be interpreted to measure the similarity of two distributions. It is zero if and only if $p(X|Z) = q(X)$. Therefore, the closeness of the lower bound $\mathcal{L}(q(X))$ to the actual log marginal distribution strongly depends on the chosen $q(X)$ and it is equal to $\ln p(Z)$ if the auxiliary density and the posterior distribution are equal.

Conversely, the closer the lower bound $\mathcal{L}(q(X))$ is to $\ln p(Z)$, the better $q(X)$ approximates the true posterior $p(X|Z)$. This relationship is exploited for Bayesian inference to obtain an approximate solution for the posterior density. For this purpose, $q(X)$ is restricted to a certain family of distributions and the lower bound $\mathcal{L}(q(X))$ is maximized with respect to $q(X)$. This only requires knowledge of $p(X, Z)$ and is equivalent to minimizing the Kullback-Leibler divergence $\mathrm{KL}(q||p)$. The optimization results in the optimal solution $q^*(X)$ which approximates the posterior distribution best for the chosen distribution family.

A particularly common class for the auxiliary density is that of factorized distributions. It is assumed that the auxiliary distribution over the latent variables can be factorized into c independent and disjoint subgroups X_i such that

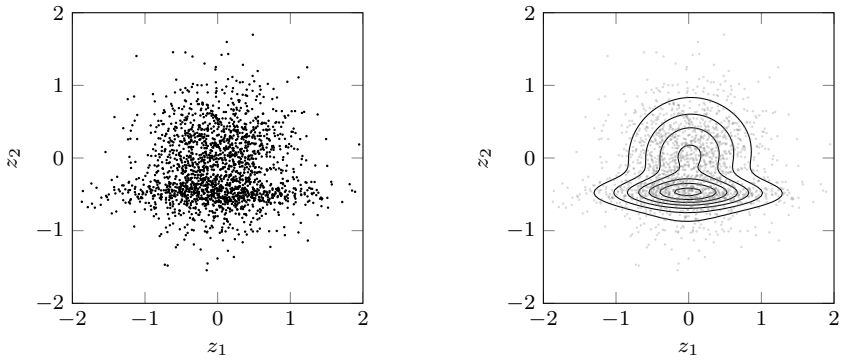
$$q(X) = \prod_{i=1}^{c} q_i(X_i). \tag{2.30}$$

This approximation scheme is oftentimes called mean field approximation due to its origin in mean field theory from theoretical physics [TLG08]. The optimal solution for $q^*(X)$ is then obtained by iteratively maximizing the lower bound $\mathcal{L}(q(X))$ with respect to the different factors $q_i(X_i)$. The optimal solution for a particular factor at a particular iteration step is [Bis13, pp. 465-466]

$$\ln q_j^*(X_j) = \mathbb{E}_{X_{i\neq j}}[\ln p(X, Z)] + \mathrm{const}, \tag{2.31}$$

where the expectation is computed with respect to all other variable subgroups $X_{i\neq j}$.

[3] In contrast to standard definitions of the Kullback Leibler divergence, the posterior density $p(X|Z)$ was directly inserted here for the application of variational Bayes.



(a) Observed data Z (b) Estimated predictive density $p(\tilde{z}|Z)$

Figure 2.2: Density estimation example: Observed data generated by a mixture of two Gaussians and resulting Student's t-mixture using the variational Gaussian mixture approach

2.4.2 Variational Gaussian Mixtures

One common application of variational Bayes is density estimation using Gaussian mixtures. Density estimation is a form of model inference, where the goal is to obtain a probabilistic model that explains the observed data Z, i.e. its distribution. Once the model is obtained, it can be used for further analysis or to make predictions about future data. Variational Gaussian mixtures (VGMs) have first been proposed for this application in [Att00; Att99] and are later employed to find a vehicle measurement model in Section 5.3.3.

Gaussian mixtures are a weighted superposition of multiple Gaussians and are a popular tool for modeling complex distributions. They provide great flexibility and are able to capture multi-modalities and rather complex relations, as long as the number of components is sufficient. When applied to density estimation, it is assumed that a Gaussian mixture model with unknown parameters generated the observed data. Then, the goal is to find the correct parameter values for the different components given the observed data. This can be done in a fully Bayesian approach using variational tools. Figure 2.2 provides an illustration for a two-dimensional example where the observed data was generated by a mixture of two Gaussian distributions. Using VGMs one can find a predictive density as shown in Fig. 2.2b which explains the observed data and allows predicting future values.

The following paragraphs sketch the different steps that are needed to formulate and solve the VGM problem. First, the probabilistic model, i.e. the Gaussian mixture

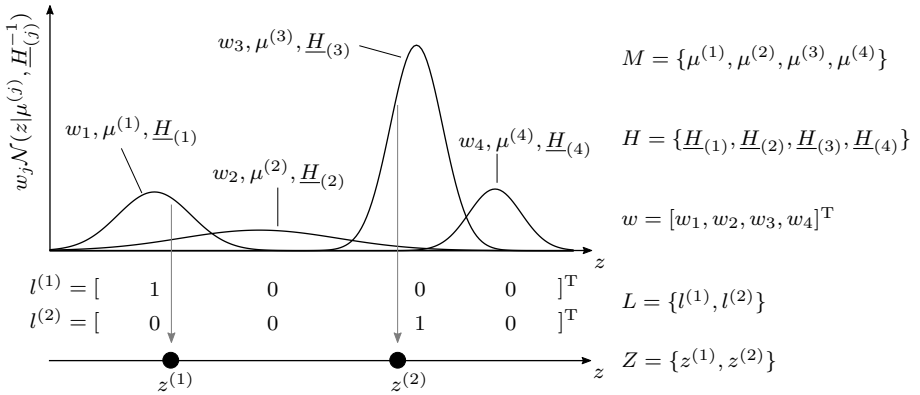


Figure 2.3: Schematic of a one-dimensional Gaussian mixture model with $c = 4$ components and parameters M, H, and w. The exemplary data set Z contains $m = 2$ measurements whose source components are encoded in the auxiliary variable set L.

model, is formulated. A schematic illustration of the Gaussian mixture model and the involved variables for a simplified one-dimensional case is shown in Fig. 2.3.

Each Gaussian distribution in a Gaussian mixture with c components is described by its mean $\mu^{(j)}$ and its precision matrix $\underline{H}_{(j)}$. The parameters of all components are combined in the sets $M = \{\mu^{(1)}, \ldots, \mu^{(c)}\}$ and $H = \{\underline{H}_{(1)}, \ldots, \underline{H}_{(c)}\}$, respectively. In the context of VGMs, it is assumed that each data point in the data set $Z = \{z^{(1)}, \ldots, z^{(m)}\}$ was generated independently by one of the c components in the mixture. This is encoded in the auxiliary variables $l^{(i)}$ which are c-dimensional binary 1-of-c vectors. That is, the entry corresponding to the component of origin is 1 and the remaining entries are 0. Again, the auxiliary variables of all m data points are combined in the set L. For given mixture parameters and auxiliary variables, the data likelihood is thus

$$p(Z|L, M, H) = \prod_{i=1}^{m} \prod_{j=1}^{c} \mathcal{N}(z^{(i)}|\mu^{(j)}, \underline{H}_{(j)}^{-1})^{l_j^{(i)}}. \tag{2.32}$$

As the data points are assumed to be independent, the above data likelihood simply multiplies the likelihoods of the individual data points. For each data point, the second product selects the Gaussian component that it originated from, as encoded in the auxiliary variable $l^{(i)}$. That is, all factors but one will become 1 as their auxiliary variables $l_j^{(i)}$ are zero.

Each component in the Gaussian mixture is assigned a mixing coefficient $0 \leq w_j \leq 1$

which indicates the contribution of the component to the mixture. The larger the mixing coefficient of a component, the more measurements this component will be responsible for. All mixing coefficients are combined in the c-dimensional vector w and sum to one. The distribution over the auxiliary variables L depends on the mixing coefficients and is a product of m multinomial distributions (cf. Section 2.2.2)

$$p(L|w) = \prod_{i=1}^{m} \text{Mult}(l^{(i)}|w) = \prod_{i=1}^{m} \prod_{j=1}^{c} w_j^{l_j^{(i)}}. \tag{2.33}$$

The two equations (2.32) and (2.33) constitute the underlying Gaussian mixture model and specify how measurements are created from the mixture if the parameters of the model, i.e. the mean vectors M, the precision matrices H, and the mixing coefficients w, are known. Note that the use of the auxiliary latent variables L makes the associations of measurements to their source component explicit and splits the model into two parts: One part, (2.33), for modeling which components created the respective measurements and another part, (2.32), for modeling which values they take, given the components of origin and their parameters.[4]

As stated before, density estimation using VGMs assumes that the observed data was generated by the above model. Yet, the parameters M, H, and w as well as the auxiliary variables L are unknown. To be able to compute their estimates in a Bayesian fashion, conjugate prior distributions are defined for each of them. In particular, the prior distribution over the mixing coefficients is chosen as the Dirichlet distribution

$$p(w) = \text{Dir}(w|\rho_0) = C(\rho_0) \prod_{j=1}^{c} w_j^{\rho_0 - 1}. \tag{2.35}$$

Here, the parameter ρ_0 is a scalar as the same value is used for all mixing coefficients. The prior over the mean and precision matrices is

$$p(M, H) = p(M|H)p(H) = \prod_{j=1}^{c} \mathcal{N}(\mu^{(j)}|\gamma_0, \beta_0^{-1}\underline{H}_{(j)}^{-1})\mathcal{W}(\underline{H}_{(j)}|\underline{V}_0, \nu_0) \tag{2.36}$$

and thus consists of an independent Gaussian-Wishart distribution for each component with parameters γ_0, β_0, $\underline{V}_0$, and ν_0. Together with the parameter of the Dirichlet prior ρ_0 they form the hyperparameters of the model.

[4] The classical formulation of a Gaussian mixture as a sum of weighted Gaussian distributions can be obtained from (2.32) and (2.33) by marginalizing over the auxiliary variables L. This yields

$$p(Z|w, M, H) = \sum_{L} p(Z|L, M, H)p(L|w) = \prod_{i=1}^{m} \sum_{j=1}^{c} w_j \mathcal{N}(z^{(i)}|\mu^{(j)}, \underline{H}_{(j)}^{-1}). \tag{2.34}$$

Now, the variational Bayes technique is used to determine an approximate posterior distribution $q^*(L, w, M, H) \approx p(L, w, M, H|Z)$ over the unknown parameters and auxiliary variables given the observed data. As a first step, the joint density over the unknown variables and the observed data, which is necessary for computing the lower bound from (2.27), is established. It is obtained by combining the data likelihood from (2.32), the distribution over the auxiliary variables (2.33), and the prior densities from (2.35) and (2.36), which yields

$$p(Z, L, w, M, H) = p(Z|L, M, H)p(L|w)p(w)p(M|H)p(H). \tag{2.37}$$

For maximizing the lower bound via the mean field approximation and (2.31), the form of the auxiliary density $q(L, w, M, H)$ needs to be specified. It is factorized into

$$q(L, w, M, H) = q(L)q(w, M, H). \tag{2.38}$$

The above factorization is a design choice that turns out to yield convenient analytical optimization terms. The optimal solutions are then computed by iteratively maximizing the lower bound with respect to $q(L)$ and $q(w, M, H)$ until a convergence criterion is reached. The structure of the resulting optimal auxiliary density which approximates the true posterior $p(L, w, M, H|Z)$ is

$$q^*(L, w, M, H) = q^*(L)q^*(w)q^*(M|H)q^*(H), \tag{2.39}$$

where the different factors are again of the same form as the distributions from (2.33), (2.35), and (2.36). That is, the optimal density over the auxiliary variables $q^*(L)$ is a product of multinomial distributions and the optimal density over the mixing coefficients $q^*(w)$ is a Dirichlet distribution with updated hyperparameter vector elements ρ_j. For each component of the Gaussian mixture, the optimal density over the mean vector and precision matrix is a Gaussian-Wishart density with updated hyperparameters β_j, $\underline{V}_{(j)}$, and ν_j. In contrast to the prior hyperparameters, the updated hyperparameters are different for each component and depend on the data that the component explains. For the sake of completeness, the optimal densities and equations of the updated parameters are presented in Appendix B.

In summary, the above procedure yields posterior distributions over the mixing coefficients of the mixture model, over the precision matrix and mean vector of each component, and over the auxiliary variables which indicate the source component of each measurement. This information can then be used to derive a predictive density

$$p(\tilde{z}|Z) = \sum_{\tilde{l}} \iiint p(\tilde{z}|\tilde{l}, M, H)p(\tilde{l}|w)q^*(w)q^*(M|H)q^*(H)\,\mathrm{d}w\,\mathrm{d}M\,\mathrm{d}H, \tag{2.40}$$

which reveals where a new measurement $\tilde{z}$ will most likely occur given the previously

observed data. It is obtained by first combining the mixture model equations (2.32) and (2.33) for the new measurement $\tilde{z}$ with its single auxiliary variable $\tilde{l}$ and the posterior densities of the model parameters from (2.39) to a joint density and marginalizing over the parameters and the auxiliary variable. The resulting predictive density is the multivariate Student's t-mixture [Bis13, pp. 482-483]

$$p(\tilde{z}|Z) = \frac{1}{\sum_{j=1}^{c} \rho_j} \sum_{j=1}^{c} \rho_j \, \mathrm{St}(\tilde{z}|\gamma^{(j)}, \underline{\Lambda}_{(j)}, \nu_j + 1 - n_{\tilde{z}}), \tag{2.41}$$

where $n_{\tilde{z}}$ denotes the dimension of $\tilde{z}$ and

$$\underline{\Lambda}_{(j)} = \frac{(\nu_j + 1 - n_{\tilde{z}})\beta_j}{1 + \beta_j} \underline{V}_{(j)}. \tag{2.42}$$

Interestingly, the predictive density from (2.41) is not a Gaussian mixture any more. This is due to the uncertainty in the model parameters that is captured by their posterior densities and is integrated in the predictive density. If one would instead extract MMSE estimates for the parameters from their posterior densities, neglect the uncertainty, and directly insert the estimates into the mixture model, one would end up with an approximate predictive density that is again in Gaussian mixture form.

An illustration of a predictive density is shown for the two-dimensional introductory example in Fig. 2.2b. Here, the number of mixture components c was set to five. In the resulting Student's t-mixture, however, only two components received relevant weights and contribute significantly to the result. Hence, the VGMs approach was able to identify the correct number of mixtures elements in the data-generating Gaussian mixture.

Chapter 3

Fundamentals of Object Tracking

Tracking is defined as the estimation of a moving object's state using extraneous or remote measurements [BLK01, p. 2]. The specification of the measurements being extraneous, i.e. from a sensor that is not mounted on the moving object itself, helps to distinguish tracking from ego-motion estimation using on-board sensors. While the definition does not restrict tracking to a certain technique, the probably most widely used methods are based on the Bayesian principle. In contrast to the inference methods from Chapter 2, however, Bayesian tracking is tailored to a more specific application. That is, the latent variables usually correspond to the state of an object, which for instance describes the pose and motion. Furthermore, the process is run iteratively to account for steadily arriving measurements, and an additional step that accounts for the object's motion is included.

Depending on the number of objects that are tracked, tracking is oftentimes categorized into single object or multi-object tracking. Alternatively, tracking algorithms can also be distinguished with respect to the number of measurements that are received from the objects. As mentioned in the introduction, most traditional approaches assume that each object generates a single measurement per time step, whereas extended object tracking allows for objects to give rise to multiple measurements. Merged measurements constitute another special case in which the sensor resolution is so low that multiple objects only give rise to a single measurement. This case, however, is not considered in this thesis. In general, the complexity of the tracking problem grows with the number of possible events and outcomes. Besides having to estimate the state of multiple objects, for example, multi-object tracking has to find out how many objects are present. Also, different possible associations between measurements and objects need to be considered, which is even more complicated in the extended object case.

This chapter provides the necessary tracking preliminaries for this thesis and an overview of existing approaches. It is intended to provide a basic understanding of

relevant concepts but does not assess the different concepts. The motivation for the chosen methods and developments of this work is later discussed in Chapter 4 with respect to the particular application of vehicle tracking.

Section 3.1 starts by introducing the basic formulation for the single-object case and the particle filter as a specific tracking method. Section 3.2 then extends the problem to the multi-object case and Section 3.3 presents multi-object tracking based on finite set statistics (FISST). Section 3.4 focuses on measurement models and algorithms for extended object tracking. Lastly, Section 3.5 presents an overview of tracking architectures for fusing measurements from multiple sensors.

3.1 Single Object Tracking

In the simplest case, the tracking problem consists of a single object which gives rise to exactly one measurement at each time step. This section starts by first presenting the Bayesian foundations and a brief overview of popular solutions before detailing particle filters approaches in Sections 3.1.2 and 3.1.3.

3.1.1 Problem Formulation

The goal is to recursively provide estimates of a moving object's state $x \in \mathbb{X}$ based on cyclic measurements z from a sensor. Whereas x referred to arbitrary latent variables in Chapter 2, it is confined to denoting the state of an object in the tracking context. This is due to the commonly used state space representation of the object model and the fact that the variables of interest are quantities from the object's state, e.g. its position, speed, or orientation. Analogously, $\mathbb{X}$ is referred to as the state space. The discrete-time system model is given by

$$x_k = f_m(x_{k-1}, \Delta t, \epsilon_{f,k-1}) \tag{3.1}$$

and

$$z_k = h_m(x_k, \epsilon_{h,k}), \tag{3.2}$$

for some initial value x_0 and with $k \in \mathbb{N}$ denoting the time index from the natural numbers $\mathbb{N}$. Note that the time instants represented by the different k correspond to the arrivals of new measurements and are not necessarily spaced equally in time. Therefore, difference between two consecutive time instants is encoded in Δt. Equation (3.1) is the object model and describes the object's behavior over time using the transition function $f_m(\cdot)$, where the subscript m is used for differentiation

from other functions. In contrast, the measurement model (3.2) relates the object state x_k to the current measurement z_k through the measurement function $h_m(\cdot)$. The arguments $\epsilon_{f,k-1}$ and $\epsilon_{h,k}$ model process and measurement noise sequences. While $f_m(\cdot)$ depends on the time difference, $h_m(\cdot)$ is assumed to be independent of the time throughout this thesis. Under the assumption of uncorrelated noise, (3.1) is a Markov process where the state from time step k only depends on the state and process noise from step $k-1$ and not on other previous states or noise values.

To obtain estimates of the object state recursively, the problem is cast in a Bayesian inference problem with two steps. The prediction step

$$p(x_k|z_{1:k-1}) = \int f(x_k|x_{k-1})p(x_{k-1}|z_{1:k-1})\,\mathrm{d}x_{k-1} \tag{3.3}$$

computes the prior state density $p(x_k|z_{1:k-1})$ based on the posterior density of the previous step $p(x_{k-1}|z_{1:k-1})$ and the transition density $f(x_k|x_{k-1})$. The notation $z_{1:k-1}$ implies that the prior for the state x_k incorporates the information from all previous measurements, i.e. from step 1 to $k-1$. The transition density models the change of the object state over time and results from the deterministic and stochastic portion of (3.1). Equation (3.3) is also referred to as the Chapman-Kolmogorov equation. In the update step, the posterior density of the object state x_k is computed using Bayes' theorem

$$p(x_k|z_{1:k}) = \frac{g(z_k|x_k)p(x_k|z_{1:k-1})}{\int g(z_k|x_k)p(x_k|z_{1:k-1})\,\mathrm{d}x_k}. \tag{3.4}$$

Here, the likelihood function $g(z_k|x_k)$ is a density obtained from (3.2). To be able to apply (3.3) and (3.4) iteratively, the filter is initialized with an initial density $p(x_0)$.

As in Chapter 2, the Bayes update is not analytically solvable in the general case. This additionally holds for the prediction step in (3.3) which also involves an integral over the previous posterior and the transition density. Thus, the state density needs to be a conjugate prior to the likelihood function and closed under the Chapman-Kolmogorov equation, if an analytical solution is desired. The Kalman filter [Kal60; BLK01, Section 5.2; RAG04, pp. 7-9] provides such a solution in the case of linear transition and measurement functions with zero-mean white Gaussian noise. It assumes that the initial state follows a Gaussian distribution and exploits that linear transformations of Gaussian distributions again result in Gaussian distributions as well as the conjugacy of Gaussian priors to Gaussian likelihood functions.

Yet, the assumptions made for deriving the Kalman filter do not hold for nonlinear systems or non-Gaussian process or measurement noise. Therefore, approximate solutions have been derived for such cases. The extended Kalman filter (EKF)

[BLK01, pp. 381-387], for example, linearizes the transition and measurement equations to be able to apply the linear Kalman filter routine. Both process and measurement noise are assumed to be additive white Gaussian noise. In contrast, the unscented Kalman filter (UKF) [JU97; Wv00] conducts an approximation of the resulting densities through sigma points instead of approximating the system model. Thus, the computation of Jacobians for linearization is avoided and noise that enters the system equations nonlinearly can be considered. Still, it is necessary that process and measurement noise are sufficiently described by their central second order moment, i.e. their covariance matrix.

3.1.2 Particle Filter

Particle filters provide another approach to nonlinear tracking problems based on importance sampling methods (cf. Section 2.3). In contrast to the Kalman filter and its derivatives, particle filters do not make assumptions about the nature of the process and measurement noise or linearity of the system model. Thus, they work for systems with strong nonlinearities or multi-modal noise. This is, however, achieved at the expense of a typically increased computational burden due to the involved sampling procedures.

The particle filtering technique uses sequential importance sampling which is an extension of importance sampling. That is, the posterior density is approximated by a set of n_p particles which are assigned weights. It can be shown [RAG04, pp. 38-39] that the particles approximate the true posterior by iteratively updating the particle weights according to

$$w(x_k^{(i)}) \propto w(x_{k-1}^{(i)}) \frac{g(z_k|x_k^{(i)}) f(x_k^{(i)}|x_{k-1}^{(i)})}{q(x_k^{(i)}|x_{k-1}^{(i)}, z_k)}, \tag{3.5}$$

where $q(x_k^{(i)}|x_{k-1}^{(i)}, z_k)$ is again the proposal density from which the particles are sampled and which may be conditioned on the last state and the current measurement. The approximate posterior is then

$$p(x_k|z_{1:k}) \approx \sum_{i=1}^{n_p} w(x_k^{(i)}) \delta\left(x_k - x_k^{(i)}\right), \tag{3.6}$$

with $\delta(\cdot)$ denoting the Kronecker delta.

While there are many different particle filter variants [RAG04, Section 3.5], the most common version is the sequential importance resampling (SIR) filter [GSS93; RAG04,

pp. 48-49; AMGC02]. It uses the transitional density $f(x_k^{(i)}|x_{k-1}^{(i)})$ as proposal density and proceeds in the following steps:

1. Draw samples $x_k^{(i)}$ from the proposal density $f(x_k^{(i)}|x_{k-1}^{(i)})$ by applying the transition function to the particles from the previous posterior.

2. Compute the particle weights by evaluating the likelihood function

$$w(x_k^{(i)}) \propto g(z_k|x_k^{(i)}) \tag{3.7}$$

   and normalize them using (2.22) to obtain the approximate posterior (3.6).

3. Resample the particles by drawing a new set of particles from (3.6) such that the probability of drawing a particular particle is $w(x_k^{(i)})$. Subsequently, set the new weights to $w(x_k^{(i)}) = 1/n_p$.

The resampling step helps to avoid the degeneracy problem which occurs if the number of particles with significant weight reduces over time until all but one particle have zero weights. This effect is unavoidable for particle weights in the form of (3.5); see [DGA00] for a theoretical explanation. By drawing new particles at each time step, particles with insignificant weight are pruned from the distribution and highly weighted particles are reproduced. In this thesis, the strategy of systematic resampling [RAG04, p. 42] is used.

As resampling always introduces a certain loss of information, the classical SIR scheme can be adapted such that resampling is only conducted if the estimate of the effective sample size [RAG04, p. 40]

$$\hat{n}_{\text{eff}} = \frac{1}{\sum_{i=1}^{n_p} w(x_k^{(i)})^2} \tag{3.8}$$

falls below a certain threshold. In case the resampling step is not executed, the subsequent computation of the particle weights has to be modified to $w(x_k^{(i)}) \propto w(x_{k-1}^{(i)}) g(z_k|x_k^{(i)})$.

3.1.3 Rao-Blackwellized Particle Filter

The number of particles is a crucial factor for the computational demands of particle filters. It has been reported that the required number of particles depends on the dimension of the state space as well as other factors, see [RAG04, pp. 58-59;

DdMR00; DH03; GSS93], and computations may become rather involved for large systems. Rao-Blackwellized particle filters (RBPFs) [DdMR00; DGA00] alleviate this issue by reducing the space that is sampled from through marginalization of analytically solvable substructures. For this purpose, the state vector is split into two parts $x_k = [x_{1,k}^{\mathrm{T}}, x_{2,k}^{\mathrm{T}}]^{\mathrm{T}}$ and it is assumed that the posterior density can be written in the form

$$p(x_{1,k}, x_{2,k}|z_{1:k}) = p(x_{2,k}|x_{1,k}, z_{1:k})p(x_{1,k}|z_{1:k}), \tag{3.9}$$

i.e. that one is able to find suitable models for such a factorization. Under the assumption that computing the posterior density $p(x_{2,k}|x_{1,k}, z_{1:k})$ is analytically tractable, e.g. for a linear Gaussian or discrete substructure of $x_{2,k}$, only the distribution of $x_{1,k}$ is approximated by samples. This yields

$$p(x_{1,k}, x_{2,k}|z_{1:k}) \approx \sum_{i=1}^{n_p} w(x_{1,k}^{(i)})\delta\left(x_{1,k} - x_{1,k}^{(i)}\right) p_i(x_{2,k}|x_{1,k}^{(i)}, z_{1:k}). \tag{3.10}$$

Intuitively, each particle represents a particular realization of $x_{1,k}$ and holds an analytical density over $x_{2,k}$ conditioned on its value $x_{1,k}$. The densities $p_i(x_{2,k}|x_{1,k}^{(i)}, z_{1:k})$ are then analytically updated for each particle while the particle weights are updated using the classical importance sampling scheme. Here, the subscript i has been introduced for the analytical densities to emphasize that the densities are distinct for each particle. For the proposal density $p(x_{1,k}|x_{1,k-1})$, the particle weights are

$$w(x_{1,k}^{(i)}) \propto g(z_k|x_{1,k}^{(i)}) = \int g(z_k|x_{1,k}^{(i)}, x_{2,k})p_i(x_{2,k}|x_{1,k}^{(i)}, z_{1:k-1})\,\mathrm{d}x_{2,k}. \tag{3.11}$$

That is, they are determined by marginalizing the uncertainty in $x_{2,k}$.

3.2 Overview of Multi-Object Tracking

In multi-object tracking, the problem from the previous section is extended by assuming the presence of not only one but multiple objects. Therefore, multi-object tracking approaches need to solve the far more complex problem of estimating the state of multiple objects as well as their number given a set of measurements. There may be misdetections and measurements may either originate from the objects or be clutter measurements that are caused by other irrelevant objects or by sensor noise. Moreover, the number of objects may vary as objects enter or leave the sensor field of view (FOV) and objects may influence each other or their measurement generation process, e.g. in case of occlusion.

One of the greatest underlying challenges is measurement-to-object association, i.e. determining which object is updated with which measurement and which measurements are considered clutter. The simplest solutions is to combine multiple single object filters and to include a greedy nearest neighbor (NN) association step, which assigns the closest measurement to each object. Yet, this method has, for example, been reported to fail in cases where objects are so close to each other (see e.g. [FBS83]) that a measurement is assigned to multiple objects or that there is considerable ambiguity in the correct association decision. Hence, different approaches for rigorously solving the multi-object problem have been developed in the literature. The following paragraphs provide a brief outline of two widespread approaches to multi-object tracking, namely joint probabilistic data association (JPDA) and multiple hypotheses tracking (MHT). The third approach, which is based on FISST and is used in this thesis, is presented in more detail in the following section. A more comprehensive overview of multi-object tracking can be found in [VMB+15].

Probabilistic data association (PDA) was first used for tracking a single target in the presence of multiple measurements, i.e. one object measurement and additional clutter measurements, in [BT75]. As the correct association is unknown, association probabilities are computed and a weighted update using all measurements is conducted. The likelier an association, the greater is the influence of the corresponding measurement during update. The JPDA filter [FBS83] is an extension of the PDA approach to multi-object tracking. It computes the association probabilities jointly for all present targets while assuming that each measurement may only be assigned to one object or to clutter. In the JPDA filter, the object number is assumed to be known [VMB+15] which implies that all tracked objects exits. The joint integrated probabilistic data association (JIPDA) filter [ME02; ME04] additionally incorporates an existence probability for each object which serves as a quality measure and can be used for principled track management.

In contrast, MHT approaches construct different association hypotheses which are then evaluated individually instead of using a weighted update. This results in different object hypotheses that represent variants of how the trajectory may have evolved. These hypotheses are propagated over several time steps which allows for resolving uncertainty in previous situations as soon as new measurements are able to provide the missing information. This leads to a deferred decision logic [Bla04]. To avoid combinatorial explosion, MHT approaches typically use pruning of insignificant hypotheses or gating and clustering to only construct necessary and meaningful hypotheses. While ideas of association chains with different branches date back to [Sit64] and the propagation of multiple hypotheses in combination with suboptimal variants have been investigated in [SSH74], [Rei79] is considered the first full systematic MHT approach. An extensive overview of different formulations and variants of MHT is presented in [Bla04].

3.3 Multi-Object Tracking Using Finite Set Statistics

JPDA and MHT methods have been used for several decades and essentially try to find solutions to the association problem which are then combined with classical single-object tracking techniques for each object. In contrast, multi-object tracking based on FISST is a rather recent approach which provides a fundamental formulation of the multi-object tracking problem in a Bayesian sense. From this formulation, different multi-object tracking algorithms can be derived. This section presents FISST preliminaries and tracking algorithms necessary for understanding this thesis. Concise tutorials on the topic can be found in [Mah04; Mah13] and for extensive details, refer to [Mah07a; Mah14].

3.3.1 Multi-Object Bayes Filter

In FISST-based multi-object tracking, the multi-object state is described by the random finite set (RFS) $X = \{x^{(1)}, \ldots, x^{(|X|)}\}$ with the elements representing the object states. An RFS is a set-valued random variable and its realizations may be any finite subset of the state space $\mathbb{X}$ or the empty set $\emptyset$. That is, all elements are from $\mathbb{X}$ and the number of elements in the set, its cardinality $|X|$, may be 0 or any positive integer. Note that in contrast to the sets of random variables from Section 2.4, which were a collection of a fixed number of random vectors, the number of elements in an RFS is also a random variable. Therefore, this representation allows capturing both the uncertainty in the object states as well as in the number of objects. Likewise, the measurements at each time step are combined in the measurement set $Z = \{z^{(1)}, \ldots, Z^{(|Z|)}\}$ which is again an RFS that may take the value of any finite subset (including $\emptyset$) of the measurement space $\mathbb{Z}$. Thus, the variation in the number of measurements at each time step can be captured.

As in single object tracking, the goal is to recursively compute the posterior distribution over X_k given all measurement sets $Z_{1:k}$ that have been received so far. Similarly to (3.3) and (3.4), the multi-object Bayes filter [Mah07a, Chapter 14] proceeds in two steps. In the prediction step, the prior density

$$\pi(X_k|Z_{1:k-1}) = \int f(X_k|X_{k-1})\pi(X_{k-1}|Z_{1:k-1})\delta X_{k-1} \tag{3.12}$$

is determined using the multi-object transition density $f(X_k|X_{k-1})$ and the posterior multi-object density from time step $k-1$, $\pi(X_{k-1}|Z_{1:k-1})$. The new measurements Z_k are then incorporated through the multi-object likelihood function $g(Z_k|X_k)$ in

the update step. This yields the posterior multi-object density

$$\pi(X_k|Z_{1:k}) = \frac{g(Z_k|X_k)\pi(X_k|Z_{1:k-1})}{\int g(Z_k|X_k)\pi(X_k|Z_{1:k-1})\delta X_k}. \tag{3.13}$$

Despite the close resemblance to the single object Bayes filter from Section 3.1.1, (3.12) and (3.13) exhibit considerable differences:

1. The involved densities are densities over set-valued random variables and not random vectors.

2. The integrals are set integrals which are indicated by the differential operator δ and defined in the following section.

3. In contrast to the transition density in (3.3), which only models the evolution of the state, the multi-object transition density $f(X_k|X_{k-1})$ may also govern the evolution of the number of objects, e.g. through object appearance or disappearance. It may also model dependencies among the present objects.

4. The multi-object likelihood function $g(Z_k|X_k)$ describes the entire measurement generation process which can, for example, include misdetections or clutter measurements.

3.3.2 Multi-Object Distributions

As for standard random variables, there are different distributions that can be used to model the probabilistic characteristics of an RFS. This section first presents some basic properties of multi-object densities and then presents four common types.

A multi-object density $\pi(X) = \pi(\{x^{(1)}, \ldots, x^{(|X|)}\})$ captures both the uncertainty in the values of the set elements as well as their number. Note that the argument of the density is a set. Alternatively, multi-object densities can also be written in vector notation

$$\pi(x^{(1)}, \ldots, x^{(|X|)}) = \begin{cases} \frac{1}{|X|!}\pi(\{x^{(1)}, \ldots, x^{(|X|)}\}), & \text{for distinct } x^{(i)}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.14}$$

The difference in both notations lies in the factorial. It arises because sets are unordered and there are $|X|!$ different possibilities to obtain ordered vectors from the set.

Set integrals of multi-object densities as used in (3.12) and (3.13) are defined as [Mah07a, p. 361]

$$\begin{aligned}\int \pi(X)\delta X &= \sum_{i=0}^{\infty} \frac{1}{i!} \int_{\mathbb{X}^i} \pi(\{x^{(1)}, \ldots, x^{(i)}\})\, \mathrm{d}x^{(1)} \ldots \mathrm{d}x^{(i)} \\ &= \pi(\emptyset) + \sum_{i=1}^{\infty} \int_{\mathbb{X}^i} \pi(x^{(1)}, \ldots, x^{(i)})\, \mathrm{d}x^{(1)} \ldots \mathrm{d}x^{(i)}. \end{aligned} \tag{3.15}$$

As in standard probability theory, the set integral of a multi-object density has to equal one when integrating over the entire sample space [Mah07a, p. 362].

The cardinality distribution, i.e. the distribution over the number of elements in a set, is obtained by integrating over the state space [Mah07a, p. 363],

$$p(|X|) = \frac{1}{|X|!} \int \pi(\{x^{(1)}, \ldots, x^{(|X|)}\})\, \mathrm{d}x^{(1)} \ldots \mathrm{d}x^{|X|}. \tag{3.16}$$

In FISST, the first-order moment of a multi-object density is called the probability hypothesis density (PHD). It is an intensity function over the state space which measures how many objects are present in a particular area of the state space and integrates to the expected number of objects in that area. As PHDs do not play a central role in this thesis, the reader is referred to [Mah07a, pp. 568-569] for more information.

Bernoulli RFS

A Bernoulli RFS X is one of the simplest multi-object densities as it only takes two cardinality values. With probability r, the set either contains one element, whose state is distributed according to some density $p(x)$, or it is empty with probability $1 - r$. The multi-object density is thus [Mah14, pp. 100-101]

$$\pi(X) = \begin{cases} 1 - r, & \text{if } X = \emptyset \\ r \cdot p(x), & \text{if } X = \{x\} \\ 0, & \text{if } |X| \geq 2 \end{cases}. \tag{3.17}$$

The cardinality distribution is $p(|X| = 0) = 1 - r$ and $p(|X| = 1) = r$.

Multi-Bernoulli RFS

The previous concept can be extended to RFSs with multiple elements by forming a union of multiple Bernoulli RFSs. This yields a combined parameter set $\{(r^{(i)}, p^{(i)}(x))\}_{1 \leq i \leq c}$ which specifies each of the c components' existence probability $r^{(i)}$ and density $p^{(i)}(x)$. The corresponding multi-object density for $|X| \leq c$ is (cf. [Mah14, p. 102])

$$\pi(X) = \sum_{\theta \in \Theta} \prod_{\theta(i)=0} \left(1 - r^{(i)}\right) \prod_{\theta(i)>0} \left(r^{(i)} p^{(i)}(x^{(\theta(i))})\right), \tag{3.18}$$

where the summation is over all possible association mappings $\theta : \{1, \ldots, c\} \to \{0, \ldots, |X|\}$ from the association space Θ. The association mappings assign each Bernoulli component to either one element of the set $|X|$ or to the event of being empty which is encoded by the value 0. The association mappings are due to the uncertainty in which component gave rise to which set element and construct different hypotheses. To ensure that only one component is assigned to an element of X, it is postulated that $\theta(i) = \theta(j)$ implies $i = j$. The cardinality distribution of a multi-Bernoulli RFS is a Poisson binomial distribution [Wan93]. It is unimodal and its mean is $\mathbb{E}[|X|] = \sum_{i=1}^{c} r^{(i)}$.

Independent Identically Distributed RFS

In contrast to multi-Bernoulli RFSs, independent identically distributed RFS are more flexible in the cardinality distribution but more restrictive in the distribution of the element values. In particular, they suppose that the cardinality follows some arbitrary cardinality distribution $p(|X|)$ and that all elements are independent and identically distributed (i.i.d.) according to some common density $p(x)$—hence the name. The multi-object density is given by [Mah14, p. 100]

$$\pi(X) = |X|!\, p(|X|) \prod_{x \in X} p(x) = |X|!\, p(|X|)[p(\cdot)]^X. \tag{3.19}$$

The second equality introduces the multi-object exponential notation

$$[h(\cdot)]^X \triangleq \prod_{x \in X} h(x), \quad [h(\cdot)]^{\emptyset} = 1, \tag{3.20}$$

which is a short notation for products of real-valued functions $h(\cdot)$ applied to all elements of a set.

Poisson RFS

A Poisson RFS is a special form of an independent identically distributed RFS and uses a Poisson distribution for the cardinality. Inserting the Poisson distribution into (3.19) yields [Mah07a, p. 366]

$$\pi(X) = e^{-\lambda}[\lambda p(\cdot)]^X, \tag{3.21}$$

where $\lambda p(\cdot)$ is also referred to a as the intensity function.

3.3.3 Labeled Multi-Object Distributions

In tracking applications, one is not only interested in obtaining an estimate of the objects and their states at the current time but also in the trajectories of the objects over time. Hence, the notion of object identity which allows to link estimates over multiple time steps becomes relevant. So far, however, the presented RFSs were defined on the state space $\mathbb{X}$. Thus, the set elements typically comprise kinematic states such as position or speed and do not provide a means of identification. In particular, the RFS elements are unordered and therefore, identity cannot be assigned from a particular position in the set. Yet, element identity can be introduced by augmenting the object states with an additional label state.

This concept has been rigorously formulated in [VV13] and was termed labeled RFSs. A labeled RFS $\boldsymbol{X} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(|\boldsymbol{X}|)}\}$ is an RFS defined on $\mathbb{X} \times \mathbb{L}$ which contains labeled states $\boldsymbol{x} = [x^{\mathrm{T}}, \ell]^{\mathrm{T}} \in \mathbb{X} \times \mathbb{L}$. Note that labeled quantities and their densities are notationally distinguished from unlabeled RFSs, states, and densities by using bold letters. The labels $\ell \in \mathbb{L}$ are taken from the label space $\mathbb{L}$ which is a collection of distinct positive integers. The set of labels occurring in a labeled RFS can be retrieved by using the projection function

$$\mathcal{L}(\boldsymbol{X}) = \{\ell \mid [x^{\mathrm{T}}, \ell]^{\mathrm{T}} \in \boldsymbol{X}\}, \tag{3.22}$$

which iterates over all labeled states in the set and extracts their labels. An additional requirement for a a labeled RFSs is that its labels be distinct. This requirement ensures unique identities and can be mathematically expressed using the distinct label indicator

$$\Delta(\boldsymbol{X}) = \delta(|\mathcal{L}(\boldsymbol{X})| - |\boldsymbol{X}|) \tag{3.23}$$

which is one if and only if the number of labels equals the number of RFS elements[1].

[1] This notation uses the fact that sets only contain distinct elements and that the cardinality of a set does not count multiple occurrences of a particular element.

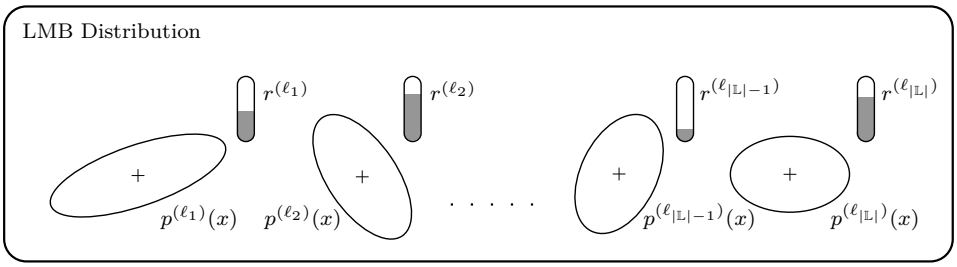


Figure 3.1: Schematic illustration of an LMB distribution

As the elements of labeled RFSs contain discrete labels, the set integral over a labeled RFS is [VV13]

$$\int \pi(\boldsymbol{X})\delta\boldsymbol{X} = \sum_{i=0}^{\infty} \frac{1}{i!} \sum_{[\ell^{(1)},\ldots,\ell^{(i)}]\in\mathbb{L}^i} \int_{\mathbb{X}^i} \pi(\{\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(i)}\})\,\mathrm{d}x^{(1)}\ldots\mathrm{d}x^{(i)}. \tag{3.24}$$

Having defined labeled RFSs, two particular labeled multi-object distributions are presented in the following. They have been proposed in [VV13] and have proven to be useful for multi-object tracking with object identities.

Labeled Multi-Bernoulli Distribution

A labeled multi-Bernoulli (LMB) distribution [VV13] is the extension of a multi-Bernoulli distribution to labeled RFSs. Again, the parameter set of the distribution $\{(r^{(\ell)}, p^{(\ell)}(x))\}_{\ell\in\mathbb{L}}$ contains the existence probability $r^{(\ell)}$ and state density $p^{(\ell)}(x)$ of each component and the components are independent of each other. This formulation equates the Bernoulli component identity with the label and it is assumed that if component ℓ gives rise to an element (i.e. it is not empty), the label of the element is ℓ. It also implies that the label space $\mathbb{L}$ defines the number of components that are present in the LMB distribution. A schematic illustration of an LMB distribution is shown in Fig. 3.1. It illustrates the distribution as a collection of independent pairs of state densities and existence probabilities which are depicted by ellipses and gauges, respectively.

The multi-object density is

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X})w(\mathcal{L}(\boldsymbol{X}))[p(\cdot)]^{\boldsymbol{X}} \tag{3.25}$$

with

$$w(I) = \prod_{i \in \mathbb{L}} \left(1 - r^{(i)}\right) \prod_{\ell \in I} \frac{1_{\mathbb{L}}(\ell) r^{(\ell)}}{1 - r^{(\ell)}}. \tag{3.26}$$

Here, the identity $p(x, \ell) = p^{(\ell)}(x)$ was used and the inclusion function $1_{\mathbb{L}}(\ell)$ which is equal to 1 if and only if $\ell \in \mathbb{L}$ is used to assign zero probability to labeled RFSs with nonexistent labels. Equation (3.25) still resembles the multi-Bernoulli density from (3.18). It does, however, differ in the additional elements for ensuring distinct and valid labels. Also, it avoids association mappings as the association between the Bernoulli components and set elements is implicitly given by the labels. The compact notation furthermore separates the densities over the element values from the weight $w(\mathcal{L}(\boldsymbol{X}))$ which computes the probability that all elements in $\boldsymbol{X}$ exist. As the labels do not affect the cardinality distribution, it is a unimodal Poisson binomial distribution as for multi-Bernoulli distributions.

One of the major limitations of LMB distributions is that the Bernoulli components are assumed independent and dependence among objects can thus not be modeled. For example, statements such as "either both objects are present or none" cannot be accurately represented as they would require a multi-modal cardinality distribution.

Generalized Labeled Multi-Bernoulli Distribution

Generalized labeled multi-Bernoulli (GLMB) distributions [VV13] overcome the limitations of LMB distributions and provide more flexibility. Their multi-object density is

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{c \in \mathbb{C}} w^{(c)}(\mathcal{L}(\boldsymbol{X})) \left[p^{(c)}(\cdot)\right]^{\boldsymbol{X}} \tag{3.27}$$

where the weights have to satisfy

$$\sum_{I \subseteq \mathbb{L}} \sum_{c \in \mathbb{C}} w^{(c)}(I) = 1. \tag{3.28}$$

There are two major differences to LMB distributions: First, the weights are not as strictly defined as in an LMB distribution and not limited to a multiplication of independent components. Secondly, the density allows for multiple realizations c from the space of possible realizations $\mathbb{C}$.

Since the definition of the GLMB distribution is rather abstract and difficult to imagine, Fig. 3.2 illustrates the special case of a GLMB distribution with two components and only one realization c. The distribution holds the state densities of the two components and the weights for all possible label combinations, i.e. $w(\{\emptyset\})$,

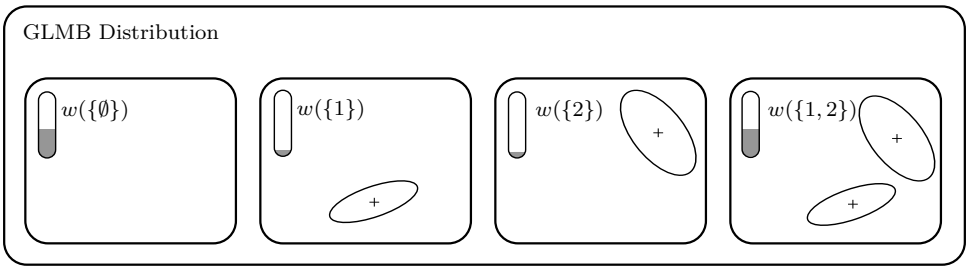


Figure 3.2: Illustration of a GLMB distribution with two objects and one realization

$w(\{1\})$, $w(\{2\})$, and $w(\{1,2\})$. These weights indicate how likely the particular combinations of objects are and may be chosen arbitrarily as long as (3.28) is satisfied. Intuitively, such a GLMB distribution can thus be interpreted as a collection of weighted multi-object state hypotheses that define which objects are present and which are not. In contrast to the LMB distribution, the freedom in choosing the weights allows capturing object dependence and, for instance, to model the multi-modal example of either both or no object being present.

By introducing multiple realizations, the GLMB definition permits the inclusion of additional multi-object hypothesis variants with distinct weights and state densities. These variants allow including several outcomes with respect to some discrete event c. In Fig. 3.2, adding different realizations to the distribution would add new rows of hypotheses that are indexed by different values for c and in which state densities and weights differ. For example, realizations can be used to model different association possibilities which lead to different posterior state densities and weights. The relationship is then indicated by indexing the weights and densities with the association mapping.

δ-generalized labeled multi-Bernoulli (δ-GLMB) distributions [VV13] are a special case of GLMB distributions which specify particular forms for the realizations and weights that are useful for emphasizing association variants and efficient ways for implementation. In this work, however, the derivations use the general notation of GLMB distributions and the reader is hence referred to [VV13] for more information.

Conversion Between LMB and GLMB Distributions

An LMB distribution constitutes a special case of a GLMB distribution with a single realization and weights as defined by (3.26). Consequently, an LMB distribution can

be transformed to the representation of Fig. 3.2 by enumerating all possible label combinations I and computing the weights $w(I)$ through (3.26).

Due to the ability of GLMB distributions to represent more complex constellations with object dependencies, however, a direct conversion from a GLMB distribution to an LMB distribution is not possible in the general case. Instead, an approximation scheme is necessary. Such a scheme was proposed for the special case of δ-GLMB distributions in [RVVD14; Reu14, pp. 95-97] and used for GLMB distributions in [BRG$^+$16]. It preserves the first-order moment, i.e. the PHD, and involves computing the existence probability of each object

$$r^{(\ell)} = \sum_{I \subseteq \mathbb{L}} \sum_{c \in \mathbb{C}} w^{(c)}(I) 1_I(\ell) \tag{3.29}$$

by summing the weights of all components that contain object ℓ. Additionally, the state densities are obtained as the mixture distribution

$$p^{(\ell)}(x) = \frac{1}{r^{(\ell)}} \sum_{I \subseteq \mathbb{L}} \sum_{c \in \mathbb{C}} w^{(c)}(I) 1_I(\ell) p^{(c)}(x, \ell). \tag{3.30}$$

Since the resulting LMB distribution with parameters from (3.29) and (3.30) assumes independent objects, this approximation leads to a loss of information about object dependencies. See [Reu14, pp. 97-100] for a detailed discussion.

3.3.4 Multi-Object Filters for Tracking

Just as the standard Bayes filter from Section 3.1.1, the multi-object Bayes filter from (3.12) and (3.13) is in most cases computationally intractable. This is again due to the integrals that have to be evaluated and further complicated by the fact that the integrals are now computed over sets. Therefore, several tractable multi-object filters that are approximate or based on simplifying assumptions have been proposed. Before giving a short overview of existing filters, the standard multi-object transition model and multi-object likelihood function, which form the basis for most of these multi-object filters, are presented.

Standard Multi-Object Transition Model

The full standard multi-object transition model as defined in [Mah07a, pp. 472-473] considers four different cases of object behavior: object appearance, object disappearance, object motion, and object spawning. The latter case is a special form

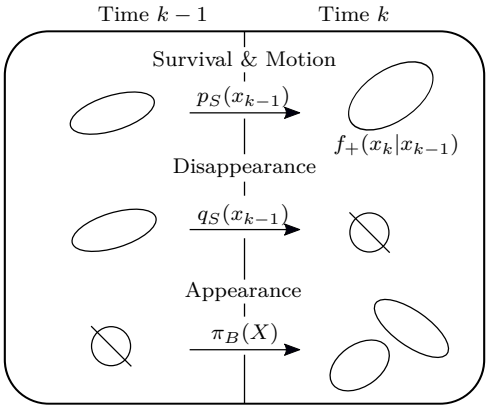


Figure 3.3: Three cases handled by the standard multi-object transition model: survival and motion, disappearance, and appearance (cf. [Mah07a, p. 478])

of object appearance. It is used for cases where new objects emerge from an existing object and allows conditioning the number of spawned objects and their state on the parent object. This is oftentimes of interest in military applications where, for example, missiles emerge from fighter jets. Since such cases are not of interest for the applications considered in this thesis, the case of spawning is neglected.

The remaining three cases are illustrated in Fig. 3.3 and are based on the following assumptions:

1. An existing object x_{k-1} may survive to the next time step k with the persistence probability[2] $p_S(x_{k-1})$. It may disappear with complementary probability $q_S(x_{k-1}) = 1 - p_S(x_{k-1})$.

2. If an object survives, its state evolves according to the single object transition density $f_+(x_k|x_{k-1})$.

3. New objects appear according to some birth density $\pi_B(X)$.

4. All objects move, appear, or disappear independently.

[2]Note that $p_S(x_{k-1})$ is not a density over x_{k-1} but a function of x_{k-1} that specifies the probability of survival.

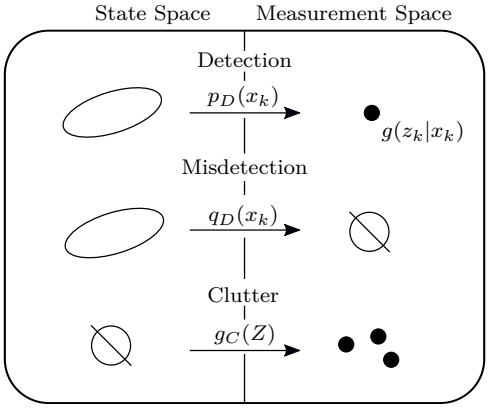


Figure 3.4: Three cases handled by the standard multi-object likelihood: detection, misdetection, and clutter (cf. [Mah07a, p. 409])

Standard Multi-Object Likelihood

The standard multi-object likelihood [Mah07a, pp. 420-421], cf. Fig. 3.4, is similar in its structure and considers three cases: object detection, misdetections, and clutter. The underlying assumptions are

1. An object may be detected by the sensor with the detection probability[3] $p_D(x_k)$ or it is not detected (misdetection) with complementary probability $q_D(x_k) = 1 - p_D(x_k)$.

2. If detected, the object gives rise to one measurement z_k which follows the single object likelihood function $g(z_k|x_k)$.

3. The measurement set Z is a union of object and clutter measurements and the clutter measurements follow a Poisson RFS $g_C(\cdot)$ with intensity function $\kappa(z_k) = \lambda_C p_C(z_k)$. That is, the expected number of clutter measurements is λ_C and they are i.i.d. according to the clutter density $p_C(z_k)$.

4. Each object measurement is generated by a single object only and the object measurements as well as clutter measurements are (conditionally) independent.

[3] Again, the detection probability $p_D(x_k)$ is a function of the state that outputs a probability value and not a density.

Overview of Multi-Object Filter Variants

As in standard Bayesian inference and single-object tracking, there are different strategies for finding tractable solutions to the multi-object Bayes filter. The following paragraphs provide a brief overview of different approaches. A more detailed comparison including a table summarizing the different characteristics is given in [Reu14, Appendix C].

PHD filters [Mah03], for example, only propagate the PHD of the multi-object distribution. This moment approximation avoids the computation of complex set integrals but there is no information about track identities and the cardinality estimates are unstable. Cardinalized probability hypothesis density (CPHD) filters [Mah07b] are an extension to PHD filters and try to overcome this limitation by additionally propagating the cardinality distribution.

An alternative is to apply sampling-based methods and to approximate the multi-object posterior by particles as proposed in [VSD03; VSD05]. In contrast to classical particle filters, the particles represent the full multi-object state and may thus have different dimensionality. While such sampling approaches allow for more complicated transition and likelihood models, the high-dimensional spaces require a large number of particles which may lead to practical performance problems [VSD05]. A real-time capable implementation based on a simplified multi-object likelihood function has been presented in [RWW+13].

In contrast to the moment approximation and sampling-based approach, other filters choose a particular type of multi-object distribution and propagate their parameters. This is for example done in the cardinality-balanced multi-target multi-Bernoulli (CBMeMBer) filter [VVC09] which models the multi-object distribution as a multi-Bernoulli distribution. Yet, the multi-Bernoulli distribution is not a conjugate prior to the standard multi-object likelihood function and an approximation step is necessary to again obtain a posterior density in multi-Bernoulli form.

The concept of using a particular form for the multi-object distribution has also been transferred to labeled RFSs in [VV13]. The presented equations for the GLMB and δ-GLMB filters[4] explicitly consider object identity. Also, they make use of GLMB-type distributions being closed under the standard multi-object transition (i.e. the resulting density is again in GLMB form) and that they are a conjugate prior to the standard multi-object likelihood function (again in terms of the resulting posterior density being of the same form). Thus, these filters are able to propagate the full multi-object distribution and do, in contrast to the CBMeMBer filter, not

[4] Note that the GLMB and δ-GLMB filters are conceptually identical but differ in the used notation. The δ-GLMB formulation accentuates an efficient implementation.

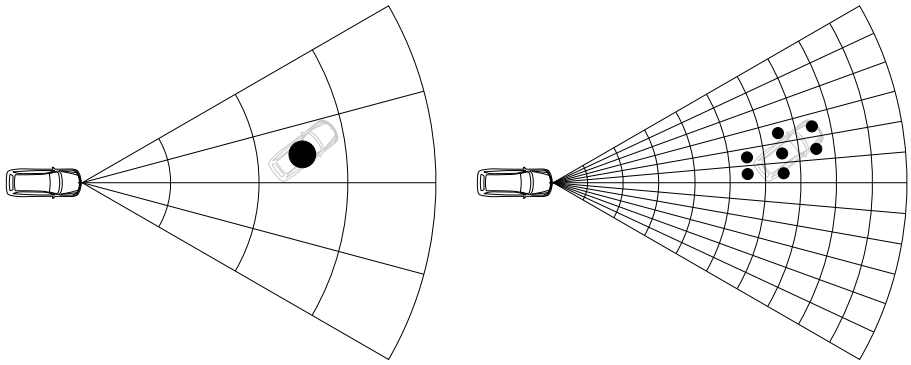
include an approximation step during update. Yet, a major drawback of the GLMB and δ-GLMB filters is that the number of components in the multi-object distribution increases exponentially over time (cf. [RVVD14; VV13]). The LMB filter [RVVD14] diminishes this issue by combining the ideas from the CBMeMBer and GLMB-type filters. It uses an LMB distribution, which provides the notion of object identity and is closed under the standard multi-object transition, for modeling the multi-object density. As the update step yields a GLMB distribution for an LMB prior, an approximation step which reduces the posterior to an LMB distribution with matching first-order moment is introduced (cf. Section 3.3.3).

Poisson multi-Bernoulli mixtures (PMBMs) [GWGS18; Wil15a] constitute another conjugate prior to the standard measurement model and are closed under the prediction when using a Poisson RFS as birth density. The PMBM distribution consists of a Poisson RFS that models the distribution of undetected objects and a mixture of multi-Bernoulli distributions for detected objects. The weights of the different mixture components depend on the different association hypotheses that can be constructed. As for the GLMB filter, the number of mixture components increases drastically over time and different approximation variants have thus been proposed in [Wil15a; Wil15b]. The use of a Poisson RFS for undetected objects helps to incorporate information about the number of such objects and where they could be located (e.g. in occluded areas) [GBR17; Wil15a].

3.4 Extended Object Tracking

The aforementioned multi-object tracking approaches extend the standard single object tracking problem from Section 3.1 in two ways. First, they consider multiple objects and multiple measurements at the same time and secondly, they explicitly model effects such as object appearance, disappearance, clutter, and misdetections. Yet, they are still based on the assumption that the object size is negligible with respect to sensor resolution and an object gives rise to at most one measurement. This is called the point object assumption and illustrated in Fig. 3.5a.

If, however, the object size exceeds sensor resolution (cf. Fig. 3.5b), the point object assumption is no longer valid and multiple measurements may be received. As previously established in the introduction, tracking algorithms are faced with the extended object problem in such cases. To avoid the disadvantages of preprocessing routines that were mentioned in the introduction, i.e. the loss of information and difficulties in ambiguous situations, a variety of new single-object measurement models that are able to process multiple measurements at a time and multi-object filters that are able to accommodate such measurements models have been developed.



(a) Point object (b) Extended object

Figure 3.5: Schematic comparison of point and extended objects in dependence on the sensor resolution

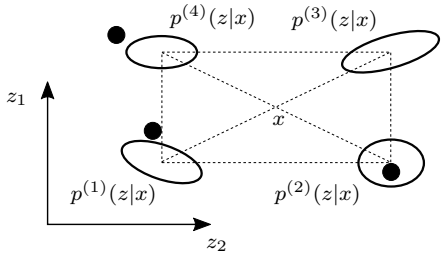
The following subsections provide a short summary of the basic and most known approaches. A more complete overview of different measurement models and multi-object filters for extended object tracking is given in [GBR17].

3.4.1 Measurement Models for Extended Object Tracking

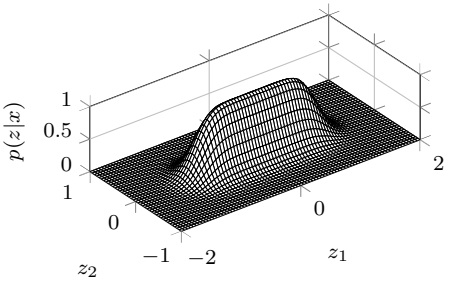
Measurement models for a single extended object can be constructed in different ways. This section reviews three common approaches that use a set of reflection points, define a spatial distribution, or estimate the shape using a contour model.

Reflection Point Models

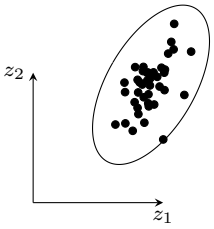
Reflection point models assume that an extended object consists of several distinct measurement sources that may be perceived by the sensor. Early variants of such an approach have for instance been used for extended object tracking in [Dez98; SG99]. As illustrated in Fig. 3.6a, the reflection points could, for example, be the corners of an object which may be especially dominant features for a sensor. At each measurement cycle, the sensor measures one or multiple reflection points and the union of the resulting measurements constitutes the measurement set Z. The reflection points are usually distributed over the object extent and their position is typically given with respect to the object center of motion. Thus, they provide a description of the shape or structure of the object.



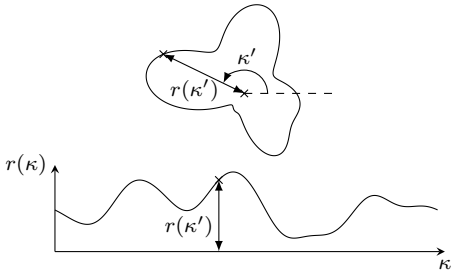
(a) Reflection point model: Example with four reflection points at the corners of an object giving rise to three measurements.



(b) Spatial distribution model: An exemplary stick-shaped object is convolved with Gaussian measurement noise.



(c) Random matrix model: Measurements are distributed according to a multivariate Gaussian which also describes the object extent.



(d) Star convex shapes: The contour is constructed using a radius function $r(\kappa)$.

Figure 3.6: Extended object measurement models

There are different possible variations of the reflection point approach. The reflection points may, for example, be fixed in both position and number if the appearance of the tracked object is well-known. Alternatively, their position and number could be made part of the object state and estimated concurrently. Also, one may restrict the number of measurements a sensor receives from a single reflection point to one or allow for multiple measurements.

To demonstrate the likelihood function for a reflection point model, the following explanations focus on the simple case of a set of c fixed reflection points that each give rise to at most one measurement. Particularly, a reflection point generates a measurement if and only if it is detected, which is the case with detection probability $p_D^{(i)}$. The reflection point's likelihood function $p^{(i)}(z|x)$ (cf. Fig. 3.6a) models where measurements are expected in case of detection. The resulting likelihood function

for the overall reflection point model is (cf. [GBR17])

$$p(Z|x) = \sum_{\theta \in \Theta} \prod_{\theta(i)=0} (1 - p_D^{(i)}) \prod_{\theta(i)>0} p_D^{(i)} p^{(i)}(z^{(\theta(i))}|x), \tag{3.31}$$

where the number of measurements is at most the number of reflection points. Also, θ is again an association mapping $\theta : \{1, \ldots, c\} \to \{0, \ldots, |X|\}$ which maps the reflection points to the measurements or the case of misdetection (0) under the restriction that each measurement may only be assigned to a single reflection point. Hence, different variants of measurement-to-reflection-point relationships are constructed. This is necessary since the origins of the measurements are unknown and the likelihood function has to take all possibilities of which reflection point gave rise to which measurement into account.

Depending on the number of reflection points and measurements as well as their positions, this association problem can be difficult to solve and there may be many possibilities. As the problem is identical to that in multi-object tracking, where measurements need to be associated to objects, techniques from multi-object tracking can be used for obtaining good solutions in the presence of ambiguity. Increasing the flexibility in the model, e.g. by allowing for multiple measurements from one reflection point or varying numbers and positions of reflection points, further complicates the association problem and estimation.

Spatial Distribution Models

Spatial distribution models as proposed in [GGMS05; GS05] directly define the distribution of the measurements around the object $g(z|x)$, i.e. the likelihood function. They assume that all measurements are distributed according to this likelihood function and are conditionally independent. Additionally, they model a distribution over the number of received measurements. In particular, the original definition used an inhomogeneous Poisson point process for this purpose. Yet, other cardinality distributions are possible and thus the overall likelihood can be written in the form of an independent identically distributed RFS (cf. (3.19))

$$p(Z|x) = |X|!\, p(|X|)[g(\cdot|x)]^Z, \tag{3.32}$$

which again uses the multi-object exponential notation from (3.20). The term spatial distribution indicates that the model directly specifies a distribution over the locations of the measurements. Note, however, that the principle is not limited to measurement locations and can be extended to other measurement quantities.

In [GS05], the authors suggest that the spatial distribution can be obtained by first specifying a distribution over possible measurement sources y and convolving it with a sensor error model $p(z|y)$ that captures sensor noise. This yields [GS05]

$$g(z|x) = \int p(z|y)p(y|x)\,\mathrm{d}y \tag{3.33}$$

and allows for splitting the modeling process into two basic effects. As an alternative to this procedure, the spatial distribution can also be modeled directly. Figure 3.6b shows an exemplary distribution of a stick-shaped object with Gaussian noise as presented in [GS05]. One of the major advantages of the spatial distribution models is the avoidance of explicit association hypotheses, which renders efficient particle filter implementations possible [GS05].

A special form of a spatial distribution that allows for an analytical solution has been presented in [Koc08]. It is referred to as random matrix model and assumes that objects have an elliptical shape. The measurement likelihood or spatial distribution is a Gaussian distribution

$$g(z|x, \underline{\Sigma}) = \mathcal{N}(z|x_{\mathrm{pos}}, \underline{\Sigma}), \tag{3.34}$$

where the mean is given by the object position x_{pos}. The spread of the measurements and thus the object size is governed by the covariance matrix $\underline{\Sigma}$. To estimate the object extent, the covariance matrix is assumed unknown and estimated in conjunction with the kinematic state. For this purpose, the combined state distribution is modeled as Gaussian-inverse-Wishart distribution which is a conjugate prior to the likelihood model from (3.34). Due to the Gaussian structure, the random matrix model is especially suited for objects that create unstructured measurement clusters. An illustration of the model is provided in Fig. 3.6c. Further details on the approach including modeling assumptions can be found in [Koc08] and an overview of different variants and enhancements is given in [GBR17].

Contour Models

A third variant of extended object models uses a parametric description of the object contour to describe the object shape. Two particular variants for star-convex shapes are the random hypersurface model (RHM) [BH14] and the Gaussian process (GP) model [WÖ15]. In a star-convex shape, there exists a center point inside the object from which all points on the contour are visible, i.e. there exists a line connecting the center point and the contour point which lies entirely inside the object. Therefore, the contour can be described by a radius function $r(\kappa)$ that defines the radius over the angle κ of the connecting line. See Fig. 3.6d for an illustration.

RHM models assume that the measurements are generated on a scaled contour of the object and the measurement equation is [GBR17]

$$z = s \cdot r(\kappa, p) + x_{\text{pos}} + \epsilon. \tag{3.35}$$

Here, s is the random and unknown scaling factor, p are the parameters of the radius function, x_{pos} is again the position of the object center, and ϵ is additive noise. The use of the scaling factor allows measurements to be generated inside the object if necessary. While the initial proposal [BH09] demonstrated the approach for elliptically shaped targets, [BH11] proposed to use a Fourier series expansion of the radius function

$$r(\kappa, p) = \frac{a_0}{2} + \sum_{i=1}^{n_f} a_i \cos(i\kappa) + b_i \sin(i\kappa). \tag{3.36}$$

It uses n_f terms, is defined over the angle κ, and assumes a periodicity of 2π. The Fourier coefficients a_0, a_i, and b_i, which define the shape of the radius function and hence the contour, are combined in the parameter vector

$$p = [a_0, a_1, b_1, \ldots, a_{n_f}, b_{n_f}]^{\mathrm{T}}. \tag{3.37}$$

For tracking and shape estimation, the parameter vector is appended to the object state and concurrently updated in a nonlinear Gaussian filter. Thus, the radius function is adapted to the measurements while tracking.

The GP model from [WÖ15] represents the radius function using a probabilistic description in terms of a GP [RW06, p. 13]. The values of the radius functions at specific angles are assumed to follow a Gaussian distribution

$$r(\varphi) \sim \mathcal{N}(r(\kappa) | \gamma(\kappa), \underline{\Sigma}(\kappa, \kappa')) \tag{3.38}$$

with a mean vector $\gamma(\kappa)$ that depends on the angles κ and the covariance matrix

$$\underline{\Sigma}(\kappa, \kappa') = \begin{bmatrix} k(\kappa_1, \kappa_1) & \ldots & k(\kappa_1, \kappa_{n_\kappa}) \\ \vdots & \ddots & \vdots \\ k(\kappa_{n_\kappa}, \kappa_1) & \ldots & k(\kappa_{n_\kappa}, \kappa_{n_\kappa}) \end{bmatrix} \tag{3.39}$$

which is constructed using a periodical kernel $k(\cdot, \cdot)$ (see e.g. [RW06, p. 92]). This kernel governs the shape of the resulting function and how the function values influence each other. GPs allow computing a predictive distribution for function values at specific angles by learning from previously observed function and angles pairs. This is achieved by conditioning the joint Gaussian distribution over observed and unknown function values on the observed values. As GPs are originally a batch method, a recursive formulation with fixed basis angles is used in [WÖ15]. The

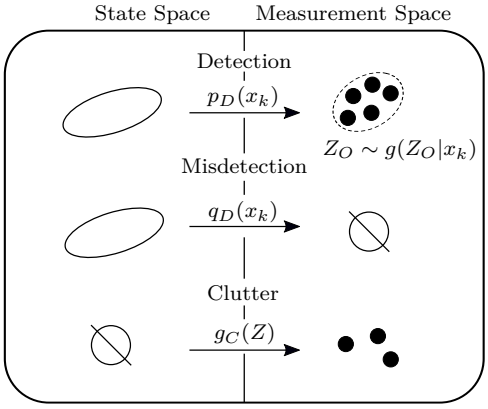


Figure 3.7: Three cases handled by the generalization of standard multi-object model to the extended object case: detection, misdetection, and clutter

corresponding function values and the object state can then be estimated concurrently and the predictive density is used as measurement likelihood function $g(z|x)$.

3.4.2 Filters for Extended Object Tracking

Several filter variants based on FISST have been proposed to accommodate such measurement models for single extended objects in a multi-object tracking framework. Most of these approaches are based on a generalization of the standard multi-object likelihood from Section 3.3.4; see [Mah14, pp. 115-116]. As illustrated in Fig. 3.7, it replaces Assumption 2 by allowing objects to give rise to a set of measurements Z_O if detected. These measurements follow the single-object likelihood function $g(Z_O|x)$ which incorporates the suitable extended object model, e.g. (3.31) or (3.32).

An elaborate overview of different adaptions of the multi-object filters from Section 3.3.4 to the extended object case is given in [GBR17]. Examples are the extended object versions of the PHD [Mah09] and CPHD [LGO13], the GLMB and LMB [BRG$^+$15; BRG$^+$16] filters, and those of the PMBM filter variants in [GFS16; XGSF18]. While these filters differ form their point object variants in the measurement model, their respective characteristics in terms of assumptions, advantages, or disadvantages remain the same.

3.5 Sensor Fusion Approaches

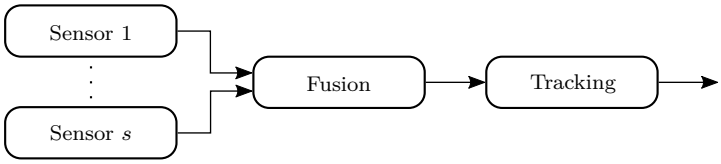
Tracking filters are one method to fuse data from multiple sensors and to achieve the benefits that were listed in the introduction. This section briefly discusses three common fusion architecture variants following the comprehensive overview given in [BTW11, Chapter 8]. In contrast to [BTW11], however, where the procedure is split into first associating and then fusing the data, this section uses the word fusion to refer to the entire procedure in a more general sense.

The timely order of measurements is a crucial factor in sensor fusion as sensors may be asynchronous and may exhibit different latencies. If the latencies differ considerably, measurements from sensors with high latency may arrive after newer measurements from fast sensors have arrived, which leads to out-of-sequence problems. While there has been some work on solving such issues, see e.g. [Bar02], this thesis assumes that sensor measurements arrive in correct order and out-of-sequence problems are therefore not further considered.

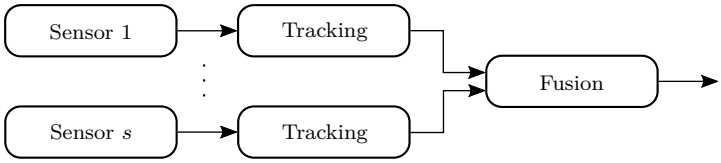
In static fusion, a composite measurement is formed by combining measurements from multiple sensors into one. This composite measurement can then be used in a tracking filter, as shown in Fig. 3.8a. However, static fusion requires synchronized sensors. Also, it becomes especially susceptible to sensor failure if it uses combination algorithms that require the availability of data from both sensors at all times.

Track-to-track fusion (see Fig. 3.8b) is another architecture variant in which each of the sensor modules possesses its own full tracking pipeline and outputs object tracks. These tracks are then combined in a fusion center by fusing the track estimates. There are different variants of track-to-track fusion which differ in the communication between the fusion center and the sensor modules as well as in the way tracks are processed over time in the fusion center. That is, the fusion center may or may not provide feedback to the sensor modules and it may or may not keep tracks in memory. The core issue to be considered in designing track-to-track fusion architectures is to avoid overconfidence in the fused estimates. This can be caused by ignoring correlation in the track estimates of the sensor modules due to common process noise or by reusing information from measurements in architectures with memory. Different approaches tackling these issues are presented in [BTW11, Chapter 9]. In the special case of linear systems, optimal results can be achieved by using information matrix fusion [BTW11, pp. 559-566].

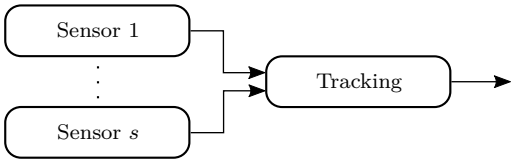
In centralized fusion, see Fig. 3.8c, measurements from all sensors are fed to a central tracking filter which processes the measurements iteratively. This variant achieves optimal performance as it uses all available information [BTW11, p. 557]. The issue of correlated data is avoided even in the presence of complicated nonlinear



(a) Static fusion



(b) Track-to-track fusion



(c) Centralized fusion

Figure 3.8: Illustration of sensor fusion architectures

systems. Moreover, it does not require synchronized sensors as in static fusion since measurements can be easily processed at the time of arrival. One of the major drawbacks of centralized fusion is that all measurements need to be transmitted to the fusion center which typically requires more data to be transfered than for object lists and may be an issue in presence of limited communication bandwidth.

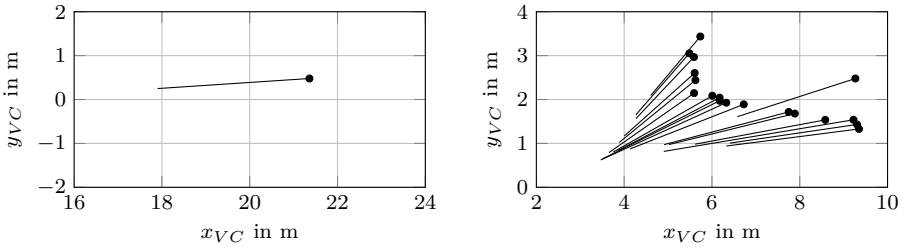
Chapter 4

Framework for Tracking Multiple Vehicles

After having laid the foundations of Bayesian inference and tracking algorithms, this chapter turns the attention to the application of vehicle tracking using different homogeneous sensors. The chapter starts in Section 4.1 by reviewing the current state of the art in vehicle tracking. It highlights and discusses different solutions in terms of data processing, multi-object tracking, as well as sensor fusion to provide a more detailed motivation and differentiation of the work in this thesis to existing methods. Subsequently, the chapter introduces the proposed tracking framework that tackles the three challenges in a fully Bayesian fashion while using all available information and providing the ability to easily add and remove sensors. First, the underlying Bayesian problem formulation is presented in Section 4.2 and Section 4.3 then details the system architecture as well as the different steps in the filter cycle.

4.1 State of the Art in Vehicle Tracking

Due to the great interest in developing driver assistance systems and automated vehicles in the past years, literature on vehicle tracking is vast and a complete overview of all proposed approaches for the different sensor types is beyond the scope of this thesis. Nonetheless, this section provides a concise overview of different ways that have been chosen to deal with the extended object and multi-object problems as well as sensor fusion.



(a) Single measurement of a vehicle in the far range

(b) Multiple measurements of a vehicle in the close range

Figure 4.1: Exemplary radar measurements of a vehicle plotted in vehicle coordinates (cf. Section 4.2). The Doppler velocities are indicated by the vectors.

4.1.1 Radar Data Processing

Radar sensors emit and receive electromagnetic waves to perceive their environment. Particularly their ability to directly measure the relative radial velocity (Doppler velocity) is one of their most distinctive features in comparison to other automotive sensors. Processing the reflected electromagnetic waves constitutes a whole discipline of research itself, see e.g. [Sko02; Win16]. In this thesis, it is assumed that radar sensors directly provide measurements on detection level. That is, they yield a set of measurement points with corresponding range, azimuth angle, and Doppler velocity values. Figure 4.1 shows two examples of radar measurements that originate from a vehicle and have been recorded using a sensor that is later used in the experimental evaluation.

Processing radar data for vehicle tracking is challenging for several reasons. First, radar measurements oftentimes do not exhibit a clear geometric structure as it is the case for lidar measurements. Secondly, the number of measurements that are received from an object hinges on the sensor's capability to resolve different reflections. This can depend on various factors including the sensor-to-object constellation as well as the vehicle motion. In case of the sensor that was used in this thesis, for example, the amount of measurements varies from a complete misdetection over a single measurement in the mid and far range (cf. Fig. 4.1a) to over twenty measurements in the close range (cf. Fig. 4.1b). Thirdly, the Doppler velocity leaves considerable ambiguity in the object motion as it only measures the radial component of the object velocity vector.

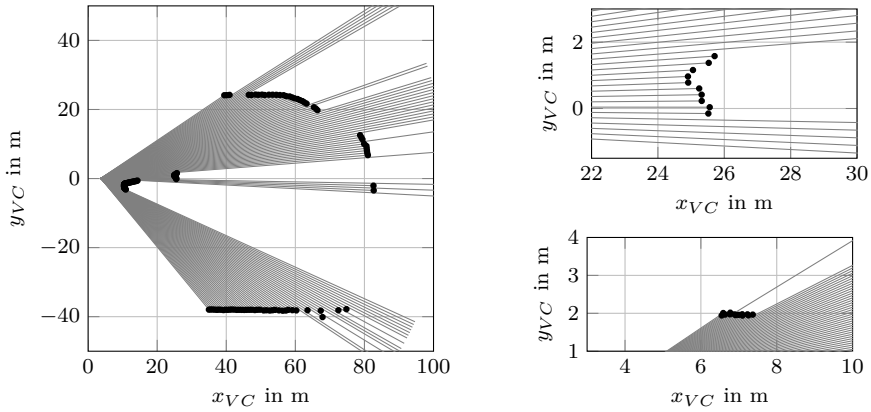
Researchers have proposed several approaches that first process such radar data

before using the resulting meta-measurements in tracking. Clustering methods have, for example, been presented in [KKD12; SRKW16] and [DAK+15; EAK16] discuss tracking approaches that are based on clustered data. Also, box fitting methods for radar data have been proposed in [RKDW16; RKK+15; SSSW17]. A third variant of preprocessing routines aims at exploiting the Doppler information and extracts motion estimates from a single frame by analyzing the Doppler profile of the received measurements. When using a single sensor only, this method allows to extract the vehicle speed [FR06; KBD+13; RX17]. If two synchronized radar sensors are available, an additional estimate of the yaw rate can be computed [KBK+14]. The application of such approaches in a tracking framework is discussed in [KBK+16].

Such preprocessing routines are particularly successful if there is a sufficient amount of measurements for a single vehicle, as for instance in the example of Fig. 4.1b. By simply computing the centroid of a measurement cluster, however, one is subject to migration of the resulting position due to varying aspect angles and sensor noise. Also, box fitting and Doppler profile analysis fail if the amount of measurements is low, as for example in the extreme case of Fig. 4.1a.

Apart from the preprocessing routines, a variety of extended object measurement models for radar-based tracking have been proposed. In [CSKR14; DRBU09; KYY+15], the authors assume that vehicles can be described as rectangles and that measurements are generated at their corners or edge centers. This can be interpreted as a particular form of a reflection point model. The publications [GSDB07; HSSS12] employ more detailed reflection point models that are constructed from typical radar reflection characteristics and consider visibility regions for each reflection point. Reflection point models that allow for estimating the number and position of the reflection centers have been presented in [HLS12; SRW14]. Other extended object approaches that use an elliptically shaped spatial distribution are the volcanormal density [BDD17] and a random matrix model that has been extended to account for Doppler measurements [SRW15]. However, the latter approach was only applied to vessel tracking and does not consider the variation of the Doppler measurements over the tracked object. Furthermore, a stick model that describes the rear surface of a vehicle and is intersected with radar beams for tracking preceding vehicles [ASW13] and the application of Gaussian processs (GPs) without considering Doppler measurements [MBMW17] have been proposed.

In contrast to the preprocessing routines, these extended object models do not require a particular number of radar measurements. Yet, they also exhibit some disadvantages: A drawback of reflection point models is, for instance, that finding suitable sets of reflection points becomes more complex as the sensor resolution increases and the sensors are able to resolve more and more scatter centers (cf. Fig. 4.1b). Also, the other methods either use a rather coarse or partial description of the vehicle shape or do not fully exploit the available Doppler information.



(a) 2D lidar scan

(b) Measurements of a vehicle in front of the lidar (top) and a vehicle that is leaving the FOV

Figure 4.2: Exemplary lidar measurements plotted in vehicle coordinates (cf. Section 4.2). The lidar rays are indicated as gray lines.

4.1.2 Lidar Data Processing

Automotive lidar sensors scan the environment by emitting laser impulses from one or more vertical layers and rotating the rays over the field of view (FOV). If a ray is reflected by an object, the distance is measured using the time of flight principle [GG16]. Thus, they provide dense point clouds with precise range and angular measurements that oftentimes exhibit the geometrical structure of observed objects. Due to the focused laser rays, lidar measurements are subject to occlusion effects as known from camera images. Also, the object material strongly influences the return signal strength and absorption, e.g. for black vehicles, may occur. In principle, a Doppler shift is observable for objects that move relative to the sensor. Due to the high frequency of the involved electromagnetic waves, however, measuring these shifts is costly and is omitted in automotive sensors.

Lidar-based tracking can be categorized into 2D and 3D approaches depending on the type of lidar sensor that is used. The work on lidar tracking in this thesis is based on a sensor that mostly provides 2D information and therefore the thesis supposes the availability of planar lidar scans as illustrated in Fig. 4.2a. Accordingly, the following literature review focuses on 2D approaches and only mentions some work from the large body of 3D lidar processing.

Like in radar-based methods, preprocessing routines play an important role in lidar-based tracking. A majority of approaches, see e.g. [BRAD16; CSKR14; DRBU09; KBD05; MDM09; MNM+13], fits bounding boxes or L- and I-shapes to the data. Alternatively, [MCH+08] extracts the closest distance and the outermost angles of point cloud clusters as meta-measurements. A routine for extracting a detailed surface from 3D lidar measurements was presented in [SKD+11] and approaches that extract polyline segments from lidar data to update a polyline vehicle model have been presented in [Eff09; RBL+08]. Recently, the application of convolutional neural networks to 3D lidar data for bounding box regression as in [LZX16] has also become popular.

Again, the lidar preprocessing routines, particularly the box fitting approaches, face difficulties if the correct meta-measurements cannot be easily extracted. The measurements may, for example, be corrupted by noise and may not clearly exhibit the shape; see the upper example of Fig. 4.2b. Here, the measurements appear to follow an L-shape that is rotated by approximately 45° whereas the actual vehicle was oriented in parallel to the lidar rays. Interestingly, the correct orientation becomes apparent to the observer when examining the adjacent rays that indicate free space to both sides of the vehicle. Yet, most pre-processing routines do not make use of this negative information, i.e. of the absence of measurements, and only work on point clouds of positive returns. Box fitting also becomes difficult when objects enter or leave the sensor FOV and are only partially visible as in the lower example of Fig. 4.2b. Heuristic rules are oftentimes necessary to cope with these special cases.

Therefore, several extended object measurement models that process the lidar measurements directly have been proposed. For example, [GLO11] presents a method for computing multiple predicted measurement locations for rectangular and elliptical object shapes. This approach is extended using two measurement modes in [GRMS14] and motivated as an approximation of a spatial distribution approach. Moreover, a polyline object model that processes all measurements is discussed in [WC17], and GPs have been applied to vehicle tracking in [HSRD16; MBMW17; WÖ15]. Additionally, local grid maps, i.e. small grid maps that are attached to the tracked object and allow for free-form descriptions, have been applied in [SADD13; SADD14b] for the 2D case and in [SKD+12] for the 3D case. There has also been work on ray tracing approaches in [PT09; VA09; VWN12] which closely emulate the sensor principle and directly compare the expected and measured distance readings using geometric object models. Lastly, some approaches [HLT13; KBS17] that are based in optimization as opposed to Bayesian inference accumulate a point cloud over time as shape representation and obtain tracking results from matching the measured point cloud to the current estimate.

The free-form models such as local grid maps, GPs, or polylines provide the greatest flexibility in the object shape. This flexibility is advantageous for tracking objects

of considerably different shape but it may complicate estimation if objects exhibit a similar and simple shape. The rectangular models from [GLO11; GRMS14] simplify the shape but use an approximate prediction of the measurement locations, require considering different cases for the number of visible sides, and use an internal association step. The ray-based methods, in contrast, emulate the sensor measurement principle the closest. Despite being computationally more demanding, they allow for accurate predictions of the number and location of measurements and enable a coherent incorporation of negative information.

4.1.3 Camera Data Processing

Object tracking using camera images constitutes an entire discipline in the field of computer vision with an enormous variety of applications. For vehicle environment perception, information from mono cameras is mostly used in form of object detections, e.g. bounding boxes. The detections are then used as measurements in tracking filters and are oftentimes fused with data from other sensors; see for instance [CSKR14; KBD05; KNW+15; MDM09]. A recent approach that is solely based on detections with additional distance information is presented in [SBR+18]. The detections are obtained from applying a convolutional neural network (CNN) to mono camera images.

Besides object detection in images, scene labeling is another task from computer vision in which the goal is to assign semantic labels to each pixel in a mono camera image. That is, each pixel is labeled with the class of the object that it displays. Figure 4.3 shows an example using two classes. Especially the application of CNNs, see e.g. [FCNL13; LSD15], has greatly contributed to the performance and popularity of scene labeling. In contrast to bounding box detections, scene labeling provides more dense semantic information which allows inferring the contour of labeled patches. Yet, it does not provide instance-level information and several objects may be merged to a large patch of identically labeled pixels. To the best of the author's knowledge, there are no published attempts to use scene labeling information in a classical tracking approach where it is fused with information from other sensors such as radar.

4.1.4 Tracking Filters

In terms of the multi-object problem, a large body of the literature assumes that vehicles are independent of each other and that there is no considerable ambiguity in the measurement association. Therefore, they employ a bank of single-object

Figure 4.3: Scene labeling example: The image is labeled using the classes vehicle (red) and background (blue).

trackers such as Kalman filters, extended Kalman filters (EKFs), unscented Kalman filters (UKFs), or particle filters in combination with different association schemes. Examples are given in [CSKR14; DRBU09; Eff09; MNM+13; RMM15; SWBH12; VDN15].

Other publications have investigated the application of multi-object filters to vehicle tracking. For example, [MDM09; MMD10] proposed the use of the joint integrated probabilistic data association (JIPDA) filter. Also, several multi-object filters based on random finite sets have been applied. Gaussian mixture implementations of the probability hypothesis density (PHD) filter were used in [GLO11; GRMS14; MBMW17] in conjunction with different extended object measurement models. The particle implementation of the PHD filter was used in [SADD14a] to track multiple vehicles using local grid maps. The application of the labeled multi-Bernoulli (LMB) filter to preprocessed measurements from lidar, camera, or radar data has been investigated in [KNW+15; Reu14] and the extended object variant of the LMB filter in combination with a GP model was used in [HSRD16]. Also, an approach based on the Poisson multi-Bernoulli mixture (PMBM) distribution using camera detections was presented in [SBR+18].

4.1.5 Sensor Fusion

To fuse data from different sensor types, some publications propose static fusion methods and combine the sensor data before filtering. Examples are [HLT13], which combines color information from images with 3D point clouds, [KBK+14; KBK+16] which combine data from two radar sensors to obtain an instantaneous estimate of

speed and yaw rate, and [BKBD15] which feeds prior information from lidar sensors into a radar preprocessing routine.

Examples for track-to-track fusion were presented in [ASKB12] using a theoretically rigorous information fusion approach or in [DGV+16] with special focus on fusing shape estimates from heterogeneous sensor types.

A majority of vehicle tracking approaches uses a centralized fusion architecture where measurements from different sensors are processed in a single filter. Preprocessed measurements are, for instance, fused in a centralized fashion in [BRAD16; CSKR14; DRBU09; EAK16; KBD05; KNW+15; MDM09; MMD10; Reu14; SWBH12]. In contrast, centralized fusion using extended object measurement models, as presented in [MBMW17; SRW14], is rather rare.

4.1.6 Discussion

The survey of the state of the art confirms the claims that have been made in the introduction. It is observable that many of the vehicle tracking frameworks resort to a classical structure with separate preprocessing and association modules in combination with independent tracking filters for the individual objects. Nonetheless, there has been work to solve the three aspects of data processing, the multi-object problem, and sensor fusion in a principled way and without resorting to intermediate simplifications. Yet, most of the approaches focus on parts of the problem, e.g. by centering on the multi-object tracking problem alone or providing methods for tracking multiple extended objects using a single sensor type. In cases where data from multiple sensors is fused using multi-object filters, it is often the case that the likelihood functions of different sensors share identical building blocks. For example, they use the same extended object model as in [MBMW17] or the same multi-object likelihood as in [KNW+15; MDM09; MMD10; Reu14]. However, these models may not always be the most appropriate ones for all sensor types.

Moreover, there is a variety of extended object models for tracking with radar or lidar sensors. Their different strengths and weaknesses have their origin in the conflict of interest between using simple models with restrictive assumptions, elaborate models with high detail, or models that provide great flexibility. The random matrix model, which exploits the availability of conjugate priors, for example, does not play an important role in vehicle tracking as the assumptions that enable conjugacy are too simplistic. GP models, on the other hand, are also computationally efficient and do provide great flexibility in the shape. Yet, they fail at accurately modeling the lidar measurement principle as they do allow for measurements on the averted surface or do not use the negative information that is provided. In contrast, full-fledged

reflection point models for radar sensors use a detailed model of the radar reflection characteristics but require expert knowledge in creating these models including the visibility regions of each reflection point.

As stated in the introduction, the goal of this thesis is to develop a tracking framework that provides an integrated probabilistic approach to the extended object problem, the multi-object problem, and sensor fusion, which works in ambiguous situations and uses all available information from the sensors. Consequently, the goal is to firstly propose new extended object models that are specifically tailored to vehicle tracking with a particular sensor type and find a good trade off between mathematical rigor, analytical simplicity, and accurateness in the measurement process. Thus, this work strives for achieving accurate tracking results and natural handling of typical sensor characteristics such as ambiguity in the data or partly visible objects. Secondly, the thesis aims at developing a framework that provides the necessary theoretical foundation without imposing overly restrictive limitations on sensor-specific measurement models and update routines.

4.2 Problem Formulation

After having reviewed and discussed existing work on vehicle tracking, this section begins to introduce the tracking framework that is put forward in this thesis by providing the basic problem formulation. Its task is to estimate the number of vehicles in an ego-vehicle's surrounding as well as their states based on data obtained from the ego-vehicle's on-board sensors. For this purpose, a finite set statistics (FISST)-based formulation is chosen. The used random finite sets (RFSs) allow for varying numbers of objects or measurements and the multi-object Bayes filter from Section 3.3.1 provides the means for a probabilistic end-to-end formulation which embraces the entire problem from raw measurements to the state estimates. Due to the explicit consideration of multiple measurements, different extended object measurement models can be incorporated naturally.

The multi-object state $\boldsymbol{X}_k = \{\boldsymbol{x}_k^{(1)}, \dots, \boldsymbol{x}_k^{(n)}\}$ is modeled as a labeled RFS where each element is the labeled state vector $\boldsymbol{x}_k = [x_k^{\mathrm{T}}, \ell]^{\mathrm{T}}$ of a vehicle and the labels ℓ indicate the respective vehicle's identity. Each vehicle's state $x_k = [\xi_k^{\mathrm{T}}, \zeta_k^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{X}$ is composed of two portions. The kinematic state $\xi_k = [x_{R,k}, y_{R,k}, \varphi_k, v_k, \omega_k]^{\mathrm{T}}$ describes the vehicle pose and motion at time step k and consists of the position of the rear axle center $[x_{R,k}, y_{R,k}]^{\mathrm{T}}$, the vehicle orientation φ_k, the speed v_k in direction of the vehicle orientation, and the yaw rate ω_k. As illustrated in Fig. 4.4, all quantities are estimated with respect to the ego-vehicle's coordinate system which is indicated by the subscript VC. The second part of the state vector constitutes

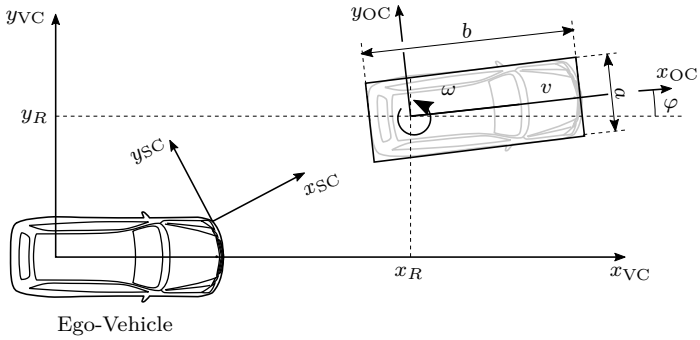


Figure 4.4: Schematic illustration of the vehicle state vector

the extent portion $\zeta_k = [a_k, b_k]^{\mathrm{T}}$ which consists of the vehicle width a_k and length b_k. The position of the rear axle is assumed to be fixed at 77% of the vehicle length, which has empirically shown to be a suitable value for many vehicle types[1].

The perception sensors are mounted on the ego-vehicle and provide measurements in the sensor coordinate (SC) systems. From sensor calibration, the transformation between sensor and vehicle coordinates is known and objects can be transformed from one coordinate system to another. The mathematical representation of the measurements depends on the sensor characteristics. If the number of measurements is able to change from cycle to cycle, they are represented using a measurement set Z_k, otherwise all measurements are combined in a fixed-size vector z_k. Further details are presented in the measurement model sections of Chapter 5.

Apart from the vehicle and sensor coordinate systems, this thesis uses the object coordinate system as a third frame if quantities are best described with respect to the object under consideration. It is denoted by the subscript OC and attached to the tracked vehicles in the same way, the vehicle coordinate system is attached to the ego-vehicle. To simplify notation, this thesis avoids the explicit declaration of transformations between the coordinate systems. Instead the convention is that all quantities are given with respect to the vehicle coordinate system if there is no involvement of sensors and their measurements. Accordingly, the quantities are defined in the sensor coordinate system in the opposite case. For example, the object position and orientation are automatically assumed to be given with respect to the sensor coordinate system when used in the measurement model formulation. The use of the object coordinate system is explicitly specified by the appropriate subscript where applicable.

[1] The value was chosen after examining the rear axle position from the technical documentation of different vehicles, including sedans, a compact car, and a van.

4.3 System Architecture and Filter Cycle

To solve the multi-object Bayes filter, the multi-object distributions are modeled as LMB and generalized labeled multi-Bernoulli (GLMB) distributions as in the LMB filter [RVVD14] and the extended object LMB filter from [BRG$^+$16]. The framework uses a structure that is similar to the LMB filter and which outputs a posterior LMB distribtion over the multi-object state. Therefore, each vehicle is described by its state distribution $p^{(\ell)}(x)$ and its probability of existence $r^{(\ell)}$. Due to dependencies that are introduced during prediction and update, GLMB distributions become necessary in intermediate steps.

A labeled RFS approach is chosen since different studies have indicated that such methods are able to outperform other filters such as the cardinalized probability hypothesis density (CPHD) [BRG$^+$16] or cardinality-balanced multi-target multi-Bernoulli (CBMeMBer) [RVVD14] filters. In comparison to Poisson multi-Bernoulli mixture (PMBM) methods, they also provide a rigorous notion of object identity.

The data from multiple sensors is fused in a centralized fusion architecture. Thus, the sensors are not required to be synchronized and all information from the data is directly available at the fusion center. Also, there is no dependence on the availability of other sensors as in static fusion. Figure 4.5 shows a schematic illustration of the overall system architecture and the filter cycle. Every time new measurements arrive from a sensor, a new filter cycle is triggered. It is assumed that measurements arrive in correct order and out-of-sequence problems are not considered. In set-ups where latencies differ considerably among the sensor types, adequate buffering strategies can be used to achieve a correct order of the arriving measurements.

Both the prediction step, which models the evolution of the multi-object state over time, and track management are independent of the current sensor type. In contrast, the update routine for processing the measurements and initialization procedure for generating new objects differ for each sensor. These operations are therefore capsuled in sensor modules which expect a prior multi-object distribution in GLMB form and output a posterior LMB distribution as well as an LMB birth density with new object hypotheses. Each sensor module is parameterized for a particular sensor, i.e. it contains all necessary information such as detection probabilities and mounting position or orientation. If a particular sensor type is used multiple times, each sensor uses its own instance of the corresponding sensor module.

Note that the sensor module architecture differs from previous approaches such as [KNW$^+$15; MDM09; MMD10] which were also aiming at a generic fusion architecture. They achieve clear sensor interfaces by using identical update procedures and specifying measurement models with common parameters such as the detection probability.

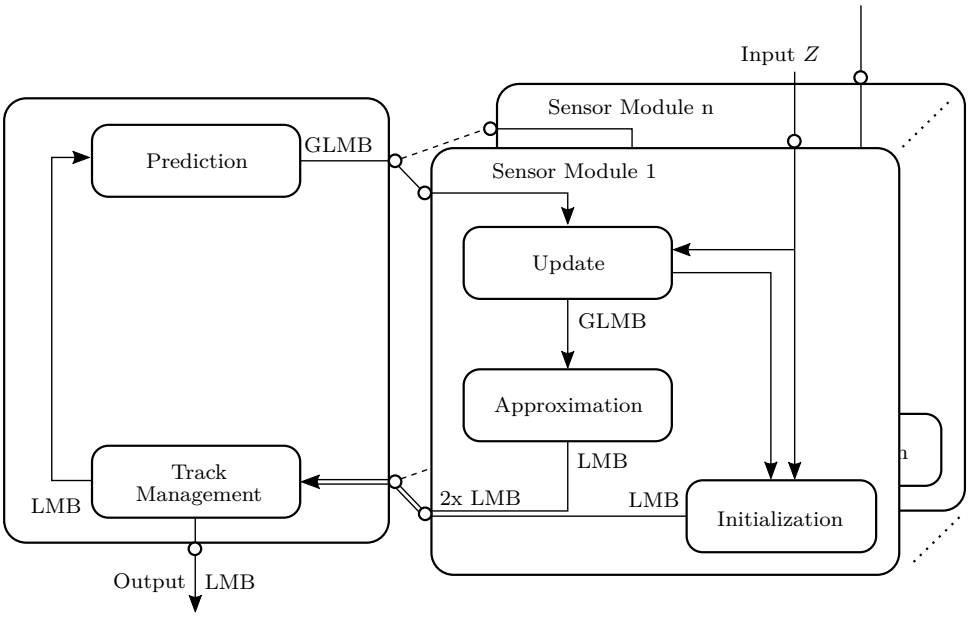


Figure 4.5: Schematic of the system architecture

Here, the interface is instead defined in terms of multi-object distributions that are transmitted from and to the sensor modules. This still achieves an architecture with clear and generic interfaces that allow for easy adding or removing of sensors. At the same time, all operations that involve sensor measurements and knowledge about the sensor characteristics are clearly capsuled in sensor modules. Thus, the architecture avoids heavy restrictions on the update, and measurement models that best suit the corresponding sensor data can be included. The framework is therefore not limited to using the generalized standard multi-object measurement model that is for example employed in the extended object LMB filter.

In terms of hardware implementation, the architecture allows for two different options. Sensor modules could be either implemented in the fusion center which would require a transmission of the entire measurement data to the fusion center or the modules could be implemented on sensor processors which would require transmitting the predicted and posterior multi-object densities between fusion center and sensors.

The remainder of this chapter provides further details on the involved processing steps. All explanations focus on a single filter cycle. To shorten notation, the time indices k are dropped in the following and prior quantities are distinguished by the subscript $+$.

4.3.1 Prediction

In the standard multi-object transition model from Section 3.3.4, all objects are predicted independently. Thus, it may happen that two vehicles are predicted to nearby positions which would—under consideration of the vehicle extent—lead to physically impossible overlaps. To avoid such situations, [RWD12; Reu14, Section 4.2] have proposed adaptions of the prediction step for sequential Monte Carlo (SMC) implementations of the multi-object Bayes filter. They include a second prediction step which validates predicted multi-object state and adjust it by discarding particles with impossible object constellations or by adapting their weights. This thesis adopts the introduction of a second prediction step and transfers the idea to filters using LMB distributions. The resulting two-stage prediction routine ensures physical consistency among extended objects and is presented in the following.

Step 1: Standard Multi-Object Transition Model

Before prediction, the posterior distribution of the multi-object state is in LMB from and hence follows (3.25) and (3.26). The label space $\mathbb{L}_+$ is made up of the current vehicle hypotheses' labels. The first prediction step uses the standard multi-object transition model from Section 3.3.4 and considers cases 1 and 2, i.e. object survival with corresponding state transition or object disappearance. The third case, object appearance, is delegated to the sensor modules and track management as it involves sensor measurements and knowledge about the sensor characteristics; see the following subsections. As in the LMB and extended object LMB filters [BRG$^+$16; RVVD14], the parameters of the resulting prior LMB distribution are

$$r_+^{(\ell)} = \eta(\ell) r^{(\ell)}, \tag{4.1}$$

$$p_+(x_+, \ell) = \frac{\int p_S(x, \ell) f_+(x_+|x, \ell) p(x, \ell) dx}{\eta(\ell)}, \tag{4.2}$$

$$\eta(\ell) = \iint p_S(x, \ell) f_+(x_+|x, \ell) p(x, \ell) dx dx_+. \tag{4.3}$$

Equation (4.2) mostly corresponds to the standard prediction equation of the Bayes filter (3.3) and computes the prior state density $p_+(x_+, \ell)$ under consideration of the persistence probability. In contrast, (4.1) computes the prior existence probability by discounting the previous value with the normalization factor (4.3), which is the expected value of the probability of persistence[2] $p_S(x, \ell)$. Therefore, the decrease in the existence probability depends on the persistence probability values in the

[2]Recall that $p_S(x, \ell)$ is a probability, which is a function of the labeled state, and not a density (cf. Section 3.3.4).

areas where the object is located. If the persistence probability is low, the existence probability decreases accordingly.

The evolution of the kinematic states is modeled by a constant turn rate and velocity (CTRV) model [Ger89] (see [SRW08] for a comparison of models) which assumes curvilinear motion of the vehicle with constant speed and a constant yaw rate. Its deterministic part is superimposed with process noise for all kinematic states. This yields

$$\begin{bmatrix} x_{R,+} \\ y_{R,+} \\ \varphi_+ \\ v_+ \\ \omega_+ \end{bmatrix} = \begin{bmatrix} x_R + \frac{v_+}{\omega_+}\left(\sin(\omega_+ \cdot \Delta t + \varphi) - \sin(\varphi)\right) + \epsilon_x \\ y_R + \frac{v_+}{\omega_+}\left(-\cos(\omega_+ \cdot \Delta t + \varphi) + \cos(\varphi)\right) + \epsilon_y \\ \varphi + \omega_+ \cdot \Delta t + \epsilon_\varphi \\ v + \epsilon_v \\ \omega + \epsilon_\omega \end{bmatrix}, \tag{4.4}$$

where Δt is the time difference from the last to the current measurement, and ϵ_x, ϵ_y, ϵ_φ, ϵ_v, as well as ϵ_ω are the process noise terms. See Section 6.3.3 for more implementation details and information on the chosen noise.

If the probability of persistence $p_S(x, \ell)$ is chosen independently of the update time intervals, the decay rate of a vehicle's existence probability strongly depends on the frequency of newly arriving measurements. This frequency is mainly governed by the number and type of sensors that are used in the current fusion constellation. In the proposed framework, the probability of persistence is modeled such that adding and removing sensors does not affect the intended behavior. In a first step, the FOVs of all registered sensors are combined and it is tested if vehicles are located inside or outside of the overall FOV. The probability of persistence is determined by assuming exponentially distributed events of object disappearance and computing the probability that a vehicle has disappeared within the last cycle interval. By choosing large expected time intervals between object disappearance in the FOV and short intervals outside the FOV, it is ensured that vehicles which are not visible to the sensors are discarded quickly whereas vehicles inside the sensor FOVs are maintained for longer periods of time.

Step 2: Accounting for the Extent

As previously mentioned, the goal of the second prediction step is to achieve physical consistency without overlaps. For this purpose, the second prediction step conditions the predicted multi-object state on the event of being physically consistent, which is denoted by $\mathcal{F}$. This method has been developed as part of this thesis and has been presented in the prior publication [SRD16]. The conditional prior multi-object

distribution

$$\boldsymbol{\pi}_+(\boldsymbol{X}_+|\mathcal{F}) = \frac{g(\mathcal{F}|\boldsymbol{X}_+)\boldsymbol{\pi}_+(\boldsymbol{X}_+)}{\int g(\mathcal{F}|\boldsymbol{X}_+)\boldsymbol{\pi}_+(\boldsymbol{X}_+)\delta\boldsymbol{X}_+}. \tag{4.5}$$

is obtained by applying Bayes' theorem and using a likelihood for a physically consistent state in the form of

$$g(\mathcal{F}|\boldsymbol{X}_+) = g(\mathcal{F}|\mathcal{L}(\boldsymbol{X}_+)) = \prod_{\{(\ell_i,\ell_j)\in\mathcal{L}(\boldsymbol{X}_+)|\ell_i\neq\ell_j\}} \left(1 - p_O(\ell_i,\ell_j)\right). \tag{4.6}$$

It evaluates the overlap probability $p_O(\ell_i,\ell_j)$ of all label pairs in $\boldsymbol{X}_+$ and assigns high likelihood values to multi-object states where the overlap probabilities are small for all pairs. As summarized in the following proposition, the resulting predicted multi-object density is in GLMB form.

Proposition 1. *Let the current distribution of the predicted multi-object state $\boldsymbol{X}_+$ be an LMB distribution with label space $\mathbb{L}_+$ and parameters from (4.1) and (4.2). The predicted multi-object state conditioned on the event of being physically consistent $\mathcal{F}$ using the consistency model from (4.6) then follows a GLMB distribution of the form*

$$\boldsymbol{\pi}_+(\boldsymbol{X}_+|\mathcal{F}) = \Delta(\boldsymbol{X}_+)w_+(\mathcal{L}(\boldsymbol{X}_+))[p_+(\cdot)]^{\boldsymbol{X}_+}, \tag{4.7}$$

with

$$w_+(I) = \frac{g(\mathcal{F}|I)\tilde{w}_+(I)}{\sum_{J\in\mathbb{L}_+} g(\mathcal{F}|J)\tilde{w}_+(J)}, \tag{4.8}$$

and where $\tilde{w}_+(L)$ is obtained by inserting the existence probabilities (4.1) into (3.26).

Proof. The numerator of (4.5) is

$$g(\mathcal{F}|\boldsymbol{X}_+)\boldsymbol{\pi}_+(\boldsymbol{X}_+) = g(\mathcal{F}|\mathcal{L}(\boldsymbol{X}_+))\Delta(\boldsymbol{X}_+)\tilde{w}_+(\mathcal{L}(\boldsymbol{X}_+))[p_+(\cdot)]^{\boldsymbol{X}_+}. \tag{4.9}$$

Integrating (4.9) yields the denominator

$$\begin{aligned}
&\int g(\mathcal{F}|\boldsymbol{X}_+)\boldsymbol{\pi}_+(\boldsymbol{X}_+)\delta\boldsymbol{X}_+ \\
&\quad= \int g(\mathcal{F}|\mathcal{L}(\boldsymbol{X}_+))\Delta(\boldsymbol{X}_+)\tilde{w}_+(\mathcal{L}(\boldsymbol{X}_+))[p_+(\cdot)]^{\boldsymbol{X}_+}\delta\boldsymbol{X}_+ \\
&\quad= \sum_{J\in\mathbb{L}_+} g(\mathcal{F}|J)\tilde{w}_+(J)\left[\int p_+(x_+,\cdot)\,\mathrm{d}x_+\right]^J \\
&\quad= \sum_{J\in\mathbb{L}_+} g(\mathcal{F}|J)\tilde{w}_+(J)
\end{aligned} \tag{4.10}$$

where the second equality follows from applying Lemma 3 from [VV13]. Inserting (4.9) and (4.10) into (4.5) yields (4.7) and (4.8). □

The formulation (4.6) makes the assumption that the overlap probability $p_O(\ell_i, \ell_j)$ is a function of the labels of the involved objects and does not depend on their state. This assumption is crucial for finding a simple analytical solution for computing the denominator (4.10) which avoids full consideration of the state uncertainties and the accompanying complexity. However, the vehicle states certainly play a central role in determining the overlap of two vehicles and therefore, the approximate overlap probability

$$p_O(\ell_i, \ell_j) = \begin{cases} 1, & \text{if } A(\bar{x}_+^{(\ell_i)}) \cap A(\bar{x}_+^{(\ell_j)}) \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{4.11}$$

is chosen. It uses a function $A(x)$ which returns the occupied area of a vehicle with state x and passes the expected values of the prior state distributions from (4.2)

$$\bar{x}_+^{(\ell)} = \mathbb{E}[p_+^{(\ell)}(x)] \tag{4.12}$$

to this function. These expected values are indexed by the vehicle labels, can be calculated beforehand, and are not dependent on particular object states in $\boldsymbol{X}_+$. Of course, the approximation of the actual states by the expected value introduces errors if the state density is rather broad. In such situations it may be the case that certain vehicle state combinations which either cause or avoid an overlap are neglected.

As a result, the two prediction steps output a prior multi-object state distribution which is in GLMB form. In contrast to the standard multi-object transition model that is used in the LMB and extended object LMB filters and which predicts objects independently, the check for physical consistency introduces object dependencies. Multi-object hypotheses with overlapping vehicles (in terms of their mean states) are excluded. Thus, some vehicles may only exist in multi-object hypotheses that do not contain another overlapping vehicle or vice versa.

4.3.2 Sensor Modules

The sensor modules receive the predicted GLMB distribution and comprise all processing steps that involve sensor measurements or are sensor-specific. This includes computing the posterior GLMB distribution based on the received measurements, initializing new object hypotheses, and approximating the posterior GLMB distribution by an LMB distribution.

The computation of the posterior GLMB distribution in the update step implements (3.13) and strongly depends on the multi-object measurement model. This is particularly the case as this thesis avoids using a fixed multi-object likelihood model and argues that different formulations may be appropriate for the different sensor types. Therefore, the ability to include different multi-object measurement models is crucial and the placement of the entire multi-object measurement model into the sensor modules provides the necessary flexibility.

As initially proposed in [RVVD14] for the LMB filter, the posterior GLMB distribution is reduced to a posterior LMB distribution in the approximation step. Thus, a combinatorial explosion of GLMB multi-object hypotheses over time is avoided. For this purpose, the approximation scheme from Section 3.3.3 is used. As the set of multi-object hypotheses depends on the chosen multi-object measurement model, the resulting approximation equations slightly differ for the sensor modules. The posterior LMB distribution then contains multiple vehicle hypotheses from the label space $\mathbb{L}_+$ with updated state densities and existence probabilities.

Apart from the posterior LMB distribution, the sensor modules output a birth density which is also in LMB form and comprises new vehicle hypotheses. The initialization routine is again sensor-dependent and only implemented for the lidar and radar modules. The common rationale is that objects are initialized for all measurements which have not considerably contributed to updating an existing vehicle but which indicate a possible vehicle. All new vehicle hypotheses receive labels from the label space $\mathbb{B}$, an initial existence probability $r_B^{(\ell)}$, and an initial state distribution $p_B^{(\ell)}(x)$. Thus, the parameter set of the birth density is $\boldsymbol{\pi}_B = \{(r_B^{(\ell)}, p_B^{(\ell)}(x))\}_{\ell \in \mathbb{B}}$.

More detailed information about the measurement models used in the respective sensor modules are presented in Chapter 5 and Chapter 6 provides implementation details for the initialization routines.

4.3.3 Track Management

After the posterior LMB density and birth density have been transmitted by the sensor module, the track management step post-processes the posterior density by triggering resampling procedures if necessary and pruning object hypotheses with low existence probabilities from the density. This results the posterior label space $\mathbb{L}$ which excludes the pruned labels. The posterior multi-object density is then output and used to obtain estimates for the cardinality and each vehicle's state.

Before passing the multi-object distribution to the next prediction step, the birth

density is appended to the posterior LMB distribution. Thus, the label spaces of both distributions, i.e. $\mathbb{L}$ and $\mathbb{B}$, which comprise the present vehicle hypotheses are combined. This leads to the augmented prior label space $\mathbb{L}_+ = \mathbb{L} \cup \mathbb{B}$ that is used in the next iteration.

Chapter 5

Extended Object Measurement Models

Extended object measurement models that are tailored to the measurement principles of the different sensors are the core components of the sensor modules. This chapter describes the measurement models that were developed for radar, lidar, and semantic labeling data. The chapter starts by introducing the lidar measurement model in Section 5.1 and then continues to present the measurement models for the semantic labeling module in Section 5.2 and the radar sensors in Section 5.3.

5.1 Lidar Measurement Model

As presented in Section 3.3.4, most extended object filters that are based on finite set statistics (FISST) utilize a generalization of the standard multi-object measurement model, which assumes that objects are detected with a certain probability of detection and, if detected, independently generate sets of measurements. Filters that have been derived using this measurement model have been widely used for lidar-based tracking in conjunction with different extended object models; see for example [GLO11; GRMS14; HSRD16; MBMW17; SADD14a]. To make the model applicable, all positive returns from the laser rays are collected and treated as a set of detection-type measurements.

This practice discards all negative information that is contained in the measurements that did not yield a return, i.e. information about the absence of objects as discussed in Section 4.1.2. Yet, the use of negative information may convey additional information and the benefits have been studied in a variety of tracking applications, e.g. [Koc04]. The standard extended object measurement model additionally neglects the fact that the sensor scans the environment in a predefined pattern and therefore

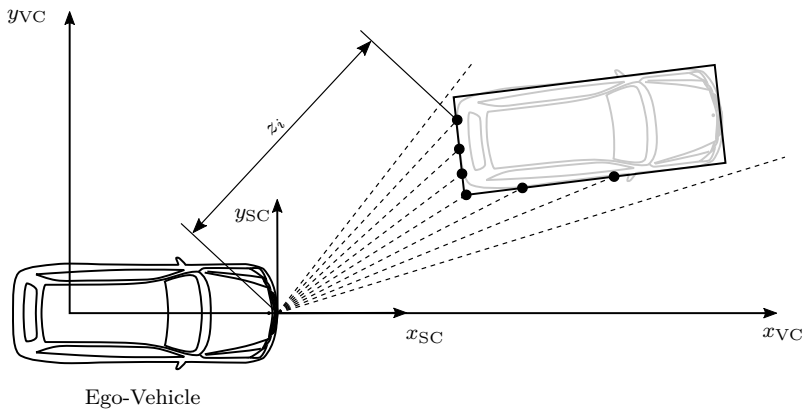


Figure 5.1: Schematic illustration of lidar measurements and the sensor coordinate (SC) system

provides a constant number of measurements in each laser scan. Furthermore, the measurement process depends on object constellations, for example in cases where one object occludes another. Consequently, objects do not generate measurements independently.

In contrast to the mentioned approaches, the goal of the lidar measurement model developed in this thesis is to accurately model the measurement process and to be able to exploit the known scan pattern as well as knowledge about the absence of objects. Therefore, a ray-based extended object model similar to [PT09] is combined with a separable likelihood approach for the multi-object portion. Preliminary results on this approach have been presented in the prior publications [SRD16; SRD17].

It is assumed that the lidar sensor operates with a single layer and therefore yields a 2D scan of the environment with known azimuth angle pattern, as illustrated in Fig. 5.1. For each ray, the sensor either provides a distance measurement z_i if it receives a return or the information that no object is present, i.e. $z_i = \nexists$. As the number of rays m is fixed, all measurements are combined in the measurement vector $z = [z_1, \ldots, z_m]^{\mathrm{T}}$.

5.1.1 Multi-Object Measurement Model

The predefined angular pattern of lidar measurements is very similar to the pixel patterns of images where the number of pixels is also constant and the angular region that a pixel covers is known. For such image tracking tasks, Vo et al. [VVPS10]

developed the separable likelihood model for FISST-based multi-object tracking. It works on a measurement vector instead of the classical measurement set that is assumed in the standard FISST literature. Different applications of the separable likelihood model to real-world image data were demonstrated in [HVV13; HVVS12]. The similarity of the measurement patterns makes the separable likelihood model a natural candidate for incorporating lidar data in a multi-object tracking framework.

The following assumptions constitute the original separable likelihood model:

1. The indices of measurements that are affected by an object can be computed for a given object state $\boldsymbol{x}$ and are denoted by the set $T(\boldsymbol{x})$.

2. Objects are separable in the measurement space. That is, $T(\boldsymbol{x}) \cap T(\boldsymbol{x}') = \emptyset$ holds for two distinct objects with labels $\ell \neq \ell'$.

3. The measurement likelihood of a single measurement z_i is

$$p(z_i|\boldsymbol{x}) = \begin{cases} \varphi_i(z_i|x), & i \in T(\boldsymbol{x}) \\ \psi_i(z_i), & i \notin T(\boldsymbol{x}) \end{cases}, \tag{5.1}$$

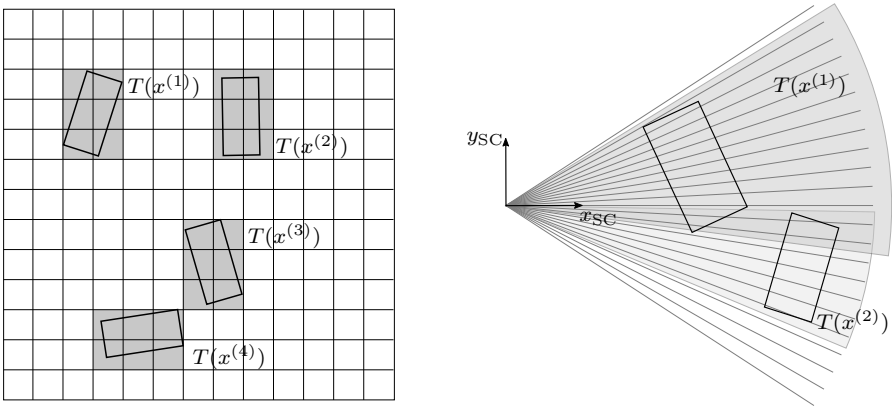
   where $\varphi_i(z_i|x)$ is the foreground likelihood that comes into effect if the measurement is affected by an object and $\psi_i(z_i)$ is the background likelihood.

4. The ray likelihoods are (conditionally) independent and the overall likelihood of the measurement vector z is thus

$$g(z|\boldsymbol{X}) = \Bigg(\prod_{\boldsymbol{x} \in \boldsymbol{X}} \prod_{i \in T(\boldsymbol{x})} \varphi_i(z_i|\boldsymbol{x}) \Bigg) \Bigg(\prod_{i \notin \cup_{\boldsymbol{x} \in \boldsymbol{X}} T(\boldsymbol{x})} \psi_i(z_i) \Bigg). \tag{5.2}$$

   Here, the first term multiplies all foreground likelihoods by iterating over all indices that are contained in any $T(\boldsymbol{x})$ and the second term multiplies the remaining rays' background likelihoods, as they were not affected by any object.

The assumption of separability in the measurement space (i.e. Assumption 2) is paramount for being able to compute independent and efficient object updates. It rules out any uncertainty in associating measurements to different objects and states that objects do not interfere in the measurement process. This assumption is valid in the initial tracking problem from [VVPS10]. It assumes that multiple extended objects, which cannot overlap, move in a common plane. This plane is evenly split and covered by the image pixels. As each pixel covers a particular portion of the plane and objects do not overlap, it can only be affected by a single object. See Fig. 5.2a for an illustration.



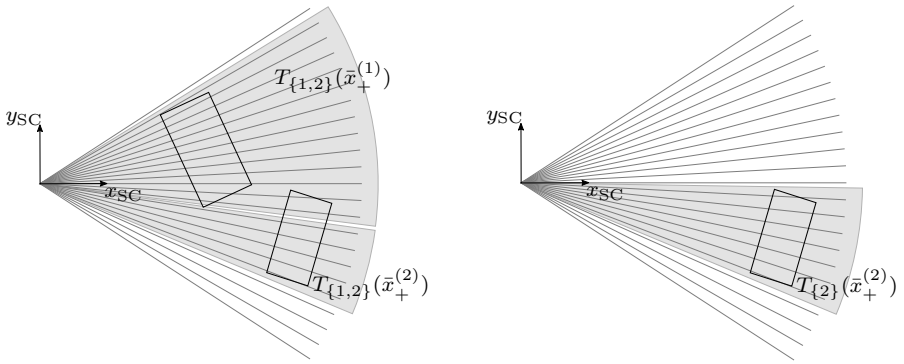
(a) Original model with pixel measurements and separable objects

(b) Separable likelihood model applied to lidar data

Figure 5.2: Schematic illustrations of the separable likelihood model for pixel and lidar measurements

In this work, the idea is to replace the original pixel measurements by distance measurements from an automotive lidar sensor. See Fig. 5.2b for an illustration. A measurement is considered affected by a vehicle, if the corresponding ray targets it. When using this formulation, however, Assumption 2 is no longer valid. While the objects are still extended and can physically not overlap, the lidar only provides measurements from a perspective on the ground. As illustrated in Fig. 5.2b, several vehicles may therefore cover the same azimuth range and some rays may be considered affected by multiple vehicles. Neglecting this violation and simply using rays for multiple objects results in estimation errors in the number of objects. See the prior publication [SRD16] for an example.

Therefore, this thesis extends the separable likelihood model by including an occlusion model. The measurement index set $T(\boldsymbol{x})$ is made dependent on all present objects and the affected rays are determined for each object considering possible occlusions caused by other vehicles. If a ray hits two objects, it is only assigned to the measurement index set of the front vehicle. Thus, the affected rays have to be computed for each multi-object state hypothesis individually. To avoid additional combinatorial complexity that arises from the uncertainty in the object state, the occluded areas are again approximately computed from the prior mean states (4.12) of the involved objects. As in the second prediction step in Section 4.3.1, these mean values are indexed by the object labels and $T(\boldsymbol{x})$ is replaced by $T_I(\bar{x}_+^{(\ell)})$ where the subscript I denotes the dependence on the set of present object labels I. The process is illustrated in Fig. 5.3. Introducing this procedure allows the multi-object



(a) I = {1, 2} (b) I = {2}

Figure 5.3: Computing the affected rays with the new occlusion model for different multi-object state hypotheses with label set I

measurement model to handle occlusion situations inherently and to only use the information from rays that can actually see the vehicle. In case a vehicle is in full occlusion, $T_I(\bar{x}_+^{(\ell)})$ returns an empty set and the presence or absence of the vehicle does not influence the multi-object measurement likelihood. Thus, the update does not affect the vehicle and only the prediction step gradually decreases its existence probability.

With the proposed modification, the multi-object likelihood from (5.2) becomes

$$g(z|\boldsymbol{X}) = \Bigg(\prod_{\boldsymbol{x}\in\boldsymbol{X}} \prod_{i\in T_{\mathcal{L}(\boldsymbol{X})}(\bar{x}_+^{(\ell)})} \varphi_i(z_i|x) \Bigg) \Bigg(\prod_{i\notin\cup_{\boldsymbol{x}\in\boldsymbol{X}} T_{\mathcal{L}(\boldsymbol{X})}(\bar{x}_+^{(\ell)})} \psi_i(z_i) \Bigg). \tag{5.3}$$

The first term still multiplies the likelihoods of all rays that were affected by an object and are contained in one of the sets $T_{\mathcal{L}(\boldsymbol{X})}(\bar{x}_+^{(\ell)})$. Yet, these sets are now depended on the constellation of all present objects. Similarly, the second term multiplies the background likelihood of all rays that are not contained in any $T_{\mathcal{L}(\boldsymbol{X})}(\bar{x}_+^{(\ell)})$. Equation (5.3) can be rearranged to

$$g(z|\boldsymbol{X}) = f(z) \prod_{\boldsymbol{x}\in\boldsymbol{X}} g_{\mathcal{L}(\boldsymbol{X})}(z|\boldsymbol{x}), \tag{5.4}$$

where the first term

$$f(z) = \prod_{i=1}^{m} \psi_i(z_i) \tag{5.5}$$

contains the background likelihood of all rays in the scan. The second term multiplies

$$g_I(z|\boldsymbol{x}) = \prod_{i \in T_I(\bar{x}_+^{(\ell)})} \frac{\varphi_i(z_i|x)}{\psi_i(z_i)}, \tag{5.6}$$

with $I = \mathcal{L}(\boldsymbol{X})$, for all objects. This function contains the ratio of foreground and background likelihood of all rays that are affected by an object $\boldsymbol{x}$ and, hence, cancels the respective background likelihoods from the background term $f(z)$. Note that if $T_I(\bar{x}^{(\ell)}) = \emptyset$, $g_I(z|\boldsymbol{x}) = 1$.

In [PVV+15], Papi et al. demonstrated that generalized labeled multi-Bernoulli (GLMB) distributions are a conjugate prior to the original separable likelihood model. As the introduced modifications do not affect the line of argument that is made in the proof, this is also true for the extended version. The posterior multi-object density is obtained by inserting the multi-object likelihood from (5.4) to (5.6) and the prior multi-object distribution from (4.7) and (4.8) into the multi-object Bayes update (3.13). Following [PVV+15, Proposition 1], the posterior multi-object density is[1]

$$\boldsymbol{\pi}\left(\boldsymbol{X}|z\right) = \Delta(\boldsymbol{X}) w(\mathcal{L}(\boldsymbol{X})) \left[p_{\mathcal{L}(\boldsymbol{X})}(\cdot|z)\right]^{\boldsymbol{X}}. \tag{5.7}$$

Substituting $I = \mathcal{L}(\boldsymbol{X})$, $w(\mathcal{L}(\boldsymbol{X}))$ and $p_{\mathcal{L}(\boldsymbol{X})}(x, \ell|z)$ are

$$w(I) = \frac{w_+(I) \left[\eta_{I,z}(\cdot)\right]^I}{\sum_{J \in \mathbb{L}_+} w_+(J) \left[\eta_{J,z}(\cdot)\right]^J}, \tag{5.8}$$

$$p_I(x, \ell|z) = \frac{g_I(z|x) p_+(x, \ell)}{\eta_{I,z}(\ell)}, \tag{5.9}$$

with

$$\eta_{I,z}(\ell) = \int g_I(z|x) p_+(x, \ell) \, \mathrm{d}x. \tag{5.10}$$

In the original separable likelihood model, the update of the object state densities is conducted independently for each object and the complexity of the multi-object distribution does not grow over time. In contrast, the modified separable likelihood model results in multiple state densities for a single object due to the different possible object constellations, as indicated by the index I in (5.9) and (5.10). Note, however, that the likelihoods of individual measurements only have to be computed once for each object and can then be assembled using (5.6) for the different index

[1] This formulation of a GLMB distribution slightly extends the original definition which does not admit state densities that are indexed by the labels of the multi-object state $\mathcal{L}(\boldsymbol{X})$. Yet, it was chosen for notational simplicity. In fact, an equivalent formulation using the standard sum over multiple realizations could be constructed and used instead.

sets. Also, the multiple state densities are immediately combined to a single state density in the labeled multi-Bernoulli (LMB) approximation step that follows the update. See Section 5.1.3 for more details.

5.1.2 Lidar Ray Model

To complete the lidar measurement model, both the foreground and background likelihoods in (5.6) need to be specified. For each ray, the measurement z_i may either take a continuous value in the interval between zero and the maximum range of the lidar sensor $z_{\max}$ or it is set to $\nexists$ if the measurement did not yield a return. Therefore, the background likelihood for rays that do not target a vehicle consists of two parts and is given by

$$\psi_i(z_i) = \psi(z_i) = \begin{cases} 1 - p_{\mathrm{R,bgr}}, & \text{if } z_i = \nexists \\ p_{\mathrm{R,bgr}} / z_{\max}, & \text{otherwise} \end{cases} . \tag{5.11}$$

The probability of receiving a return $p_{\mathrm{R,bgr}}$ is set to 0.5 and a uniform density over the interval from zero to the maximum sensor range $z_{\max}$ is used if a distance reading is obtained. Thus, the background model is made agnostic with regards to the events of receiving or not receiving a return and to where a distance measurement occurred.

The likelihoods for rays that target a vehicle are determined by using a ray tracing approach that is strongly based on [PT09]. It employs a vehicle template which is constructed using three rectangles and depicted in Fig. 5.4. The two inner rectangles confine the vehicle surface where measurements are expected to occur. The outer rectangle limits the template and defines a free space around the vehicle in combination with the middle rectangle. As vehicles are assumed to be clearly delimited objects, no other objects are expected in the free space.

The likelihood structure for rays that target a vehicle is similar to the background model. It is given by

$$\varphi_i(z_i|x) = \begin{cases} 1 - p_{\mathrm{R},j}, & \text{if } z_i = \nexists \\ p_{\mathrm{R},j} \cdot p_j(z_i|x), & \text{otherwise} \end{cases} . \tag{5.12}$$

where $p_{\mathrm{R},j}$ again denotes the probability of receiving a return and $p_j(z_i|x)$ is a density over the possible distance values. The subscripts $j \in \{1, 2, 3\}$ indicate the type of ray and allow a parameterization of both parts in dependence on the path a ray takes through the vehicle template. Figure 5.4 shows that there are three different paths depending on which rectangles the ray intersects.

Type-1 rays only pass through the free space of the template and do not hit the

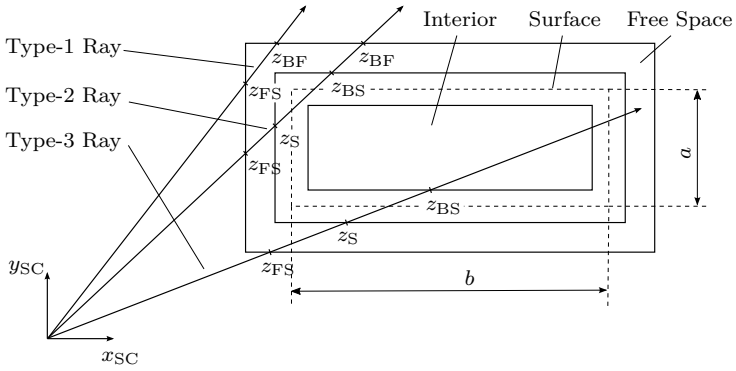
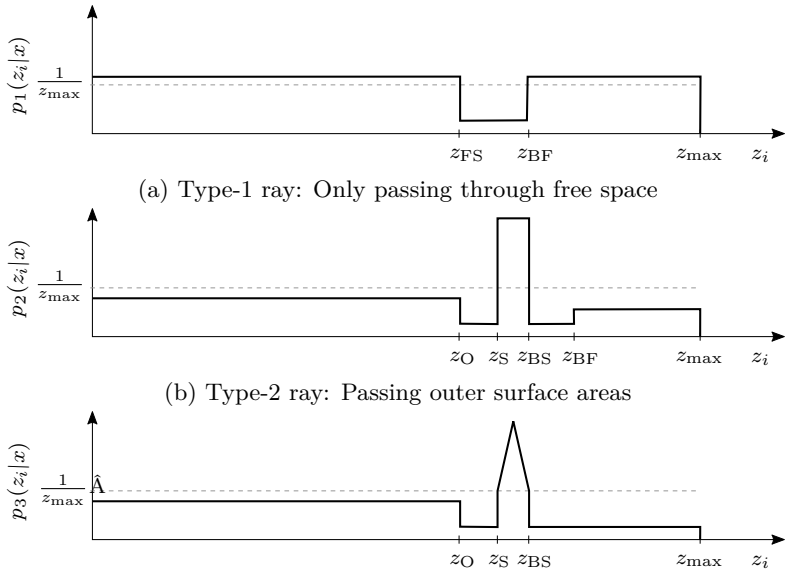


Figure 5.4: Vehicle template for the single object lidar model. Adapted and reprinted with permission from [SRD16] ©2016 IEEE.

vehicle directly. They are used to ensure that the vehicle is clearly delimited and that no measurement falls in the free space area next to the vehicle. The probability of receiving a return is set to $p_{\mathrm{R},1} = 0.5$ and therefore no preference is given to either event. If a distance measurement is received, its likelihood for falling in the free space area is low whereas the remainder of the density over the distance measurements $p_1(z_i|x)$ does not favor particular values. See Fig. 5.5a for an illustration.

Type-2 rays pass the free space and the outer surface areas. They do not intersect with the inner rectangle and therefore do not penetrate the interior. The distance density $p_2(z_i|x)$ (see Fig. 5.5b) favors measurements on the vehicle surface and penalizes measurements in the free space. To account for measurements from small objects in front of the vehicle, e.g. pedestrians or bicycles, which are not handled in the occlusion routine of the multi-object likelihood, measurements in front of the vehicle template receive a moderate likelihood. Because the rectangular shape of the surface is only an approximation of the actual vehicle contour, it cannot be ensured that rays passing the outer surface area actually hit the vehicle. Therefore, measurements behind the template are not penalized heavily and the probability of receiving a return is set to a value slightly larger than in the background model, i.e $p_{\mathrm{R},2}(z_S) > 0.5$. The exact value is furthermore made dependent on the distance to the vehicle surface z_S, to account for a decreasing probability of receiving a return with increasing distance.

Type-3 rays hit the vehicle body directly. The distance density $p_3(z_i|x)$ (see Fig. 5.5c) is very similar to that of type-2 rays but differs in two aspects: As laser rays are not able to penetrate the vehicle body, except for windows, measurements behind the vehicle surface are very unlikely and are assigned a low likelihood. Also, the density favors measurements close to the surface center in order to achieve an accurate



(a) Type-1 ray: Only passing through free space

(b) Type-2 ray: Passing outer surface areas

(c) Type-3 ray: Hitting the vehicle body

Figure 5.5: Qualitative illustration of the distance densities for the different ray types. Adapted and reprinted with permission from [SRD16] ©2016 IEEE.

angular alignment. The probability of receiving a return is set to large values, i.e. $p_{\text{R},3}(z_S) > p_{\text{R},2}(z_S) > 0.5$ and is again made dependent on the distance to the vehicle surface.

5.1.3 Approximation

In the approximation step, the posterior GLMB distribution from (5.7) to (5.10) is approximated by an LMB distribution with the new parameters

$$r^{(\ell)} = \sum_{I \subseteq \mathbb{L}_+} w(I) 1_I(\ell) \tag{5.13}$$

and

$$p^{(\ell)}(x) = \frac{1}{r^{(\ell)}} \sum_{I \subseteq \mathbb{L}_+} w(I) 1_I(\ell) p_I(x, \ell | z). \tag{5.14}$$

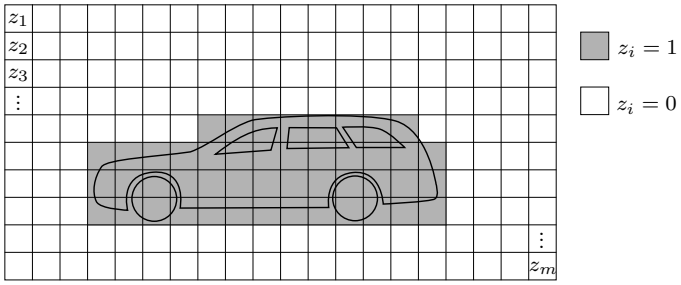


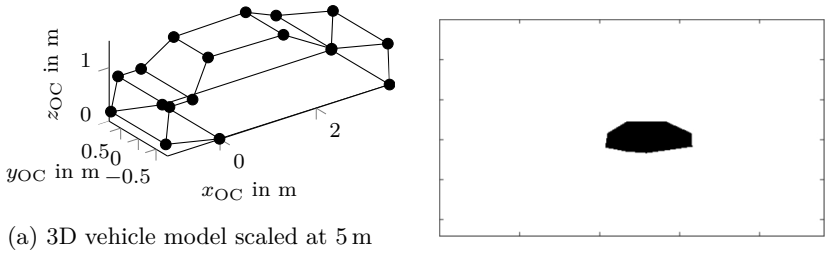
Figure 5.6: Illustration of scene labeling measurements

5.2 Scene Labeling Measurement Model

As a second measurement model, this thesis proposes a simple multi-object measurement model that allows exploiting the additional semantic information from scene labeled images for improving the vehicle tracking accuracy. The model was first published in the prior publication [SGRD17]. Similar to lidar data, the labeled images can be interpreted as a measurement vector with a fixed number of entries $z = [z_1, \ldots, z_m]$ by simply concatenating all m pixel labels. Each vector entry z_i is a binary variable which takes the value 1 if the pixel is labeled with the vehicle class or 0 if it belongs to any other class; see Fig. 5.6 for an illustration.

As the measurement vector has again a fixed size, the separable likelihood model from Section 5.1.1 is also used as multi-object likelihood for the scene labeling module. Bernoulli distributions (cf. (2.4)) over the binary measurements z_i are chosen for the foreground and background likelihood functions $\varphi_i(z_i|x)$ and $\psi_i(z_i)$, respectively. By adapting the Bernoulli distribution parameters r_φ and r_ψ, one can tune the likelihood function to the performance of the used scene labeling approach. The parameters indicate, how likely it is to obtain a vehicle class label for a pixel if the pixel actually shows a vehicle or if the pixel shows background. Note, however, that the separable likelihood model makes the important assumption of (conditionally) independent measurements. As this independence is not given in many scene labeling approaches, the Bernoulli parameters may need to be chosen conservatively. See Section 6.6.1 for a more detailed discussion.

To obtain all pixels $T(\boldsymbol{x})$ that are covered by a vehicle, a pixel mask is constructed using a simplified 3D vehicle model. It consists of a mesh with 18 points that roughly describe the shape of sedan. The points are given in a normalized object coordinate system and the model can therefore be scaled to different vehicle sizes. Figure 5.7a illustrates an exemplary vehicle model scaled to a length of 5 m, a width



(a) 3D vehicle model scaled at 5 m length, 1.8 m width, and 1.5 m height

(b) Exemplary vehicle mask $T(\boldsymbol{x})$ for a vehicle at 10 m distance

Figure 5.7: Mask generation for the scene labeling model [SGRD17]

of 1.8 m, and a height of 1.5 m. By projecting the 3D point cloud into the image plane and computing the convex hull around the projected points, a vehicle mask which indicates all relevant pixels $T(\boldsymbol{x})$ is constructed. See Fig. 5.7b. Determining the convex hull is computationally efficient but only provides approximate masks as the actual vehicle contour is not convex for certain aspect angles. As the intent of the 3D model is to only provide a coarse vehicle shape that works for different vehicle types and not to replicate the contour of a specific vehicle in all detail, this simplification is tolerable.

Due to the typical mounting position of cameras, vehicles are not generally separable in the measurement space and vehicles may occlude each other. Therefore, Assumption 2 of the separable likelihood model from Section 5.1.1 is again not satisfied and countermeasures need to be implemented to avoid estimation errors. As in the lidar measurement model, the scene labeling model therefore handles occlusion directly. The pixel masks are computed in dependence on the objects that are present in the multi-object state and on the expected value of the predicted states. Therefore, the set of affected measurements for a particular object $T(\boldsymbol{x})$ is again replaced by $T_I(\bar{x}_+^{(\ell)})$ and the multi-object likelihood is the same as the lidar model from (5.3), (5.5), and (5.6). Consequently, the updated GLMB distribution is given by (5.7) to (5.10) and the approximation equations are also identical to those of the lidar sensor module in (5.13) and (5.14).

5.3 Radar Measurement Model

The formulation of the radar model is based on the assumption of detection-type radar data, as previously stated in Section 4.1.1. That is, each sensor outputs a set of

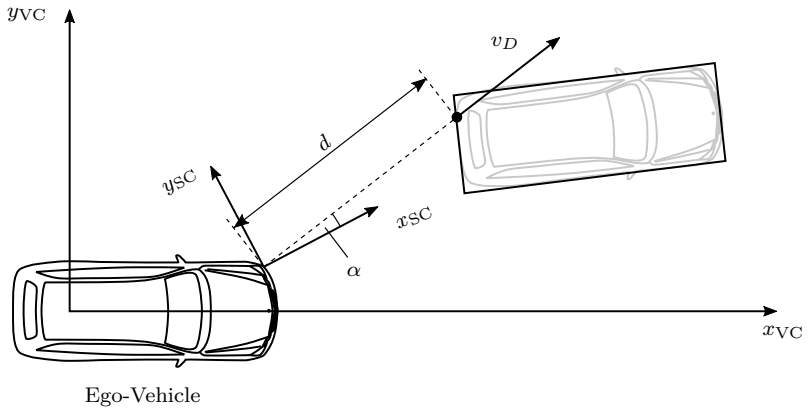


Figure 5.8: Schematic illustration of a radar measurement in sensor coordinates (SC)

radar detections $Z = \{z^{(1)}, ..., z^{(m)}\} \subset \mathbb{Z}$ which may originate from actual vehicles, are caused by sensor noise, or stem from irrelevant objects such as buildings, poles, or guard rails. In contrast to the other sensors, the number of measurements m typically varies from cycle to cycle. Each detection $z^{(i)} = [d^{(i)}, \alpha^{(i)}, v_D^{(i)}]^{\mathrm{T}}$ comprises the measured range $d^{(i)}$, azimuth angle $\alpha^{(i)}$, and Doppler velocity $v_D^{(i)}$. A schematic illustration is shown in Fig. 5.8.

The Doppler velocity measures the radial speed of an object relative to the radar sensor. It is computed by projecting the object velocity vector onto the line of sight from the sensor to the measurement location. For the vehicle model from Section 4.2, the object velocity at the location of the measurement is

$$v_O = \begin{bmatrix} v\cos(\varphi) - \omega\left(d^{(i)}\sin\left(\alpha^{(i)}\right) - y_R\right) \\ v\sin(\varphi) + \omega\left(d^{(i)}\cos\left(\alpha^{(i)}\right) - x_R\right) \end{bmatrix}. \tag{5.15}$$

It is composed of the translational portion from the longitudinal speed of the vehicle and a rotational portion due to the yaw rate. The translational portion is equal for all possible measurement locations, whereas the rotational portion depends on the lever arm between the center of motion, i.e. the center of the rear axle, and the measurement location. The radial component of the object velocity then equals the

Doppler velocity and is given by

$$\begin{aligned} v_D =& v_O^{\mathrm{T}} \cdot \begin{bmatrix} \cos\left(\alpha^{(i)}\right) \\ \sin\left(\alpha^{(i)}\right) \end{bmatrix} \\ =& \cos\left(\alpha^{(i)}\right) \left(v\cos(\varphi) - \omega \left(d^{(i)} \sin\left(\alpha^{(i)}\right) - y_R \right) \right) \\ &+ \sin\left(\alpha^{(i)}\right) \left(v\sin(\varphi) + \omega \left(d^{(i)} \cos\left(\alpha^{(i)}\right) - x_R \right) \right) \\ =& \cos\left(\alpha^{(i)}\right) \underbrace{\left(v\cos(\varphi) + \omega y_R\right)}_{s_{D,1}} + \sin\left(\alpha^{(i)}\right) \underbrace{\left(v\sin(\varphi) - \omega x_R\right)}_{s_{D,2}}. \end{aligned} \tag{5.16}$$

From (5.16), there are important observations to be made: First, the Doppler velocity only depends on the azimuth angle of the measurement and not on the range. Secondly, all five kinematic states of the vehicle affect the Doppler velocity. Thirdly, there is a considerable ambiguity due to the radial measurement principle and the kinematic state cannot be fully inferred from a single measurement. This affirms the problem of preprocessing routines when dealing with a single or few measurements as in Fig. 4.1a. Even if multiple measurements with different azimuth angles $\alpha^{(i)}$ are available, the equation does not allow for determining the individual states' values but only the values of the two terms $s_{D,1}$ and $s_{D,2}$. Also note that the above equation is only valid for the rigid parts of the vehicle body and it does not model deviating Doppler measurements from rotating wheels.

In the following, the multi-object portion for the radar measurement models is first presented. Then, two different single object models which can be incorporated in the multi-object likelihood are proposed.

5.3.1 Multi-Object Measurement Model

The detection-type nature of radar measurements and their mathematical representation as random finite set (RFS) make the generalization of the standard multi-object likelihood model to the extended object problem a suitable formulation for the multi-object portion of the measurement model. As discussed in Section 3.4.2, it extends the standard multi-object likelihood by allowing for multiple measurements from a single object. Hence, the underlying assumptions are (cf. Section 3.3.4)

1. An object may be detected by the sensor with the detection probability $p_D(\boldsymbol{x})$ or it is not detected (misdetection) with complementary probability $q_D(\boldsymbol{x}) = 1 - p_D(\boldsymbol{x})$.

2. If detected, the object gives rise to a set of measurements Z_O which is distributed according to $g(Z_O|\boldsymbol{x})$.

3. The measurement set Z is a union of object and clutter measurements and the clutter measurements follow a Poisson RFS $g_C(\cdot)$ with intensity function $\kappa(z) = \lambda_C p_C(z)$. That is, the expected number of clutter measurements is λ_C and they are independent and identically distributed (i.i.d.) according to the clutter density $p_C(z)$.

4. Each object measurement originates from a single object only and the object and clutter measurements are (conditionally) independent.

These assumptions yield the formulas of the multi-object measurement model [BRG+16]

$$g(Z|\boldsymbol{X}) = g_C(Z) \sum_{i=1}^{|\boldsymbol{X}|+1} \sum_{\substack{\mathcal{U}(Z)\in\mathcal{P}_i(Z) \\ \theta\in\Theta(\mathcal{U}(Z))}} \left[\psi_{\mathcal{U}(Z)}(\cdot|\theta)\right]^{\boldsymbol{X}} \tag{5.17}$$

with

$$\psi_{\mathcal{U}(Z)}(x,\ell|\theta) = \begin{cases} \frac{p_D(x,\ell)\, g(\mathcal{U}_{\theta(\ell)}(Z)|x,\ell)}{[\kappa(\cdot)]^{\mathcal{U}_{\theta(\ell)}(Z)}}, & \theta(\ell) > 0 \\ q_D(x,\ell), & \theta(\ell) = 0 \end{cases}, \tag{5.18}$$

and

$$g_C(Z) = e^{-\lambda_C}[\kappa(\cdot)]^Z. \tag{5.19}$$

Essentially, (5.17) determines the likelihood of the obtained measurements by summing over all variants of how the measurement set could have been composed and how well the measurements match the object or clutter models. For this purpose, the first sum iterates over the different hypotheses of how many objects from the multi-object state were actually detected. Then, different partition and association variants are constructed in the second sum. Each measurement partition $\mathcal{U}(Z)$ separates the measurement set into mutually exclusive clusters. One cluster contains all clutter measurements and each of the remaining clusters contains the measurements from a particular object. The set of all partitions that contain i clusters and therefore allow for the detection of $i-1$ objects is denoted by $\mathcal{P}_i(Z)$. For a particular partition, the association mapping $\theta : \mathcal{L}(\boldsymbol{X}) \to \{0, 1, \ldots, |\mathcal{U}(Z)|\}$ then assigns the objects to the measurement partitions or the case of misdetection under the assumption that each cluster in the partition may only be assigned to one object. That is, an object assigned to the index 0 is considered not detected and $\theta(\ell) = \theta(\ell') > 0$ implies $\ell = \ell'$. The space of all association mappings for a given partition is denoted by $\Theta(\mathcal{U}(Z))$.

The last term in (5.17) evaluates the measurement likelihood (5.18) for each object under the assumption of a given partition and association. If an object is detected,

i.e. $\theta(\ell) > 0$, the single object likelihood

$$g(Z_O|x,\ell) = e^{-\lambda_T}[\lambda_T g(\cdot|x)]^{Z_O} \tag{5.20}$$

is computed for the assigned cluster $Z_O = \mathcal{U}_{\theta(\ell)}(Z)$. Here, the number of measurements is assumed to be Poisson distributed with expected value λ_T. At the same time, the denominator of (5.18) cancels the associated measurements from the overall clutter term (5.19).

As shown in [BRG+16], GLMB densities are a conjugate prior to the multi-object measurement model from (5.17) to (5.19). For the prior GLMB distribution from (4.7) and (4.8), the posterior GLMB density is [BRG+16]

$$\boldsymbol{\pi}\left(\boldsymbol{X}|Z\right) = \Delta\left(\boldsymbol{X}\right) \sum_{i=1}^{|\boldsymbol{X}|+1} \sum_{\substack{\mathcal{U}(Z)\in\mathcal{P}_i(Z) \\ \theta\in\Theta(\mathcal{U}(Z))}} w_{\mathcal{U}(Z)}(\mathcal{L}\left(\boldsymbol{X}\right)|\theta)\left[p(\cdot|\mathcal{U}(Z),\theta)\right]^{\boldsymbol{X}} \tag{5.21}$$

with

$$w_{\mathcal{U}(Z)}(I|\theta) = \frac{w_+(I)\left[\eta_{\mathcal{U}(Z)}(\cdot|\theta)\right]^I}{\sum\limits_{J\subseteq\mathbb{L}_+}\sum\limits_{i=1}^{|J|+1}\sum\limits_{\substack{\mathcal{U}(Z)\in\mathcal{P}_i(Z) \\ \theta\in\Theta(\mathcal{U}(Z))}} w_+(J)\left[\eta_{\mathcal{U}(Z)}(\cdot|\theta)\right]^J}, \tag{5.22}$$

$$p(x,\ell|\mathcal{U}(Z),\theta) = \frac{p_+(x_+,\ell)\psi_{\mathcal{U}(Z)}(x_+,\ell|\theta)}{\eta_{\mathcal{U}(Z)}(\ell|\theta)}, \tag{5.23}$$

and

$$\eta_{\mathcal{U}(Z)}(\ell|\theta) = \int p_+(x_+,\ell)\psi_{\mathcal{U}(Z)}(x_+,\ell|\theta)\,\mathrm{d}x_+. \tag{5.24}$$

As a consequence of the multi-object likelihood, the posterior GLMB now contains distinct multi-object hypotheses for the different partitioning and association variants.

5.3.2 Direct Scattering Model

As discussed in Section 4.1.1, the lack of a clear geometric structure and the ambiguity in the Doppler measurements poses a challenge for preprocessing routines, especially in the case of few measurements. As this inverse path from measurements to a vehicle state hypothesis is difficult, the idea of the direct scattering model is to provide a simple extended object measurement model that uses the direct path and defines where measurements are likely for a given vehicle state. The approach is similar to the single object model used for lidar sensors and transfers the basic idea

to the radar realm. At the same time, the goal is a measurement model which is particularly tailored to the application of radar-based vehicle tracking and overcomes the limitations of existing approaches that were mentioned in Section 4.1.1. It fully incorporates the available Doppler information, uses a shape that is suitable for vehicles, and avoids the association problem and detailed knowledge about reflection characteristics that is required for reflection point models. Also, it is not restricted to particular traffic situations and is designed to work for cross-traffic or turning vehicles as well as ambiguous situations. First versions of the direct scattering model were presented in the master's thesis [Kni15] and in the prior publications to this thesis [KSD16; SKRD16].

In the direct scattering model, the likelihood function $g(z|x)$ that is required in (5.20) is factored into three distinct densities for the three measurement quantities,

$$g(z|x) = g(d, \alpha, v_D|x) = g(v_D|\alpha, x)g(d|\alpha, x)g(\alpha|x). \tag{5.25}$$

The azimuth density $g(\alpha|x)$ models which azimuth angles are likely for a given vehicle state, and the range density $g(d|\alpha, x)$ as well as the Doppler density $g(v_D|\alpha, x)$ model how likely particular range and Doppler measurements are, given the vehicle state and the azimuth angle of the measurement.

To determine the azimuth and range densities, the direct scattering model uses a rectangular vehicle model as depicted in Fig. 5.8. As shown in Fig. 5.9, the azimuth density is simply constructed by generating a trapezoidal density over the azimuth angle interval that is covered by the rectangle. That is, a uniform density is used for azimuth measurements that lie in between the azimuth angles of the most right and most left corner of the rectangle. The density values linearly decrease in a transition region to both sides of the rectangle to allow for measurements at azimuth angles slightly to the right and left of the rectangle. Outside the transition region, the likelihood is set to 0 as measurements that originate from the vehicle are expected close to the azimuth interval covered by the rectangle.

For obtaining the range density $g(d|\alpha, x)$, the rectangular vehicle model is intersected with an idealized radar ray with the given azimuth angle α. See Fig. 5.10 for an illustration. If a ray targets the vehicle directly, the two intersection distances with the rectangle, $d_{I,1}$ and $d_{I,2}$, are computed. The density over the possible distance measurements is then constructed using a Gaussian mixture with three components. One component is centered at $d_{I,1}$, one component at the center of the intersection interval, and the third component at $d_{I,2}$. The mixture weights and standard deviations are chosen such that measurements on the surface facing the sensor receive the highest likelihood while measurements inside the rectangle and on the averted surface receive lower likelihoods. Still, the model allows for such measurements as they may be caused from parts of the underbody or the wheels

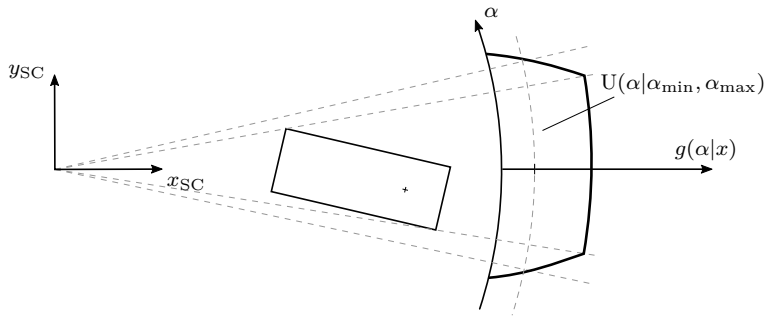


Figure 5.9: Schematic illustration of the azimuth density in the direct scattering model: The trapezoidal density is bended due to the polar representation. The uniform clutter density between the minimum and maximum azimuth angles $\alpha_{\min}$ and $\alpha_{\max}$ is also depicted as dashed line.

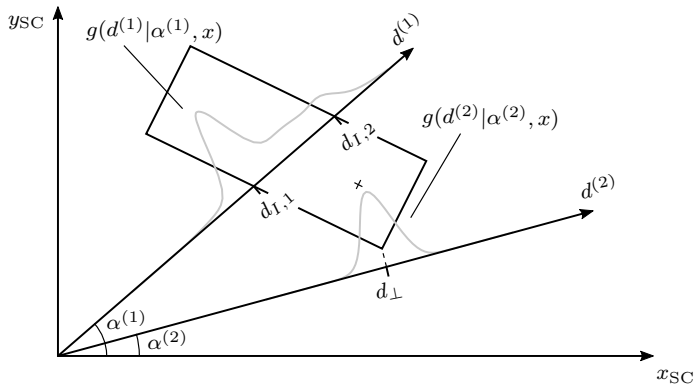


Figure 5.10: Construction of the range density $g(d|\alpha, x)$ in the direct scattering model. Adapted an reprinted with permission from [SKRD16] ©2016 IEEE.

on the far side. The distance density for rays that do not target the rectangle is modeled using a single Gaussian distribution centered at the projection point of the closest corner on the ray. The standard deviation of the Gaussian is linearly increased with the distance to the rectangle which results in a continuous transition towards a uniform density.

Lastly, the Doppler density $g(v_D|\alpha, x)$ for a given vehicle state and azimuth angle is a Gaussian distribution with the mean value set to the expected Doppler velocity as given by (5.16) and an appropriate standard deviation.

When using the direct scattering model, the clutter density $p_C(z)$ from the multi-object measurement model is set to a uniform distribution over the azimuth and range and a mixture of a uniform and a Gaussian density for the Doppler values. The Gaussian density is centered at a Doppler value of zero to emphasize that most clutter measurements stem from static objects.

To avoid overly complex computations, the direct scattering model uses rather simplified ray and rectangle models to determine expected measurement values. As a consequence, modeling errors are introduced. In practice, for example, a radar sensor may not be able to resolve reflections from different parts of the vehicle body and output merged detections that contain merged distance or Doppler values. Additionally, the actual vehicle shape may deviate from the rectangular form. It is therefore advisable to select standard deviations for the involved densities that are larger than the nominal sensor specifications to account for these simplifications.

5.3.3 Variational Radar Model

The major issue of existing radar measurement models from the literature as well as the proposed direct scattering model is that they are handcrafted. The models are either based on rather restrictive assumptions that make designing the model more convenient (e.g. by assuming an elliptical object shape) or use a more elaborate description at the expense of a certain modeling and implementation effort (e.g. reflection center models or the direct scattering model). Yet, the models still provide a simplified representation of the measurement likelihood and do not necessarily represent the actual distribution of the measurements. Also, manual adaption and tuning is necessary if certain measurement effects are to be included in the existing model. For example, none of the discussed models considers measurements from rotating wheels with strongly varying Doppler measurements and incorporating this effect would require additional modifications.

Despite for tracking, radar measurement models are also vital for simulation applications in which the goal is to simulate radar data, e.g. for testing purposes. For instance, reflection center models for simulation were derived in [SBW08] as well as [BY06] and the latter inspired the development of the reflection point models from [GSDB07; HSSS12] for tracking. Recently, there have been publications that pursue a more data-driven approach. A statistical study of vehicle radar measurement characteristics in dependence on the aspect angle was for example conducted in [BML$^+$17] and a simulation model for generating radar intensity grids based on deep learning has been presented in [WHWK17]. Also, [HHR$^+$15; HHR$^+$16] propose a simulation model based on kernel methods. Thus, a probabilistic measurement model that is constructed from previous measurement examples is achieved.

This thesis proposes a second radar-based vehicle model, the variational radar model, that transfers the idea of using machine learning techniques to tracking. By directly inferring the model from actual radar data, the goal is to overcome the shortcomings of previous approaches, i.e. to avoid manual engineering effort, to automatically capture different sensor effects, and to obtain a probabilistic model that is close to the true measurement process. The variational radar model was first presented in the prior publication [SD19].

The general underlying idea is to find the likelihood function $g(z|x)$ that is required in (5.20) as a conditional density of the joint distribution $p(z, x)$. That is,

$$g(z|x) = \frac{p(z, x)}{p(x)}, \tag{5.26}$$

where the marginal distribution $p(x)$ can be obtained by integrating $p(z, x)$ over z. The joint distribution $p(z, x)$ is learned from actual pairs of vehicle states and corresponding measurements using the variational Gaussian mixture (VGM) approach for density estimation from Section 2.4.2. This idea is similar to that of [HHR$^+$15; HHR$^+$16] but differs in the way the joint density is learned. Instead of using a kernel-based approach that requires storing all training data and online look-up operations, the VGM method outputs a more compact analytical Student's t-mixture representation. The resulting density is used as a spatial distribution model which can be easily incorporated into the tracking framework. Note that the same method could also be used to form a reflection point model if one interprets the mixture component centers as reflection points and the corresponding Student's t-densities as measurement noise. Yet, measurements would then have to be associated to particular reflection points during update. This is completely avoided by the spatial distribution approach. As discussed in the following, however, some modifications of this general principle are necessary to facilitate a practical implementation.

Dimension Reduction

As a consequence of the three-dimensional measurement space and the seven-dimensional state space, the combination of both spaces, over which the joint density $p(z, x)$ is defined, is ten-dimensional. To be able to capture all basic relationships between the vehicle state and the radar measurements, VGMs require data from all relevant portions of this joint state, despite their ability to generalize to a certain extent. That is, recordings of vehicles at different locations in the sensor field of view (FOV), with different orientations, speeds, yaw rates, and sizes would be necessary to learn typical measurement locations and to capture the nonlinear relationship between vehicle motion and Doppler velocity.

By reducing the dimension of the joint space, the amount of necessary training data can be considerably decreased and the training process is simplified. Therefore, both the measurements and the vehicle state are preprocessed using transformation functions and the joint density is then learned over the reduced space. The nonlinear transformation function for measurements is

$$z' = \begin{bmatrix} z'_x \\ z'_y \\ z'_d \end{bmatrix} = f_z(x, z) = \begin{bmatrix} z_{x,OC}/b \\ z_{y,OC}/a \\ v_D - \tilde{v}_D \end{bmatrix}, \tag{5.27}$$

where $\tilde{v}_D$ is the expected Doppler velocity given by (5.16) and $z_{x,OC}$ as well as $z_{y,OC}$ are the Cartesian positions of the measurement in the object coordinate system. Essentially, this function transforms all measurements to a normalized object coordinate system and computes the deviation of the Doppler measurement from the predicted value. Moreover, the vehicle state is transformed using

$$x' = f_x(x) = \varphi - \operatorname{atan2}(y_R, x_R). \tag{5.28}$$

Hence, it is reduced to a quantity x' that approximately equals the aspect angle under which the sensor sees the vehicle. Note that the compact notation of the transformation functions conceals that their computation involves all vehicle states. The yaw angle not only enters through (5.28) but is, in combination with the vehicle position, also necessary in (5.27) for transforming the measurements to the object coordinate system. Additionally, all states from the kinematic portion of the state vector are used for computing the expected Doppler velocity.

While this dimension reduction procedure is manually designed, it can be viewed as a means to incorporate prior knowledge about the sensor behavior. Subtracting the expected Doppler velocity avoids the necessity of relearning this well-known but nonlinear effect. Additionally, transforming measurements to a normalized object coordinate system implies the insight that the relative location of measurements does not strongly depend on the position of the vehicle in the FOV but rather on the aspect angle under which the sensor sees the vehicle. Interesting improvements of the proposed approach that have, however, not been tested in this thesis, would be to include the distance to the vehicle as additional variable or to use dimension reduction techniques such as [TSL00] that discover suitable manifolds themselves.

Computing the Measurement Likelihood

After dimension reduction, the VGM approach is applied to the set of training data Z_D where each individual training data point is given by $z_D^{(i)} = [z'^{(i)\mathrm{T}}, x'^{(i)}]^{\mathrm{T}}$. This

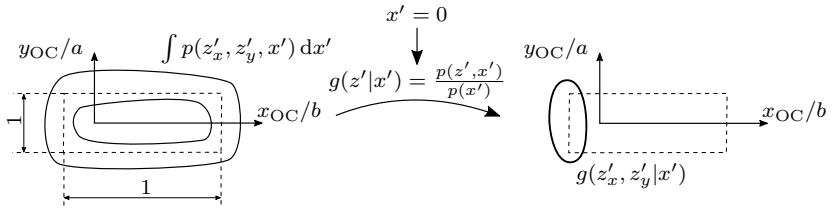


Figure 5.11: Schematic illustration of the variational radar model using the location of measurements as marginal example: The likelihood function is computed for an exemplary rear perspective by conditioning the joint density on the aspect angle $x' = 0$.

yields a predictive density $p(\tilde{z}', \tilde{x}'|Z_D)$ in the form of a Student's t-mixture (see (2.41)). In the following, the dependence on the training data and the tilde for indicating predicted values is omitted for brevity.

Determining the likelihood of radar measurements then involves two steps. First, the measurements need to be transformed to the reduced representation using (5.27) and (5.28). This involves the full object state and implicitly considers several aspects such as the distance to the vehicle or its speed and yaw rate. Afterwards, the likelihood function for the relative position of the measurements and the Doppler error conditioned on the aspect angle $g(z'|x')$ is computed as in (5.26), i.e. by dividing through the marginal density of the aspect angle. See Appendix A for the corresponding equations. The process is illustrated in Fig. 5.11 for the example of a rear perspective (i.e. $x' = 0$). For easy visualization, the illustration only depicts a schematic of the marginal density over the location of the measurements.

Incorporating the Variational Radar Model in the Multi-Object Likelihood

As a consequence of dimension reduction, the measurement likelihood $g(z'|x')$ is a density over a transformed measurement space that for instance defines the measurement location using Cartesian instead of polar coordinates. Yet, the multi-object likelihood model defines a density for the measurement set Z which is a finite subset of the original measurement space. The involved densities for single measurements therefore need to be defined over the original state space and simply inserting $g(z'|x')$ into (5.20) and subsequently into (5.18) is mathematically incorrect.

To illustrate the direct consequences, the ratio

$$\frac{g(Z_O|x,\ell)}{[\kappa]^{Z_O}} = \frac{e^{-\lambda_T}\lambda_T^{|Z_O|}}{\lambda_C^{|Z_O|}} \prod_{z\in Z_O} \frac{g(z|x)}{p_C(z)}, \tag{5.29}$$

that is computed in (5.18) for $Z_O = \mathcal{U}_{\theta(\ell)}(Z)$ in case an object has been associated with measurements, is reformulated. The right hand side directly reveals that the ratio is composed of two factors. One which is only influenced by the number of associated measurements and another factor which compares how well each measurement fits to the object and clutter likelihoods. If a density that is defined over a different space was simply inserted, a meaningful comparison between object and clutter measurements would be prohibited. Furthermore, the likelihoods of different vehicles would be rendered incomparable as this is based on a consistent comparison to the common clutter likelihood.

From the rules of computing derived distributions for transformed variables (see e.g. [GT08]), however, the identity

$$\frac{g(z'|x)}{p_C(z')} = \frac{g(f_z^{-1}(z',x)|x)\det\left(\frac{\partial f_z^{-1}(z',x)}{\partial z'_x \partial z'_y \partial z'_d}\right)}{p_C(f_z^{-1}(z',x))\det\left(\frac{\partial f_z^{-1}(z',x)}{\partial z'_x \partial z'_y \partial z'_d}\right)} = \frac{g(f_z^{-1}(z',x)|x)}{p_C(f_z^{-1}(z',x))} = \frac{g\left(z|x\right)}{p_C\left(z\right)} \tag{5.30}$$

can be derived. It states that the ratio between object and clutter likelihood remains equal if both densities are transformed using the same transformation function. Thus, the ratio in (5.18) and (5.29) can be replaced by the ratio between the measurement likelihood $g(z'|x')$ from the VGM approach and the transformed clutter density while concurrently ensuring proper comparisons between vehicle and clutter densities as well as among different vehicles. The inverse transformation function $f_z^{-1}(z',x)$ which is required for deriving the identity does exist but is not defined at the location of the sensor origin due to the ambiguity in transforming the origin to a polar representation. Since measurements at the sensor origin are irrelevant in practical applications, this pathological case is neglected. Also note that $g(z'|x) = g(z'|x')$ as x contains all information that is encoded in the aspect angle x'.

To further simplify the computation of the likelihood ratio, this thesis avoids the transformation of the clutter density from the original measurement space to the reduced space. Instead, it is assumed that clutter measurements are uniformly distributed over the Cartesian sensor coordinate system while the clutter density for the Doppler values remains identical to that of the direct scattering model. The transformation from a Cartesian sensor coordinate system to the normalized object coordinate system only involves scaling the density by the factor $a \cdot b$. This factor originates from the scaling of the coordinate system by the vehicle size and is

computed by evaluating the determinant of the inverse transformation function (cf. (5.30)). As the overall clutter density $g_C(Z)$ from (5.19) is canceled in the update step of the multi-object Bayes filter and does not appear in the posterior GLMB density, a transformation of this Cartesian clutter density to the original space is not necessary.

5.3.4 Approximation

Due to the multiple partitioning and association hypotheses, there may be many components in the posterior GLMB distribution from (5.21) to (5.24). The approximation step reduces these components to the simpler posterior LMB distribution with parameters

$$r^{(\ell)} = \sum_{I \subseteq \mathbb{L}_+} \sum_{i=1}^{|I|+1} \sum_{\substack{\mathcal{U}(Z) \in \mathcal{P}_i(Z) \\ \theta \in \Theta(\mathcal{U}(Z))}} w_{\mathcal{U}(Z)}(I|\theta) 1_I(\ell) \tag{5.31}$$

and

$$p^{(\ell)}(x) = \frac{1}{r^{(\ell)}} \sum_{I \subseteq \mathbb{L}_+} \sum_{i=1}^{|I|+1} \sum_{\substack{\mathcal{U}(Z) \in \mathcal{P}_i(Z) \\ \theta \in \Theta(\mathcal{U}(Z))}} w_{\mathcal{U}(Z)}(I|\theta) 1_I(\ell) p(x, \ell|\mathcal{U}(Z), \theta). \tag{5.32}$$

Chapter 6

Implementation and Evaluation

In this chapter, the proposed extended object models and the sensor fusion framework are evaluated on experimental data. The chapter starts by presenting the experimental set-up as well as the evaluation data and methods. Section 6.2 then discusses the procedure for learning the variational radar model and the resulting density. Afterwards, the chapter turns its attention to tracking and presents general implementation details in Section 6.3, followed by an evaluation of radar-based and lidar-based tracking in Section 6.4 and Section 6.5. Finally, Section 6.6 discusses the performance of different sensor fusion variants including the radar, lidar and scene labeling models.

6.1 Experimental Set-Up

Three different experimental vehicles were used for evaluation. One experimental vehicle was used as ego-vehicle to record the sensor data and two reference vehicles were deployed for generating ground truth data. The following section further details the sensor set-ups of the three vehicles before the data itself is discussed subsequently.

6.1.1 Experimental Vehicles

The data for evaluation purposes is recorded with the experimental vehicle shown in Fig. 6.1. It is an Mercedes E-Class S212 and the vehicle is equipped with several proprioceptive as well as exteroceptive sensors. An illustration of the fields of view (FOVs) of the exteroceptive sensors is provided in Fig. 6.2.

Four short range radar sensors are mounted in the corners of the front and rear

Figure 6.1: Experimental ego-vehicle (Photo: Elvira Eberhardt, Ulm University)

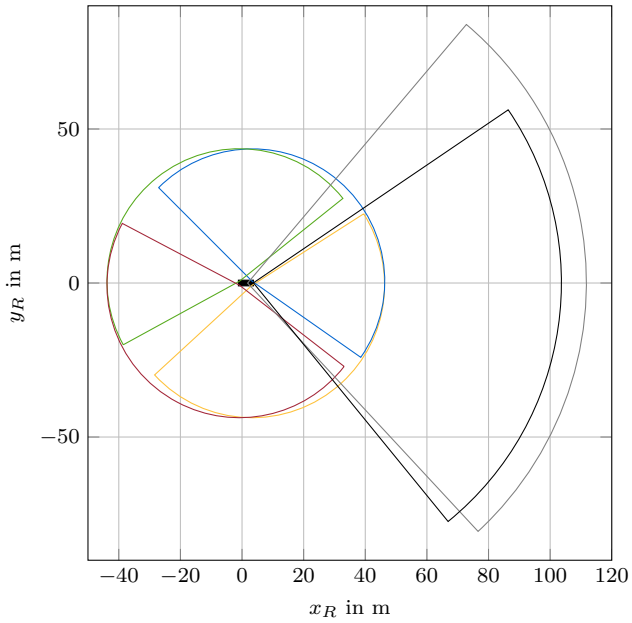


Figure 6.2: Sensor FOVs of the experimental vehicle: front left (—), front right(—), rear left (—), rear right (—) radar sensors, lidar (—), and camera (—)

bumpers. The FOV of each sensor has experimentally been determined to cover a range of 43 m with an opening angle of about 170°. All radar sensors are rotated to the sides by approximately 45° and hence cover the entire close-up range of the vehicle. The areas directly in front, behind, and next to the ego-vehicle are covered by two sensors, whereas small parts in between are only covered by a single sensor. The radars operate in the 76 GHz band and use chirp sequence modulation (cf. [Win16]). The azimuth angles are obtained by comparing the received signal from multiple antennas using a monopulse approach (cf. [Win16]). Each sensor provides 20 scans per second which contain at most 64 radar detections.

The four-layer Ibeo Lux 2010 lidar is additionally mounted in the center of the front bumper. The four scan layers are evenly spaced in the vertical direction and cover an elevation angle range of 3.2°. As the intersection with the ground plane is rather close to the ego-vehicle for the two lower layers, they are not used in this work and are therefore not further considered. The sensor is configured to provide range scans at a frequency of 12.5 Hz with an angular resolution of 0.5°. The azimuth angle coverage of the two upper layers is asymmetric and extends from −50° to the left to 35° to the right of the sensor axis. For each ray, the sensor is able to resolve three echoes. This can be advantageous if there are several objects at different distances within the vertical opening angle of the beam.

Moreover, a Baumer SXG21 camera with a charge-coupled device (CCD) imager is mounted behind the windshield. It provides images at a frequency of 15 Hz and a resolution of 1920x1080 pixels. Since it is equipped with a wide angle lens, the opening angle in azimuth direction is approximately 105°. However, the scene labeling data is only available for a portion of the entire image, as discussed in Section 6.6.1. This leads to the approximately 97° opening angle that is illustrated in Fig. 6.2.

To measure the ego-vehicle motion as well as its pose in a global coordinate system, a GeneSys Automotive Dynamic Motion Analyzer (ADMA), which combines an inertial measurement unit (IMU) and a differential global positioning system (DGPS), is mounted as proprioceptive sensor. It yields measurements at a rate of 50 Hz and in the best case reaches a root mean square accuracy in the position of up to 2 cm, of up to 0.015° in the yaw angle, and 0.0083 m/s in the speed. The measurements may, however, be less precise under imperfect conditions, e.g. if only few satellites are available. The ADMA data is used to determine the motion of the ego-vehicle in between measurements and to compute ground truth data (see below).

All sensors are calibrated with respect to the vehicle coordinate system of the ego-vehicle and the transformations from and to the sensor coordinate systems are hence known. As both the camera and the lidar sensor are triggered by external input signals, the recording times of these sensors' measurements are known relatively

precisely. In contrast, the radar sensors are running freely and their average latency caused by internal data processing and data transmission has been estimated individually. The latencies are compensated for by adjusting the recording time stamp.

In most of the experimental scenarios, two additional experimental vehicles serve as target vehicles. These vehicles are a Mercedes E-Class S212 and a Mercedes C-Class S205. Both are also equipped with a GeneSys ADMA IMU / DGPS system. All ADMA systems as well as the vehicle computers are synchronized to the time of the global positioning system (GPS). This allows for computing ground truth values for the vehicle states by transforming the pose and motion of the reference vehicles into the vehicle coordinate system of the ego-vehicle.

6.1.2 Experimental Data and Evaluation Methods

Four different data sets were recorded for evaluating the proposed tracking approach on real-world data: a training data set for learning the variational radar model as well as a single object data set, a multi-object data set, and data from a traffic scenario on public roads. The training data set is solely used for obtaining the variational radar model and is discussed in more detail in Section 6.2. In contrast, the remaining three data sets are used for assessing the tracking accuracy with respect to different performance criteria.

The single and multi-object data sets were recorded on a closed test site using the experimental ego-vehicle as well as the two reference vehicles. Therefore, precise ground truth is available for the number of visible vehicles, i.e. the true cardinality, and all vehicle states. The single object data set comprises ten different scenarios that were recorded with the E-Class reference vehicle. They cover both typical traffic situations such as cross-traffic, or turning and passing vehicles as well as two artificial scenarios in which the reference vehicle drives a complex and dynamic figure eight trajectory. The data set is used to assess the tracking accuracy for a single vehicle in detail without the influence of other objects.

The multi-object data set contains nine different scenarios with both reference vehicles. The scenarios emulate typical traffic situations such as various following and passing scenes, oncoming traffic, and cross traffic with occlusion. Using this data set, the accuracy of the cardinality estimate is evaluated and effects that are caused by object dependencies are studied.

The public road scenario was recorded at a T-intersection in an urban setting. It contains eleven different vehicles which range from small cars over sedans and

Table 6.1: Duration of the evaluation data sets and the contained number of measurement steps for each sensor type

	Single Object Data Set	Multi-Object Data Set	Real World Scenario	Total
Radar	16,956	17,491	2,280	36,727
Lidar	2,652	2,736	356	5,744
Scene Labeling	3,154	3,288	416	6,858
Duration	3 min 32 s	3 min 39 s	28.5 s	7 min 39 s

convertibles to vans. Thus, this scenario is not limited to the reference vehicle types and is used to illustrate the ability to generalize to other vehicles. Yet, no accurate ground truth is available for this data. Instead the vehicle poses and dimensions were manually labeled using the front lidar sensor of the ego-vehicle, which provides the most accurate image of the environment. The labels are only available in the lidar's FOV, i.e. in front of the ego-vehicle.

Table 6.1 lists the number of measurement steps of each sensor type in the data sets as well as the duration. Since the four radar sensors have the highest update rate, the number of radar measurement steps is considerably larger than those of the other sensors.

To assess the performance of the tracking framework, estimates are extracted for each object state as well as the cardinality in a minimum mean square error (MMSE) sense. Thus, both the single object accuracy and multi-object performance are considered. The expected values of the single object densities are computed to obtain MMSE estimates of the vehicle states and the root mean squared errors (RMSEs) are computed for each state by comparing the estimates to the ground truth. In some scenarios, the reference vehicles are at standstill and then start to slowly move. In such cases the initialization routines are not always able to pick up the correct orientation of motion. As it is assumed that vehicles only move in the forward direction, it may therefore happen that vehicles are initialized with an orientation offset of 180°. To avoid that this speed estimation error affects the orientation and position estimates, the evaluation considers both orientation hypotheses and selects the correct one for cases where the ground truth speed is below 1.5 m/s.

The cardinality estimate is also obtained in a MMSE sense by computing the expected value of the posterior cardinality distribution. It is compared to the ground truth cardinality which is obtained by counting the reference vehicles that are in the FOV of the used sensors and have a speed above 1 m/s. This constraint is added since the tracking algorithm only initializes moving vehicles (cf. Sections 6.4.1 and 6.5.1).

To mitigate random effects caused by the particle implementation that is going to be introduced in Section 6.3.1, the scenarios are run multiple times. All numerical results are given as average values over all Monte Carlo runs, whereas the figures show—unless otherwise stated—the results of one exemplary run.

6.2 Learning the Variational Radar Model

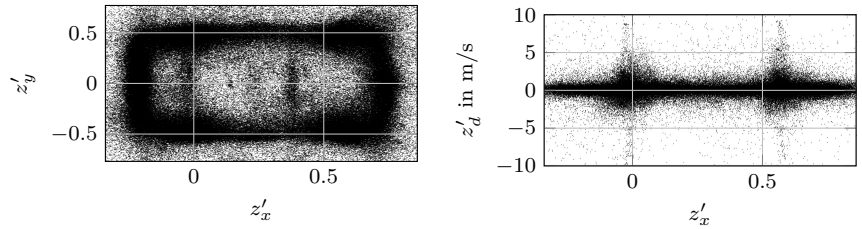
Before evaluating the actual tracking performance on the presented data sets, this section first discusses the procedure, the data set, and settings that were used for learning the variational radar model. Also, the resulting model is illustrated and discussed.

6.2.1 Data Set

One of the major issues in machine learning is obtaining the training data and annotating the data with ground truth values or labels. To avoid extensive manual effort, the availability of experimental vehicles with highly precise IMU/DGPS systems was exploited in this work. The Mercedes E-Class S212 reference vehicle was recorded using the four short range radar sensors of the ego-vehicle and the training data was generated by automatically associating the radar measurements with the ground truth states of the reference vehicle. Thus, a simple but effective procedure for generating training data is achieved. As consequence, however, the training data set only contains measurements from a single vehicle type which is not necessarily representative for all vehicles. Since radar data is not as detailed as lidar data or camera images, however, it is expected that the resulting model generalizes to other vehicles to a certain extent.

The training data set is disjunct from the remaining evaluation sequences and was recorded on public roads and a closed test site. It comprises typical cross-traffic or parallel traffic scenarios as well as artificial maneuvers, which were designed to achieve a broad coverage of the training data space. In these maneuvers, the reference vehicle for instance passed the ego-vehicle in straight lines at different distances and angles, drove small circles at different positions in the FOV, or circuited the ego-vehicle with different radii.

In cases where the ego-vehicle was at standstill, it was observed that its IMU/DGPS system had difficulties to accurately determine the global yaw angle. As small errors in the angle can translate to large errors in the ground truth position of the reference



(a) Position of the measurements in normalized object coordinates

(b) Excerpt of the Doppler error over the length axis of the vehicle

Figure 6.3: Illustrations of the training data for learning the variational radar model. Reprinted with permission from [SD19] ©2019 IEEE.

vehicle, some recorded sequences were manually post-processed before generating the training data. In cases where a considerable offset was noticeable, the yaw angle was corrected by matching the ground truth data and the precise measurements from the lidar sensor.

The data extraction routine projects the ground truth rectangle of the reference vehicle to the sensor coordinate systems. All measurements that fall in an enclosing rectangle that is enlarged by 0.5 m in each direction are then automatically associated to the vehicle. For each measurement and vehicle state pair, a training data point is then calculated by applying the transformation functions (5.27) and (5.28). The resulting $|Z_D|$ transformed data points constitute the training data set $Z_D = \{z_D^{(1)}, \ldots, z_D^{(|Z_D|)}\}$

The entire training data set was created from a total of approximately 123 minutes of recordings and consists of 336,287 data points. Two views of the data set are illustrated in Fig. 6.3. Figure 6.3a shows the positions of the measurements in the normalized object coordinate system and Fig. 6.3b depicts the Doppler errors over the normalized longitudinal axis. The plots show that most of the measurements originate from the vehicle surface and that the Doppler errors are particularly large at the positions of the front and rear axles, i.e. where one expects spurious wheel measurements.

Figure 6.4 shows a histogram over the aspect angle x' in the training data. While the data set contains roughly 20,000 observations that were taken from a rear perspective around $x' = 0°$ and many from both side perspectives, there are considerably less measurements from other aspect angles. For instance, the bin at $-5°$ only contains 4736 measurements. Experiments showed that this imbalance can cause noticeable

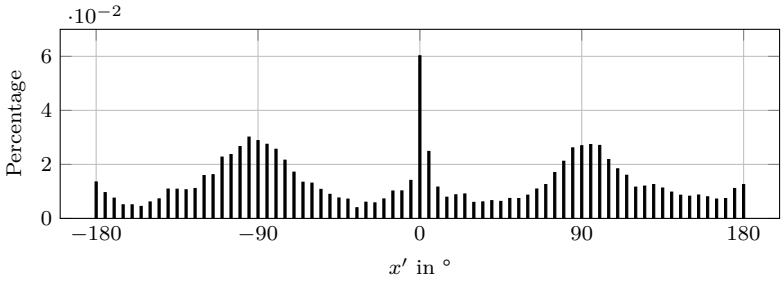


Figure 6.4: Histogram of the aspect angle x' in the training data set using bins with a width of 5°

bias in tracking as the resulting measurement model for example favors aspect angles around 0° over aspect angles around −5°. To avoid such issues, the training data set was therefore balanced by selecting an equal number of measurements from each aspect angle bin. This procedure discards many measurements and leads to a balanced data set with 95,688 data points. While this number of measurements proved to be sufficient for learning the model, it would be worthwhile to examine alternative balancing techniques that use the entire available data in future work.

Variational Gaussian Mixture Settings

For learning the variational radar model, an existing MATLAB implementation[1] of variational Gaussian mixtures (VGMs) was modified and used. The number of components of the Gaussian mixture was set to $c = 70$. Due to the lack of expert knowledge on the Gaussian mixture structure, the parameters of the prior distributions were set to yield uninformative priors. In particular, the Dirichlet hyperparameter in (2.35) was chosen as $\rho_0 = 1$. For the Gaussian-Wishart portion (2.36), the mean vectors γ_0 were set to the mean of all data points, $\beta_0 = 1$, $\nu_0 = n_{z_D} + 1$, where n_{z_D} is the dimension of the training data, and $\underline{V}_0$ was chosen as identity matrix.

6.2.2 Resulting Model

In the Student's t-mixture density $p(z_D) = p(z'_x, z'_y, z'_d, x')$ that results from the VGM approach, 20 out of the 70 components received mixing coefficients close to zero.

[1] original version by Mo Chen, https://de.mathworks.com/matlabcentral/fileexchange/35362-variational-bayesian-inference-for-gaussian-mixture-model

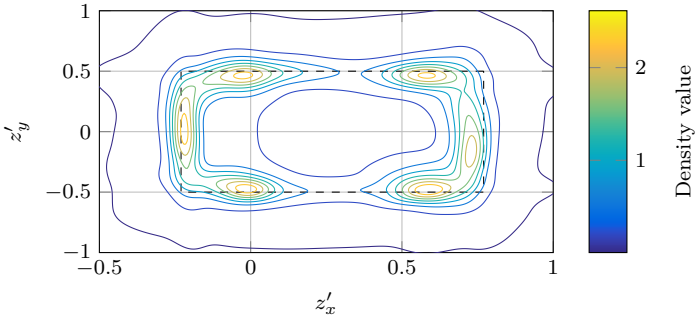


Figure 6.5: Plot of the marginal density $p(z'_x, z'_y)$. The outer contour is equivalent to the marginal clutter value for a vehicle of 2 m width and 5 m length. The dashed rectangle depicts the normalized vehicle dimensions.

They do not contribute to the overall density and only increase the computational demands when evaluating the density. Therefore, these components were pruned from the mixture density which results in a final mixture model with 50 components.

Since the four-dimensional predictive density is difficult to visualize, Fig. 6.5 illustrates the marginal density $p(z'_x, z'_y)$ in a contour plot. It corresponds to the view of the training data in Fig. 6.3a. The dashed rectangle indicates the normalized vehicle bounding box and the lowest contour value was set to the marginal clutter density if the vehicle had a size of 2 m by 5 m. While the model allows for spurious measurements around the vehicle contour and inside the vehicle, it has clearly learned that most measurements originate from the vehicle surface. Also, the centers of the front and rear surface as well as the four wheels or wheel houses are clearly identifiable as prominent measurement sources.

The actual measurement model that is derived from the predictive density by conditioning on a particular aspect angle x' using (5.26) is a three-dimensional density. Figure 6.6 depicts several examples of this density marginalized over the Doppler error, i.e. different variants of $g(z'_x, z'_y|x')$. These plots illustrate how a change in aspect angle affects the expected location of the radar measurements. Figure 6.6a shows the conditional density when looking at the vehicle from a slightly shifted front perspective. Measurements are expected over the entire front surface and particularly slightly to the right of the front center. Since the proposed VGM approach does not consider the periodic nature of the aspect angle, the front perspective exhibits a peculiarity at the $-\pi/\pi$-boundary. As the sign changes, there is an abrupt change in the involved mixture components. This transition has, however, not been noticeable when tracking and the mixture components on both

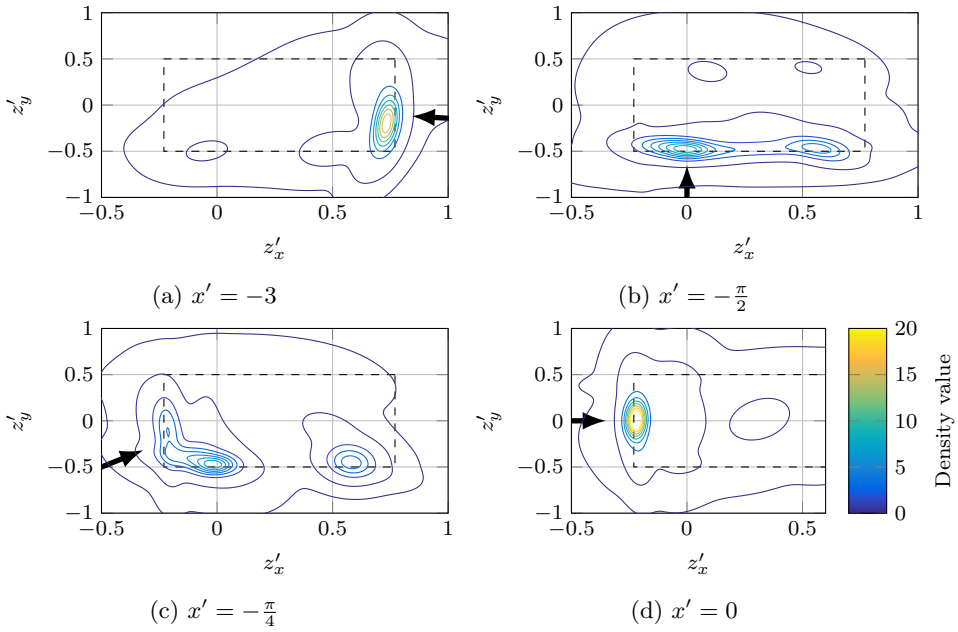


Figure 6.6: Plots of the marginalized conditional density $g(z'_x, z'_y|x')$ for different values of the aspect angle x'. The line of sight is indicated by the black arrows. The outer contour line equals the marginal clutter density for a vehicle of 2 m width and 5 m length. The dashed rectangle depicts the normalized vehicle dimensions.

sides expand over the entire front surface.

Figure 6.6b shows the conditional density when looking from the right, i.e. $x' = -\frac{\pi}{2}$. Measurements are mostly expected to originate from the right vehicle surface and in particular from the two wheel houses. When looking from the rear right perspective as illustrated in Fig. 6.6c, the expected locations of measurements generally shift towards the right rear corner of the vehicle. Yet, the two right wheel houses remain distinct features. From the rear perspective (see Fig. 6.6d), measurements are mostly expected around the center of the rear surface. In summary, the variational radar model has learned that measurements mostly originate from the surface side facing the sensor and from the wheels and wheel houses.

Figure 6.7 illustrates a different perspective of the conditional model by depicting the conditional density $g(z'_x, z'_d|z'_y = -0.5, x' = -\frac{\pi}{2})$, i.e. the density of the Doppler

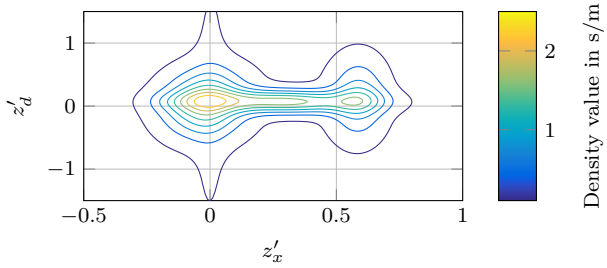


Figure 6.7: Conditional density $g(z'_x, z'_d | z'_y = -0.5, x' = -\frac{\pi}{2})$

errors over the right vehicle surface when looking from the right. The figure shows that the variational radar model expects larger Doppler errors at the position of the front and rear axis. This demonstrates that the model has learned the possible occurrence of spurious measurements from rotating wheels with deviating Doppler measurements.

Note that the presented variational radar model strongly depends on the training data and therefore the reference vehicle and radar sensor type. The transferability to other radar sensors has not been investigated in this thesis. Also, there is potential for further analysis with respect to the hyperparameters' or the number of component's influence on the model quality.

6.3 General Tracking Implementation Details

To demonstrate the performance of the measurement models and the sensor fusion framework, they were implemented in MATLAB. While the scripting language of MATLAB allows for simple implementation and evaluation, it is slow in comparison to lower-level languages such as C++ or parallelized programs for graphics processing units (GPUs). Therefore, the runtime of the algorithms is not in the scope of this thesis.

While Chapters 4 and 5 introduced the basic mathematical formulation, there are some additional factors such as the choice of model parameters or computing the prior and posterior single object densities that need to be considered. This section discusses some general implementation details that affect the entire tracking framework. Sensor-specific implementation details are introduced in the respective evaluation sections.

6.3.1 Particle Implementation for the Single Object Densities

Both the prediction and update steps alter the state densities of the individual objects. So far, it was assumed that equations (4.2), (5.9), and (5.23), which essentially conform to the prediction and update steps of the single object Bayes filter from (3.3) and (3.4), are solvable. As discussed in Section 3.1, this is not the case for general nonlinear object and measurement models. Instead, approximation techniques are necessary for such instances. Since the measurement models from Chapter 5 are heavily nonlinear, sampling-based approximations are therefore used in this work and the state densities of the vehicles are represented using particles.

In particular, an approximate Rao-Blackwellization technique (cf. Section 3.1.3) which avoids sampling from the entire seven-dimensional state space is used. It is inspired by the work in [PT09] and has been previously used in [Sch13]. Only the kinematic state portion ξ of the combined state vector $x = [\xi^T, \zeta^T]^\mathrm{T}$ is fully represented by particles. Hence, each particle represents a hypothesis for the vehicle's pose, speed, and yaw rate. Additionally, each particle holds a discrete distribution $p_i(\zeta|\xi^{(i)}, Z)$ over the extent portion ζ, which contains the width and length. As in (3.10), the posterior state densities are hence approximated as

$$p(\xi, \zeta|Z) \approx \sum_{i=1}^{n_p} w(\xi^{(i)}) \delta \left(\xi - \xi^{(i)}\right) p_i(\zeta|\xi^{(i)}, Z). \tag{6.1}$$

Due to the discrete distribution, the particles weights after the update are (cf. (3.11))

$$w(\xi^{(i)}) \propto \sum g(Z|\xi^{(i)}, \zeta) p_i(\zeta|\xi^{(i)}) \,\mathrm{d}\zeta, \tag{6.2}$$

where $g(Z|\xi^{(i)}, \zeta)$ is substituted by the appropriate likelihood function from (5.9) or (5.23).

Updating the discrete distribution proceeds in a simplified fashion. At the beginning of each filter cycle, the discrete distribution only holds a single hypothesis for the extent state with probability one. Before the update, new hypotheses are generated by varying the width and length in a sensor-specific routine. Depending on the observability of the width and length, respectively, new hypotheses with increased or decreased dimensions are generated locally around the current hypothesis and part of the probability mass is shifted to the new hypotheses. This results in a discrete extent distribution with a maximum of nine hypotheses and corresponds to a discrete transition density for the extent. After evaluating the likelihood for all hypotheses, the particle weight is determined using (6.2). Subsequently, the hypotheses are again reduced to a single hypothesis by computing the expected value. Hence, an increase of size hypotheses over time is avoided.

This approximate size estimation routine is realized at the cost of losing information from previous time steps and impedes that particles are able to capture the full extent uncertainty. Thus, detrimental effects on the filter consistency, which asks the estimation errors to follow the estimated state uncertainty, are expected. However, the procedure allows adapting the particle size in form of a simple and directed local search.

The number of particles n_p that is used for each vehicle's state density is set to 300. This number is increased by a factor of 2 and 3 for the lidar and radar model, respectively, when initializing a new track. This allows covering a larger portion of the state space when the initial state uncertainty is high. The initial number of particles is then gradually decreased in each resampling step. Resampling is triggered as soon as the effective sample size from (3.8) falls below a threshold of 0.3 and uses the technique of systematic resampling [RAG04, p. 42].

6.3.2 Ego-Motion Compensation

All vehicles are tracked in the vehicle coordinate system, i.e. in a coordinate system that is attached to the ego-vehicle and moves with it. Yet, the constant turn rate and velocity (CTRV) motion model only describes physically consistent motion in a globally fixed coordinate system. Therefore, an ego-motion compensation step is included before prediction. It transforms every vehicle from the vehicle coordinate system of the last to that of the current time step. The necessary transformation is computed from ego-motion information that is supplied by the ADMA of the ego-vehicle.

6.3.3 Vehicle Model

The process noise of the kinematic vehicle model (4.4), which was introduced in Section 4.3.1, is assumed to be uniform with zero mean. The maximum values of the uniform distributions are computed using the parameters σ_{x_R}, σ_{y_R}, σ_φ, σ_v, and σ_ω. They are listed in Table 6.2 and defined as the maximum deviation that may occur for the time interval of one second. This normalization to one second allows to adapt the process noise magnitude to the prediction time interval by multiplying with the time difference Δt. The major motivation behind using uniform distributions is to increase the chance of obtaining particles that cover corner cases, such as maximum deceleration during emergency braking maneuvers.

For computing the probability of persistence in the prediction step, the expected

Table 6.2: Process noise parameters

States	Maximum deviation
σ_{x_R} in m/s	3
σ_{y_R} in m/s	3
σ_φ in °/s	40
σ_v in m/s^2	9
σ_ω in °/s^2	3

disappearance time is set to 0.1 seconds and 10 seconds for objects outside and inside the sensor FOVs, respectively.

Possible values for the vehicle extent are restricted to reasonable dimensions of passenger cars. The limits are a minimum of 1.4 m and a maximum of 2.5 m for the width as well as a minimum of 2.5 m and a maximum of 7 m for the length. Moreover, the length-to-width ratio is constrained to values between 1.7 and 3.5. These restrictions cover vehicle sizes between a smart car and vans and limit the vehicle hypotheses to reasonable proportions. For computing new size hypotheses during the update step, the width is varied by ± 0.1 m and the length by ± 0.3 m.

It is furthermore assumed that vehicles only move in the forward direction. Therefore, the vehicle speed is defined to be always positive and vehicle hypotheses are flipped, i.e. their orientation is changed by 180°, as soon as the speed falls below 0 m/s.

6.4 Evaluation of Radar Tracking

In a first step, the performance of radar tracking is evaluated for both the direct scattering and the variational radar model. The framework uses four radar sensor modules to fuse the measurements of the four short range radar sensors. This section first introduces some radar-specific implementation details in Section 6.4.1, before presenting the single object and multi-object accuracy of both approaches in Sections 6.4.2 and 6.4.3.

6.4.1 Radar-Specific Implementation Details

The probability of detection that is required in the multi-object likelihood formulation of (5.17) and (5.18) is generally set to 0.8. Yet, the value is decreased to a minimum

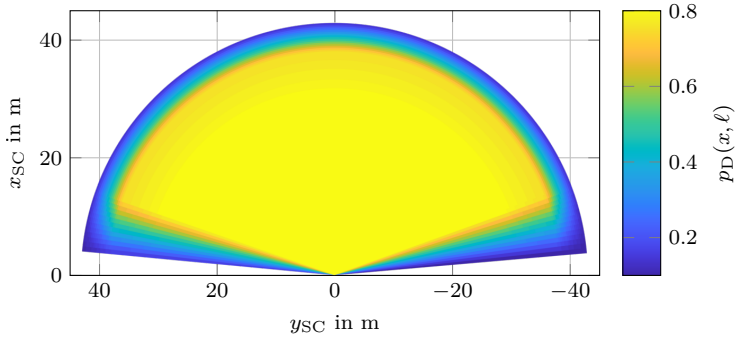


Figure 6.8: Detection probability $p_D(x, \ell)$ over the radar sensor FOV

value of 0.1 in the vicinity of the FOV boundaries to account for the reduced detection capability of the sensors in these areas and to achieve a smooth transition to areas outside the FOV. See Fig. 6.8 for an illustration. Moreover, the expect number of clutter measurements is set to $\lambda_C = 30$ and the expected number of object measurements (see (5.20)) is $\lambda_T = 5$.

In the direct scattering model, the standard deviation for the Doppler measurements is set to 0.4375 m/s. For the range density that is depicted in Fig. 5.10, the mixture weights are chosen as $c_1 = 0.4$, $c_2 = 0.55$, $c_3 = 0.05$ for the components centered at the first rectangle intersection, in between the two intersections, and at the second intersection. These values have been chosen empirically. The standard deviation of the range measurements which is used in the first and third mixture component is 0.25 m, whereas that of the second one is chosen in dependence on the intersection length. The transition region for the trapezoidal azimuth density shown in Fig. 5.9 is constructed such that the azimuth density values exceed the uniform azimuth clutter density in a tolerance region of 1° to both sides of the vehicle rectangle.

Both approaches use an identical initialization routine. It clusters all measurements which have not considerably contributed to updating an existing vehicle and which are not covered by an existing vehicle hypothesis with an existence probability above 0.2. The procedure is able to output multiple cluster hypotheses that may contain identical measurements. Thus, several plausible initialization variants can be considered and the filter can then select the best hypothesis after several time steps. The resulting clusters are further analyzed with respect to their size and Doppler characteristics. Only clusters which resemble vehicle measurements are then used to initialize new vehicle hypotheses. In particular, clusters that are too large are discarded. Also, at least 60% of the measurements in a cluster need to exhibit a Doppler velocity that significantly differs from stationary objects. Thus, it is avoided

that many new hypotheses are initialized for static objects. For small clusters with up to two measurements, the threshold for the Doppler velocity is set to 2.5 m/s and it is linearly decreased to 1 m/s for larger clusters of more than five measurements. Furthermore, the Doppler measurements in a cluster are tested for consistent rigid body motion. Using a random sample consensus (RANSAC) type approach [FB81], the algorithm checks whether a solution for the two object state terms $s_{D,1}$ and $s_{D,2}$ in (5.16) can be found with a sufficiently large consensus set.

If a suitable cluster is found, new particles are generated by sampling bounding boxes that enclose the measurements and computing vehicle speeds and yaw rates that match the measured Doppler velocities for the given vehicle poses. It is ensured that yaw rate and speed conform to realistic vehicle motion and vehicles are initialized up to a speed of 200 km/h. If the pose allows for observing the vehicle length, the vehicle length is initialized with suitable values between 2.5 m and 7 m. Otherwise, average vehicle length values between 4 m and 5 m are sampled as such values are close to the length of typical vehicles. Since vehicles directly next to the ego-vehicle are only partly visible to the four sensors, the initialization procedure agglomerates measurements from neighboring sensors for improved initialization performance.

Theoretically, the multi-object likelihood (5.17) requires the computation of all possible measurement partitions and associations. This combinatorial problem is intractable even for a small amount of measurements and a practical implementation therefore needs approximation schemes. In this work, measurement partitions are obtained by applying the density-based spatial clustering of applications with noise (DBSCAN) algorithm [EKSX96] with multiple distance thresholds between 0.5 m and 5 m. Additionally, one partition is generated by assigning all measurements to the closest predicted vehicle and treating the measurements that are not close to any vehicle as clutter measurements. The DBSCAN clusters are again analyzed for consistent rigid body motion. In case there are spurious measurements with deviating Doppler velocities, additional cluster hypotheses without the outliers are generated. This allows the algorithm to generate clusters of vehicle measurements while at the same time excluding measurements from the wheels. Once the measurement partitions are created, the ten best association variants are generated for each partition using Murty's algorithm [Mur68].

Apart from the motion of measured objects, the Doppler velocity is also influenced by the motion of the ego-vehicle itself. Upon reception of radar measurements, this influence is removed by computing the ego-motion portion using the available motion data as well as sensor mounting position and subtracting it from the measured Doppler values.

6.4.2 Single Object Performance

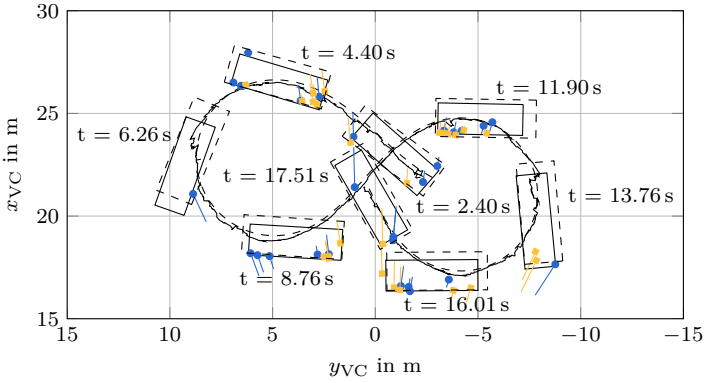
First, the single object performance of radar-based tracking is evaluated using the single object data set. Two exemplary scenarios are examined to illustrate the tracking behavior in different situations with cross traffic, turning vehicles, and parallel traffic that passes all four sensor FOVs. Also, they are used to highlight differences between the two measurement models. The tracking accuracy with respect to the ground truth is studied for both scenarios as well as the entire data set.

In the figure eight scenario, the reference vehicle drives a figure eight and circles in front of the stationary ego-vehicle. This scenario is particularly interesting as the number of radar measurements from the vehicle and the aspect angle constantly change, and the maneuver is highly dynamic with yaw rates up to approximately 60 °/s. Due to the fast turns and the crossing trajectory, the Doppler measurements sometimes strongly vary over the vehicle body and do not directly equal the actual vehicle speed as it is often the case in classical parallel traffic. Therefore, the scenario is especially suitable for demonstrating the performance of the radar measurement models in situations with complex Doppler relationships.

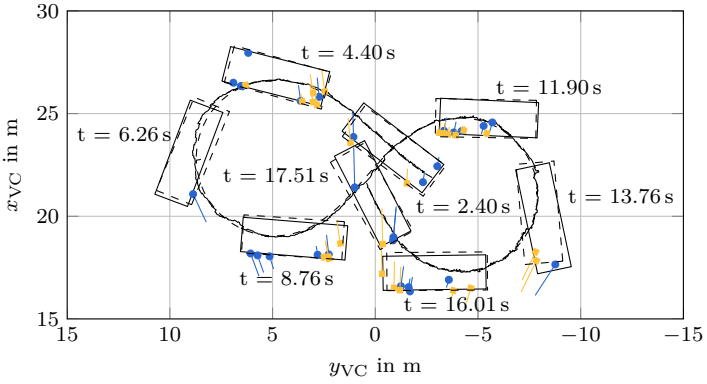
Figure 6.9 depicts excerpts of the scenario and the tracking results for both radar models. The plots compare the estimated trajectories and selected vehicle poses during the first eight in the scenario with the ground truth values. The vehicle is visible to both front sensors and it is tracked continuously using the direct scattering model (cf. Fig. 6.9a) and the variational radar model (cf. Fig. 6.9b). The two models are able to deal with ambiguous situations where only a single measurement is available, e.g. around 6.26 s, and with pronounced Doppler profiles as for instance around 8.76 s. However, a qualitative comparison of both results reveals that the estimated trajectory of the variational radar model is smoother than that of the direct scattering model and the estimated vehicle poses and size match the ground truth rectangles more precisely.

The quantitative tracking accuracy in the figure eight scenario is illustrated in Fig. 6.10 for the direct scattering model and in Fig. 6.11 for the variational radar model. The figures compare the state estimates to ground truth values. Furthermore, the overall RMSEs are listed in Table 6.3. In the figure eight scenario, the variational radar model outperforms the direct scattering model for each state. It especially improves the estimate of the vehicle dimensions, the yaw rate, and the orientation.

In a second scenario, the reference vehicle passes the stationary ego-vehicle. In contrast to the turning motion of the previous experiment, this passing scenario illustrates the performance in a typical parallel traffic situation. As the vehicle drives through the FOVs of all four radar sensors, it furthermore demonstrates how radar



(a) Direct scattering model



(b) Variational radar model. Reprinted and adapted with permission from [SD19] ©2019 IEEE.

Figure 6.9: Excerpts of the figure eight scenario using both radar models: radar measurements from the front left (●) and front right (■) sensors, estimated trajectories (solid) and exemplary vehicle poses (solid rectangles), reference trajectories (dashed) and reference poses (dashed rectangles)

tracking handles FOV transitions and changes in the number of sensors that are able to detect the vehicle. Figure 6.12 illustrates the tracking results for both models. In the mid and far ranges, the sensors only yield one or two measurements which provide little to no geometric information about the vehicle orientation, whereas there are many measurements, including spurious wheel reflections, in the near field. Both approaches are able to continuously track the reference vehicle during the entire maneuver. At the beginning and towards the end of the scenario, however,

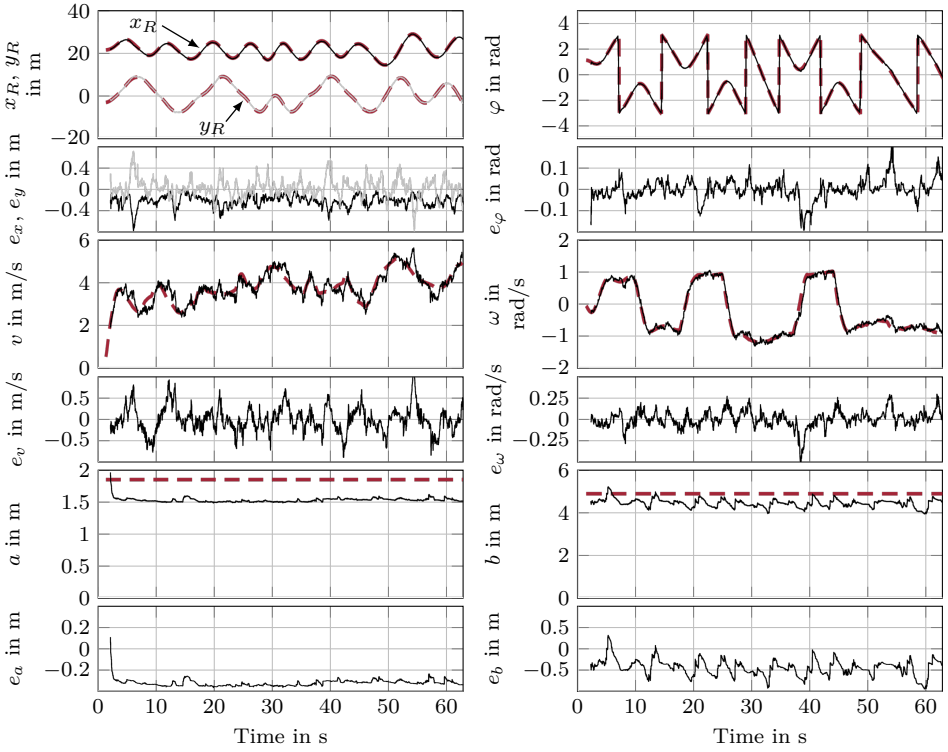


Figure 6.10: Tracking accuracy in the figure eight scenario using radar data and the direct scattering model: Comparison of estimates (solid) and ground truth (dashed red) as well as errors e averaged over 20 runs. The y-position is plotted in gray.

the estimated trajectory of the variational radar model is considerably more precise than the direct scattering estimate. Here, the direct scattering approach exhibits rather large deviations in the y-position as well as the orientation. The RMSE values that are listed in Table 6.3 confirm this observation. While the variational radar model again outperforms the direct scattering model in all state estimates, the improvement is especially large in these two quantities.

A closer analysis of the two measurement models explains this performance gap. Figure 6.13 shows the conditional density $g(d, \alpha|x)$ that follows from the direct scattering model (cf. Section 5.3.2) and indicates likely measurement positions in polar coordinates. The density is constructed for an exemplary vehicle that is viewed by the sensor form a rear perspective. Again, the Doppler values are marginalized

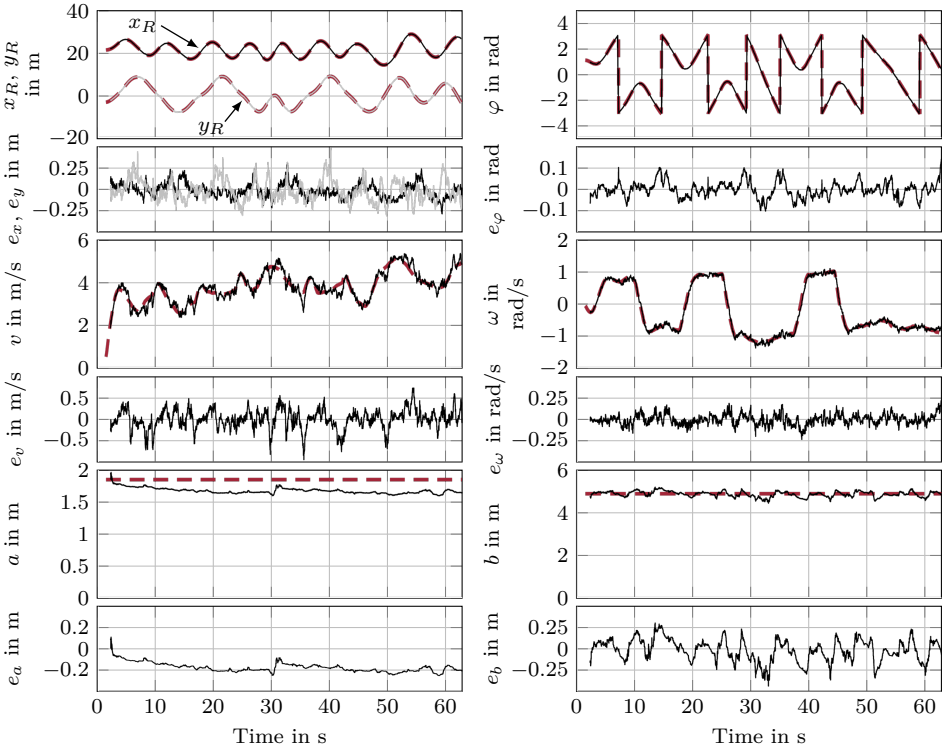
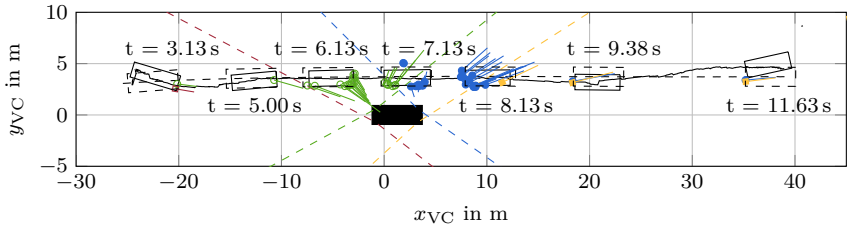


Figure 6.11: Tracking accuracy in the figure eight scenario using radar data and the variational radar model: Comparison of estimates (solid) and ground truth (dashed red) as well as errors e averaged over 20 runs. The y-position is plotted in gray. Reprinted with permission from [SD19] ©2019 IEEE.

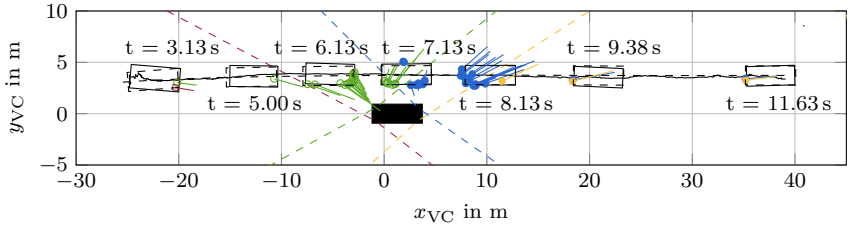
out to simplify visualization. As expected, measurements are most likely on the rear surface of the vehicle. However, the highest likelihood values occur at the rear corners. This is a result of the range density for radar rays that do not target the vehicle directly but pass it. Since the single Gaussian does not extend over a large range interval, it yields larger density values. This behavior is plausible as corners are prominent radar features and are for example explicitly used in reflection point models such as [HSSS12]. Yet, the analysis of the training data set for the variational radar model in Section 6.2 revealed different characteristics for the used sensors. For a rear perspective, they most frequently yield measurements from the center of the rear surface. Consequently, the direct scattering estimates exhibit a bias in such situations.

Table 6.3: Tracking accuracy for the direct scattering and variational radar models: RMSE values for the figure eight and passing scenarios, overall results and the relative change for all single object scenarios. The values are averaged over 20 runs for each scenario.

States	Direct Scattering Model			Variational Model			Change
	Figure eight	Passing	Overall	Figure eight	Passing	Overall	Overall
x_R in m	0.25	0.24	0.31	0.10	0.17	0.27	-12.68%
y_R in m	0.19	0.42	0.40	0.13	0.18	0.26	-33.94%
φ in °	3.60	7.92	9.41	2.29	3.12	7.28	-22.62%
v in m	0.35	0.23	0.55	0.25	0.17	0.36	-33.63%
ω in °/s	6.15	5.36	8.63	3.57	3.60	5.54	-35.83%
a in m	0.33	0.31	0.32	0.19	0.27	0.26	-19.66%
b in m	0.49	0.41	0.50	0.16	0.25	0.28	-42.92%



(a) Direct scattering model



(b) Variational radar model

Figure 6.12: Excerpts of the passing scenario using both radar models: radar measurements from the front left (●), front right (■), rear left (○), and rear right (□) sensors, sensor FOVs (colored dashed lines), ego-vehicle (filled rectangle), estimated trajectories (solid) and exemplary vehicle poses (solid rectangles), reference trajectories (dashed) and reference poses (dashed rectangles)

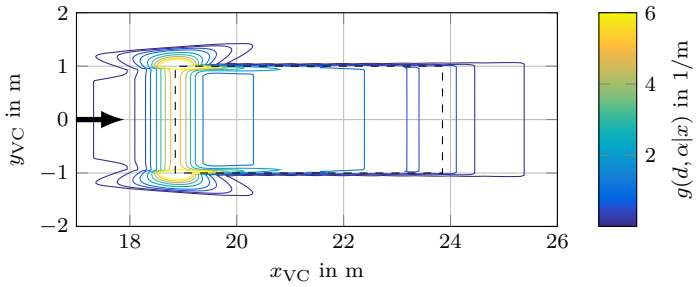
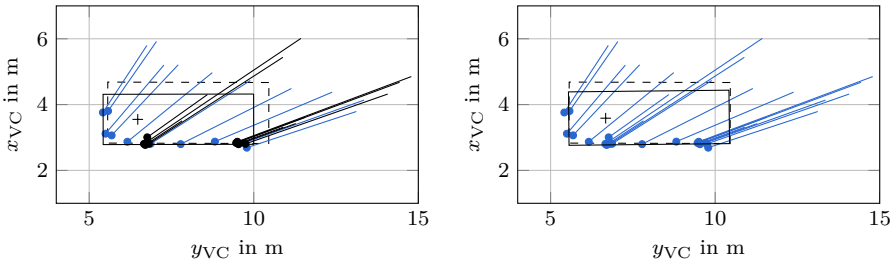


Figure 6.13: Marginalized conditional density $g(d, \alpha|x)$ of the direct scattering model for an exemplary vehicle (dashed rectangle) with state $x = [20\,\mathrm{m}, 0\,\mathrm{m}, 0°, 0\,\mathrm{m/s}, 0\,°/\mathrm{s}, 2\,\mathrm{m}, 5\,\mathrm{m}]$. The line of sight is indicated by the arrow. Note that the density is plotted in Cartesian coordinates despite being defined over polar coordinates.

A possible remedy for the rear perspective would be to alter the azimuth density such that measurements that are close to the surface center are favored. Yet, such a modification would not be valid for side perspectives and additional treatment of other special cases would be necessary. In contrast, the variational radar measurement model has automatically learned a measurement model that matches the actual data and sensor characteristics without the need for elaborate manual tuning.

The passing scenario also demonstrates differences in the measurement clusters that are favored by both radar measurement models to update the tracks. As discussed in Section 6.4.1, the measurement partitioning routine also provides clusters that have been checked for consistent rigid body motion and exclude spurious measurements. To analyze which measurement clusters are used by the two models, Fig. 6.14 depicts the clusters that contributed the most in a particular situation of the passing scenario. Additionally, the average estimated vehicle pose from the 20 Monte Carlo runs is plotted. It can be seen that the front left sensor yields measurements from the rotating wheels with Doppler measurements that are below or exceed those of the vehicle body. When using the direct scattering model, the cluster with the highest likelihood is one that excludes these wheel measurements. In case of the variational radar model, however, the cluster that includes all radar measurements contributes the most. This implies that the variational radar model is able to use all measurements. Particularly the position of the wheel measurements might be one of the major contributions to the improved length estimate that is observable when comparing the estimated vehicle to the ground truth rectangle.

Apart from the RMSE values of the two presented scenarios, Table 6.3 also lists



(a) Direct scattering model (b) Variational radar model

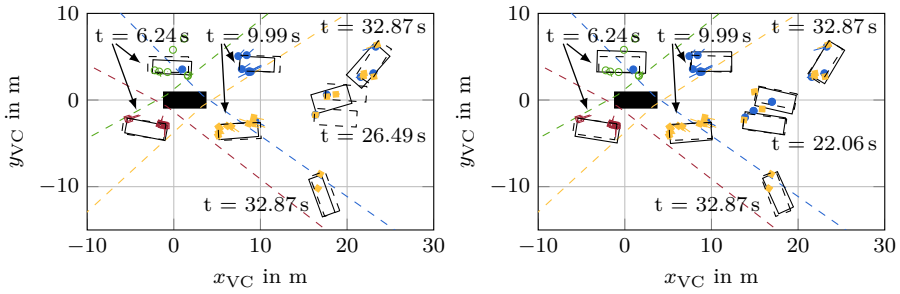
Figure 6.14: Comparison of the measurement clusters that contributed the most during the update step at 7.84 s in the passing scenario: cluster measurements (●), other measurements (●), average vehicle estimate (rectangle), and reference vehicle (dashed rectangle)

the average values for all ten single object scenarios. The variational radar model once more outperforms the direct scattering model with respect to all states. In conclusion, the close approximation of the actual likelihood function that was achieved by learning the model from actual sensor data is better in extracting the relevant information from the measurements than a manually designed model.

6.4.3 Multi-Object Performance

So far, the accuracy was only evaluated for a single vehicle. This section adds an analysis of the multi-object performance of both radar models using the multi-object data set and the real-world scenario. The goal is to analyze the behavior in situations with close-by vehicles, the ability to cope with clutter and to estimate the correct number of vehicles, as well as the applicability to the public road scenario.

The parallel scenario is a particularly challenging scenario as it contains two vehicles that drive very closely to each other. The vehicles first pass the ego-vehicle on both sides. Then, they continue to drive in front of the ego-vehicle at a close distance to each other before turning to the left and right, respectively, at the end of the scenario. Figure 6.15 shows two tracking excerpts using both radar models. Since the lateral distance is down to approximately 1 m when both vehicles are in front of the ego-vehicle, the radar measurements are rather close to each other. As the reference vehicles additionally drive at the approximately same speed, the sensors sometimes have difficulties to resolve the two vehicles and produce merged measurements in between the vehicles, see for example Fig. 6.15b at 22.06 s.



(a) Direct scattering model

(b) Variational radar model. Reprinted with permission from [SD19] ©2019 IEEE.

Figure 6.15: Excerpts of the parallel scenario using both radar models: radar measurements from the front left (●), front right (■), rear left (○), and rear right (□) sensors, sensor FOVs (colored dashed lines), ego-vehicle (filled rectangle), exemplary vehicle poses (solid rectangles), and reference poses (dashed rectangles)

In Fig. 6.15a it is observable that the direct scattering approach is facing difficulties in this scenario. It repeatedly loses tracks in the period where the vehicles are close to each other and even sets up a single track hypothesis for both vehicles at around 26.5 s. A major cause of the track losses are the previously discussed issues of the direct scattering approach to yield a stable angle and position estimate if only measurements from the rear surface are available (cf. Section 6.4.2). If, in addition, uncertainty is large upon reinitialization, the close-by measurements of the second vehicle can cause divergence of the track and lead to a single vehicle hypothesis that uses measurements from both vehicles. In contrast, the variational radar model shows superior accuracy in estimating the correct vehicle poses and is able to continuously track both vehicles throughout the entire scenario.

This difference in performance is also observable in a comparison of the cardinality estimates of both approaches as depicted in Fig. 6.16. The figure shows the expected number of vehicles that the filter outputs averaged over 20 runs as well as the ground truth for each time step. The difficulty of the direct scattering model to keep both tracks appears as an underestimated cardinality between 20 s and 30 s. On the contrary, the estimate of the variational radar model most of the time stays at the correct value of two. The spikes in the cardinality plot that exceed the true value are caused by false tracks which are initialized from clutter measurements. Such false tracks with considerable existence probability appear between 20 s and 25 s using both models and at around 5 s using the direct scattering model. While the

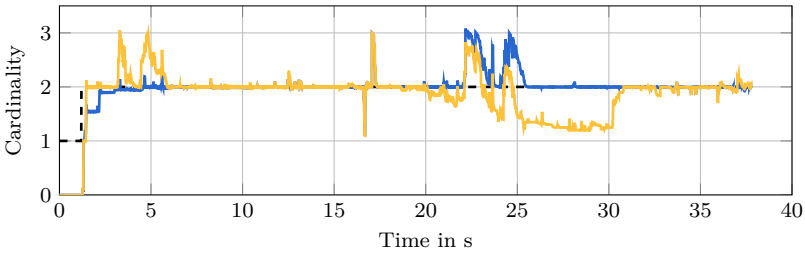


Figure 6.16: Cardinality estimates in the parallel scenario: estimated cardinality using the direct scattering model (—) and the variational model (—) averaged over 20 runs, ground truth (dashed)

variational radar model avoids the clutter tracks at the beginning, it is sometimes slower in initializing both vehicles which leads to a slower rise of the cardinality estimate.

Figure 6.17 illustrates the false track at around 25 s when using the variational radar model as an example. In this case, the track was initialized from clutter measurements with spurious Doppler velocities and it persists as long as other measurements are able to confirm it. Such measurements are available since the track was initialized in an area where multiple measurements from stationary objects occur over several time steps. From a theoretical perspective, the assumption of uniformly and uncorrelated (both in position and time) clutter that is made in the multi-object measurement model from (5.17) to (5.19) is violated in this situation. As a consequence, the filter outputs an erroneous cardinality estimate.

The accuracy of the cardinality estimate was evaluated on the entire multi-object data set to assess the overall multi-object performance of both radar models. In particular, the cardinality estimation errors were computed for all time steps and an error histogram with bin widths of 0.25 was determined. The histogram for both models is shown in Fig. 6.18. When using the direct scattering model, the filter yields the correct cardinality estimate (i.e. a deviation of less than ± 0.125) in 66.8% of the steps whereas the variational variant achieves 73.8%. The cardinality is underestimated in 15.7% of the steps when using the direct scattering model and in 11.0% when using the variational radar model, whereas it is overestimated in 17.5% and 15.2% of the steps, respectively. In summary, the variational radar model particularly reduces the number of steps in which the cardinality is underestimated whereas the improvement in the overestimation case is smaller. This suggests the interpretation that the variational model is especially better at tracking all existing vehicles and that it has a smaller positive effect on the occurrence of false tracks caused by clutter.

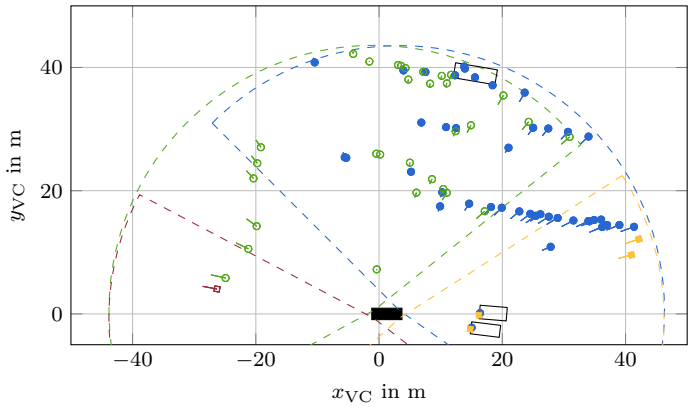


Figure 6.17: Variational radar model: Exemplary clutter track (top) and the two tracked vehicles (bottom) in the parallel scenario at around 25 s with corresponding radar measurements from the front left (●), front right (■), rear left (○), and rear right (□) sensors as well as sensor FOVs (dashed lines)

In the previous evaluation experiments, the radar models were tested on the E-Class and C-Class reference vehicles. As one of the vehicles was used to generate the training data and both vehicle types are rather similar, the T-intersection scenario is used to demonstrate the applicability to a wider range of vehicle types. Two particular situations are illustrated in Fig. 6.19 for both radar models. Figure 6.19a shows an occlusion situation in which two sedans pass the intersection. As soon as the front sedan occludes the other vehicle, both approaches loose the rear sedan as the radar sensors do not yield any measurements for 25 consecutive measurement steps. The track is reinitialized once the vehicle becomes visible again. In contrast, the front sedan is tracked continuously.

In the second situation, a van takes a turn at the intersection while a following convertible and a compact van keep straight on. All three vehicles are tracked continuously with both models. However, the direct scattering model shortly initializes a false track in the vicinity of the turning van, whereas this is avoided by the variational radar model.

The tracking accuracy of this scenario was evaluated for those areas where labels were available, i.e. in the lidar FOV. The RMSE values were computed as average values over 20 runs. The direct scattering model achieves values of 0.41 m and 0.58 m for the x- and y-position, respectively, 4.6° for the orientation, as well as 0.25 m and 0.40 m for the width and length. In contrast the results for the variational radar

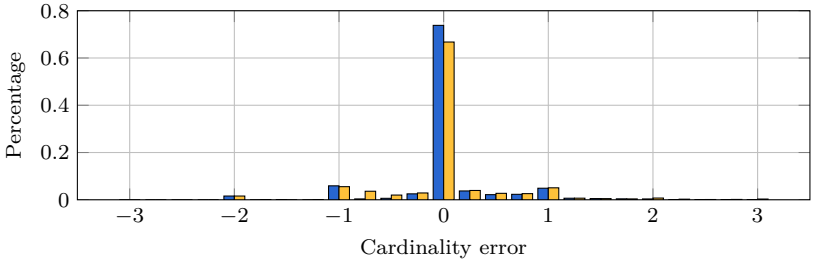
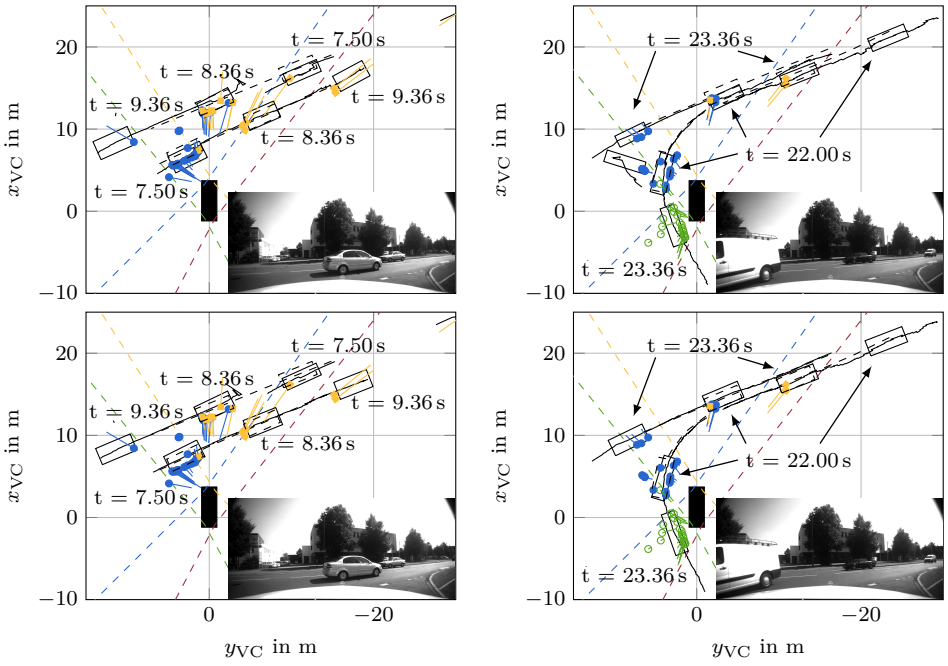


Figure 6.18: Histogram of cardinality estimation errors for the multi-object scenarios: variational radar model (▮) and direct scattering model (▮). For better visualization, the bars are shifted from their actual centers which are multiples of 0.25. Reprinted with permission from [SD19] ©2019 IEEE.

model are 0.21 m and 0.52 m for the position, 4.3° for the orientation, as well as 0.40 m and 0.54 m for the width and length. In this scenario, the performance of both models is rather balanced. The variational radar model is slightly superior in terms of estimating the pose, whereas the direct scattering is better at estimating the vehicle dimensions. The tracking filter outputs a track estimate for the labeled vehicles in 96.3% and 95.1% of the steps for the direct scattering and variational radar model, respectively. The missing steps are mostly due to belated initialization and the loss of the rear sedan in the occlusion situation. From eye inspection, the tracking performance deteriorates in the far range where the number of measurements per object and their accuracy decreases and the number of measurement drop outs and clutter increases.

6.4.4 Discussion

In summary, the evaluation experiments demonstrated several key aspects of the tracking performance when using the direct scattering and the variational radar model. As shown in the figure eight scenario, both models are able to track vehicles in rather complicated maneuvers with varying aspect angles and measurement numbers as well as complicated Doppler relationships. The direct scattering model, however, does exhibit some difficulties in generating stable estimates of the vehicle pose if few measurements are available over a longer period of time. In contrast, the variational radar model outperforms the direct scattering approach in terms of the tracking accuracy and multi-object performance. This shows how a close approximation of the true single object likelihood that is learned from data is superior to a manually designed likelihood function and uses the available information better.



(a) Occlusion situation using the direct scattering (top) and the variational radar (bottom) models

(b) Turning van and two other vehicles tracked by the direct scattering (top) and the variational radar (bottom) models

Figure 6.19: Excerpts of the T-intersection scenario using both radar models: radar measurements from the front left (●), front right (■), rear left (○), and rear right (□) sensors, sensor FOVs (colored dashed lines), ego-vehicle (filled rectangle), estimated trajectories (solid) and exemplary vehicle poses (solid rectangles), reference trajectories (dashed) and reference poses (dashed rectangles). The variational radar model figures are reprinted and adapted with permission from [SD19] ©2019 IEEE.

Moreover, the T-intersection scenario indicated a certain ability of the variational radar model to generalize to other vehicle types. This is not surprising as even high-resolution radar sensors provide a comparatively coarse image of the object. Therefore, there is not as much information about distinctive vehicle features available as in data from other sensor types such as cameras. Yet, problems are expected as soon as there are significant changes in the radar characteristics, e.g. for trucks with additional wheels and other typical scattering centers.

Interestingly, the tracking accuracy of both radar models is similar in the T-intersection scenario. This observation needs to be interpreted carefully as the ground truth was obtained by manual labeling and this single scenario is not on par with the variety of maneuvers in the single and multi-object data sets. It might, however, show that the learned measurement model does not match all vehicle types equally well and that the advantage over the direct scattering model diminishes in such situations. Therefore, studying the effects of a training data set with multiple vehicle types on the tracking accuracy would be an interesting direction for future work and would provide additional insights.

Most methods for radar-based tracking from the literature were developed for other radar sensors with different characteristics and resolution capabilities. As the reported results are moreover obtained for different scenarios, a thorough comparison is difficult. To still allow for a coarse assessment, the tracking results are compared to the tracking approach from [KBK+16] which uses a state of the art Doppler profile preprocessing routine. It lists RMSE values of 5.20° for the orientation, 0.37 m/s for the velocity, 3.71 °/s for the yaw rate, and 0.87 m for the Euclidean position error when tracking with a state-of-the-art Doppler profile preprocessing routine with two synchronized sensors. These results were obtained for a figure eight scenario similar to that of this thesis using sensors with slightly better resolution. The comparison with this similar scenario indicates an improved accuracy of the direct scattering and variational radar models. Apart from the yaw rate estimate of the direct scattering model, the RMSE values are lower. In contrast, [KBK+16] is able to report real-time capability. In terms of extended object models, a comparison with the recently published volcanormal density [BDD17] on an identical data basis would be interesting. The model is not used in a Bayesian tracking filter but in a maximum likelihood estimator that optimizes over multiple time steps and reports competitive accuracy for two scenarios.

One of the remaining challenges in the multi-object performance is to avoid false tracks caused by correlated clutter measurements. This correlation is not considered by the clutter model which assumes all clutter measurements to be independent, uncorrelated over time, and uniformly distributed. Therefore, the filter is sometimes overly confident in the existence probabilities and the deduced cardinality estimate. Finding more appropriate clutter models or concurrently estimating the clutter density could alleviate this issue.

Due to the superior performance, the following sensor fusion experiments use the variational radar model for the radar sensors unless otherwise stated.

6.5 Evaluation of Lidar Tracking

The separable likelihood approach for lidar-based tracking is evaluated as a second constellation with a single sensor type. Section 6.5.1 first reviews some lidar specific implementation issues and the evaluation results for both the single and multi-object performance are then presented in Section 6.5.2. For the single object scenarios, the focus of the performance evaluation again lies on the accuracy in comparison to the ground truth. The multi-object evaluation then analyzes the cardinality estimates as well as the effects of the proposed occlusion model on the performance.

6.5.1 Lidar-Specific Implementation Details

To obtain the planar lidar scan that the ray-based model from Section 5.1 is designed for, the four layers of the ego-vehicle's lidar sensor are reduced to an artificial single-layer scan. In this procedure, the two lower layers are discarded and only the two upper layers which provide a considerably larger range are used. The azimuth pattern of the artificial scan is set to that of the two upper layers and the measured range is set to that of the closest measurement from the two upper layers. As the measurement density and quality decreases with distance, the lidar range is limited to 100 m and measurements above this maximum range are not considered.

In the vehicle template from Fig. 5.4, the surface thickness is set to 0.45 m and the free space exceeds the outer surface rectangle by 0.7 m in all directions. To parameterize the range likelihood densities of the three ray types (cf. Fig. 5.5), likelihood factors which define the desired ratio between the uniform clutter range density and the density value for the particular section are introduced. These factors are specified for all but one section (typically the vehicle surface) and multiplied with the uniform density value. The density value of the remaining section is obtained from normalization. Table 6.4 lists the used factors for the three ray types along with the probabilities of receiving a return. As described in Section 5.1.2, these probabilities are linearly decreased to the listed minimum values starting from 70 m.

Since there is no velocity information in a single lidar scan, the lidar module uses a multi-step routine for initializing moving vehicles. In each step, the lidar points are clustered using a modified version of the adaptive break point detector from [BA04]. Clusters in which more than half of the measurements have already contributed to updating an existing vehicle as well as clusters that are too large or not sufficiently convex are discarded. The remaining clusters are associated over consecutive time steps using a nearest neighbor approach to obtain an estimate of the velocity vector.

Table 6.4: Parameterization of the lidar ray likelihoods: return probabilities and likelihood factors for the ray types from Figs. 5.4 and 5.5

Type-1 Ray		Type-2 Ray		Type-3 Ray	
Factor	Value	Factor	Value	Factor	Value
$p_{\mathrm{R},1}$	0.5	$p_{\mathrm{R},2}$	0.6	$p_{\mathrm{R},3}$	0.65
		$p_{\mathrm{R},2,\min}$	0.52	$p_{\mathrm{R},3,\min}$	0.55
$\overline{z_{\mathrm{FS}} z_{\mathrm{BF}}}$	0.001	$\overline{0 z_{\mathrm{FS}}}$	0.6	$\overline{0 z_{\mathrm{FS}}}$	0.6
		$\overline{z_{\mathrm{FS}} z_{\mathrm{S}}}$ & $\overline{z_{\mathrm{BS}} z_{\mathrm{BF}}}$	0.001	$\overline{z_{\mathrm{FS}} z_{\mathrm{S}}}$	0.001
		$\overline{z_{\mathrm{BF}} z_{\max}}$	0.5	$\overline{z_{\mathrm{BS}} z_{\max}}$	0.001

If clusters from three time steps could be associated and the two velocity estimates are consistent, a new vehicle hypothesis is initialized. Using three steps instead of the necessary minimum of two steps helps to further decrease the number of false tracks caused by sensor jitter.

For generating particles, bounding boxes which enclose the measurements are sampled similarly to the radar initialization routine. Afterwards, the best bounding boxes are selected by evaluating the single-object likelihood. In contrast to radar, there is no precise motion information and the speed is therefore sampled in a broader range around the estimated velocity vector. Also, the initial yaw rate is sampled from a zero-mean Gaussian distribution with a standard deviation of 0.5 rad/s.

As lidar provides especially dense geometric information, the size hypotheses generation of the lidar sensor module allows for changes in the vehicle length that exceed 0.3 m. Thus, the length estimate can be adapted faster.

6.5.2 Tracking Performance

The single object accuracy is again evaluated on the single object data set. The resulting RMSE values for the figure eight and passing scenario as well as the complete data set are listed in Table 6.5. Also, Fig. 6.20 shows plots of the state estimates and errors in detail for the figure eight scenario. Note that the results are not directly comparable to those of the radar models in Section 6.4.2 as the sensor FOVs and thus the time instants in which the vehicle is visible differ. This is particularly true for the passing scenario, where the radar sensors are able to perceive the vehicle for a considerably longer time. Yet, some (expected) trends are observable: While the lidar model yields better orientation estimates due to the rich geometric information, the speed and yaw rate estimation errors are larger

Table 6.5: Accuracy of lidar-based tracking: RMSE values for the figure eight and passing scenarios, overall results for all single object scenarios. The values are averaged over 20 runs for each scenario.

States	Figure eight	Passing	Overall
x_R in m	0.18	0.17	0.21
y_R in m	0.18	0.09	0.19
φ in °	1.66	1.70	2.38
v in m	0.84	0.69	0.93
ω in °/s	10.37	6.90	11.30
a in m	0.16	0.25	0.19
b in m	0.17	0.60	0.32

than those of the radar models. A more detailed comparison of the sensor-specific performance is provided in the following section on fusion of radar and lidar data.

The effect of the lidar occlusion model from Section 5.1.1 on the multi-object performance is demonstrated using two scenarios from the multi-object data set. In the occlusion scenario, both reference vehicles cross in front of the stationary ego-vehicle. As soon as they are directly in front of the ego-vehicle, the front reference vehicle fully occludes the rear vehicle. The tracking results for the first crossing are illustrated in Fig. 6.21 without (Fig. 6.21a) and with (Fig. 6.21b) the occlusion model. Even though the individual rays consider the case of measurements in front of the vehicle, the rear track is lost in occlusion when not using the full multi-object occlusion model. The fact that all measurements are in in front of the surface is counted as strong evidence against the existence of the object. In contrast, the rear vehicle is continuously tracked when enabling the occlusion model that only uses the rays that are able to target the vehicle during update. In this scenario, the occlusion model prevents a cardinality estimation error that is otherwise made due to a violation of the separability assumption.

The occlusion scenario also demonstrates that the ray-based approach handles the partial visibility of vehicles at the FOV boundaries naturally and automatically uses the remaining rays without the need for special case treatments.

In the second scenario, the ego-vehicle follows two other vehicles that first enter the lidar FOV from both sides and then drive in a row. When both vehicles are in front of the ego-vehicle, the front vehicle is occluded for 22 update steps or about two seconds before becoming visible again. Figure 6.22a illustrates this situation and the estimation results. Note that the plotted trajectories do not reflect the actual positions over ground since the trajectories are estimated in vehicle coordinates and

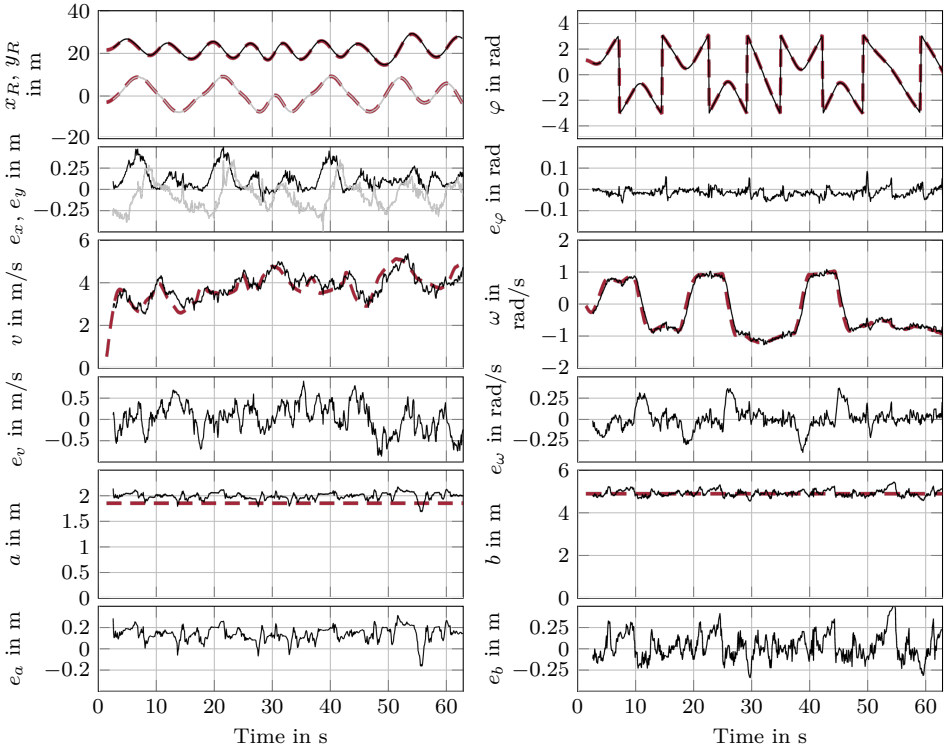
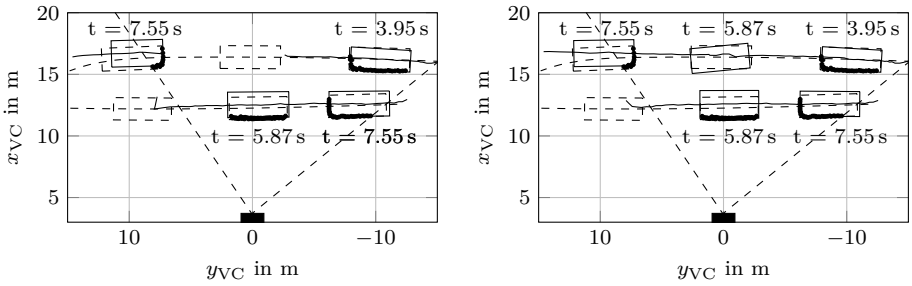


Figure 6.20: Tracking accuracy in the figure eight scenario using lidar and the separable likelihood model: Comparison of estimates (solid) and ground truth (dashed red) as well as errors e averaged over 20 runs. The y-position is plotted in gray.

the ego-vehicle is moving. The estimated trajectory of the front vehicle shows a strong deviation from the ground truth between the two plotted vehicle poses at 19.92 s and 22.32 s. This deviation occurs during the occlusion situation and causes a track loss in most runs.

In this situation, the occlusion model enables keeping the occluded track for several update steps without any measurements. Yet, the long duration causes a rapid increase in the state uncertainty. This is further assisted by the rather large process noise which is parameterized for highly dynamic maneuvers. Due to their relatively low number, the particles are not able to sufficiently cover the possible state space and, hence, no suitable particle is available in most runs as soon as new measurements occur. This causes the track loss in most runs. Therefore, an increase in the particle



(a) Without occlusion model

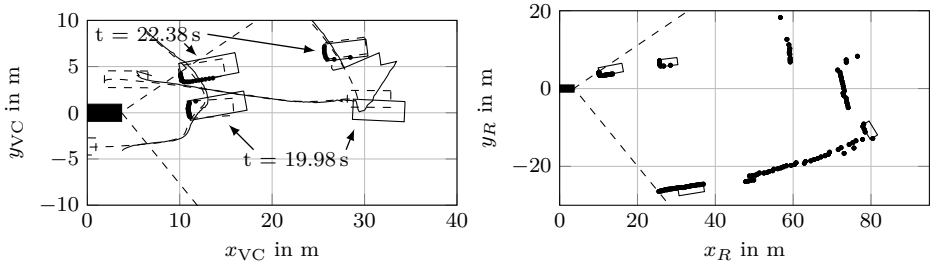
(b) With occlusion model

Figure 6.21: Effect of the lidar occlusion model in a short occlusion situation: lidar measurements (•), sensor FOV (straight dashed lines), ego-vehicle (filled rectangle), estimated trajectories (solid) and exemplary vehicle poses (solid rectangles), reference trajectories (dashed) and reference poses (dashed rectangles)

number is expected to yield improved performance. Another possible remedy would be to reduce process noise of occluded vehicles and to assume a rather monotonous motion.

Figure 6.22b shows a second perspective on the scenario and illustrates another challenge for lidar-only tracking. While both vehicles are tracked, the filter also outputs two clutter tracks that originate form measurements of stationary objects. As in the multi-object model of the radar model, this is caused by the simple clutter model which assumes uniform, independent, and uncorrelated range values for clutter measurements. In practice, however, clutter measurements are heavily correlated both within one scan as well as over consecutive scans. As a consequence, clutter measurements from a structured environment fit oftentimes better to the vehicle than to the clutter model.

The cardinality errors with and without the clutter model are plotted in Fig. 6.23 as histograms over the entire multi-object data set. The percentage of correct cardinality estimates is lower as in the radar results. In contrast to lidar, radar sensors provide Doppler measurements as additional information to distinguish between moving objects and stationary clutter. Also, it is observable that the occlusion model slightly reduces the cases where the cardinality is underestimated but it increases the cases of overestimation. Here, the occlusion model not only allows keeping true positive vehicle tracks but also false tracks in occlusion situations.



(a) Tracking in a long occlusion situation (b) Excerpt of the full scene at t = 22.32 s

Figure 6.22: Lidar tracking in a following scenario with two vehicles: lidar measurements (●), sensor FOV (straight dashed lines), ego-vehicle (filled rectangle), estimated trajectories (solid) and exemplary vehicle poses (solid rectangles), reference trajectories (dashed) and reference poses (dashed rectangles)

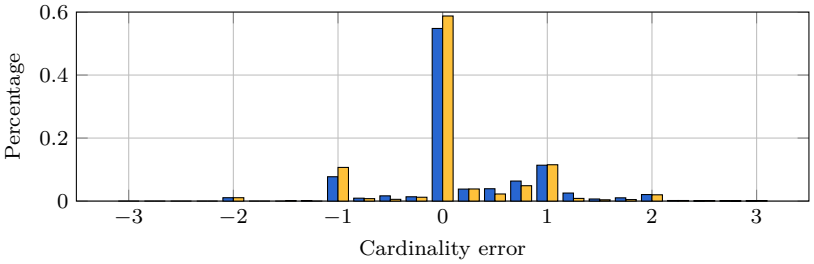


Figure 6.23: Histogram of cardinality estimation errors for lidar-based tracking: with occlusion model (▮) and without occlusion model (▯). For better visualization, the bars are shifted from their actual centers which are multiples of 0.25.

6.5.3 Discussion

The lidar evaluation has demonstrated that the ray-based lidar model is able to produce accurate results in the vehicle pose whereas it faces more difficulties than the radar models in estimating the vehicle motion and in distinguishing clutter measurements from vehicles. The lidar measurement model accurately reproduces the lidar measurement principle and therefore avoids—apart from track initialization—clustering routines and finding thresholds for segmentation. Also, it allows handling aspect angle changes such as in the figure eight scenario, partly visible tracks at the FOV boundaries, and occlusion situations naturally. Thus, the model circum-

vents special case treatments which are oftentimes necessary in classical box fitting approaches. The proposed occlusion model helps to maintain tracks in occlusion situations but the availability of reasonable estimates in occlusion is limited by the duration and increasing uncertainty. Keeping occluded vehicles over many update steps seems questionable if no measurements are available. Yet, the following section will demonstrate the usefulness of the occlusion model when fusing information from other sensors.

While the cardinality estimates are correct in most of the time steps, there is a certain susceptibility to cardinality errors caused by false tracks. As in the radar models, this is due to the clutter density which neglects possible correlation among real clutter measurements. Therefore, work on more precise clutter models or approaches that concurrently estimate the clutter density could lead to further improvements.

6.6 Evaluation of Sensor Fusion

After having evaluated the performance of radar and lidar-based tracking for set-ups with a single sensor type, this section investigates the benefits and performance of sensor fusion. Different sensor constellations which also include the scene labeling measurement model are evaluated and compared. To achieve a fair evaluation basis, only the area that is covered by all three sensor types is used for evaluation. The vehicle accuracy is moreover only evaluated if the reference vehicle is fully in this area. Thus, estimates are only compared if all sensors were able to fully see the vehicle and the radar sensors' advantage of a larger FOV is diminished.

In the following, Section 6.6.1 first details some scene labeling implementation details before Sections 6.6.2 to 6.6.4 present results for the fusion of radar and lidar, radar and scene labeling, as well as the fusion of all three sensor types.

6.6.1 Scene-Labeling-Specific Implementation Details

The labeled images are obtained from processing the ego-vehicle's wide angle camera images using the scene labeling approach presented in [TG17]. It uses a convolutional neural network (CNN) architecture to classify the pixels of the input image into the classes vehicle, person, road, and background. The original input image is downscaled by a factor of 0.25 which yields the input size of 480x270 pixels. As the CNN only outputs class labels for those pixels where the receptive field is fully in image, the resulting labeled image that is used as measurement has a resolution of

383x171 pixels. Each pixel is assigned the class with the highest score in the final soft-max classification layer of the network.

In the scene labeling sensor module, the labeled image is then undistorted and the classification is transformed to a two-class problem by retaining the vehicle class and merging the classification as person, road, and background to a single background class. Additionally, irrelevant pixels such as those that lie on the ego-vehicle's engine hood are masked and not considered during update.

In contrast to the other two sensor types, the scene labeling module is not allowed to initialize new tracks and is merely used to update vehicle tracks that have been initialized by the other sensors. While principally possible, this avoids the implementation of computer vision algorithms for detecting initial vehicle guesses and handling the uncertainty in distance that exists in mono camera images.

As previously mentioned in Section 5.2, the parameters of the Bernoulli distributions that are used as foreground and background likelihood functions for each pixel need to be chosen carefully. Contrary to the independence assumption of the separable likelihood model, the classification results of the individual pixels are not independent due to the CNN architecture. If the scene labeling fails, it oftentimes misclassifies larger connect pixel patches. Also, it was observed that the particular parameterization of the scene labeling approach tends to erroneous classifications of vehicles if they are close to the edges of the labeled image. Therefore, the Bernoulli parameters are set to $r_\varphi = 0.51$ for foreground and $r_\psi = 0.5$ for background pixels. Moreover, the foreground parameter r_φ is linearly decreased to 0.5 in the two outer thirds of the image plane towards the left and right image border. The small likelihood difference for a single pixel is amplified by multiplying the many pixels that a vehicle typically covers.

6.6.2 Fusion of Radar and Lidar

In a first step, radar data is fused with lidar data using the variational radar model and the ray-based lidar model. The evaluation analyzes the effects on fusing both sensors on the tracking accuracy, the benefits of the lidar occlusion model, and the ability to cope with sensor failure.

The accuracy results for the single object data set are listed in Table 6.6 as average values of 20 Monte Carlo runs and compared to the performance when only using radar and lidar. The relative change of the RMSE values with respect to the best and worst error of the single sensor results are also specified. The fusion result is always better than the worst estimate from both individual sensor results and

Table 6.6: Comparison of average RMSE values over 20 Monte Carlo runs when fusing radar and lidar data

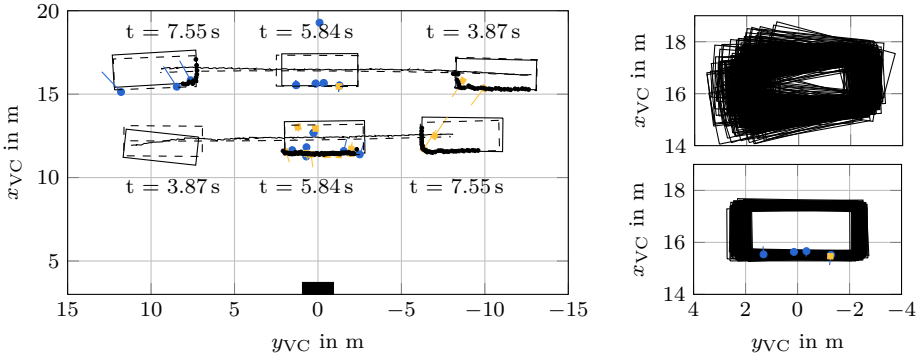
States	Variational Radar Model	Lidar Model	Fusion	Comparison to best	Comparison to worst
x_R in m	0.15	0.16	0.15	-3.60%	-7.75%
y_R in m	0.23	0.18	0.15	-15.72%	-33.52%
φ in °	6.55	1.69	3.25	93.00%	-50.29%
v in m	0.35	0.87	0.37	6.60%	-57.83%
ω in °/s	5.54	11.43	5.01	-9.60%	-56.20%
a in m	0.20	0.17	0.16	-5.77%	-23.51%
b in m	0.22	0.26	0.23	3.43%	-12.27%

considerably improves these values. This demonstrates that fusion particularly helps to cancel the weaknesses of the respective sensors. For example, the relatively large error in the yaw rate estimate of lidar-based tracking is reduced by more than fifty percent.

Also, the fusion results are better than the best estimates of a single sensor, except for the orientation, speed, and length. While fusion still improves the orientation estimate in comparison to radar-only tracking, it does not achieve the accuracy of lidar tracking. These deviations are due to a particular scenario from the data set in which the reference vehicle drives a figure eight with yaw rates above those of the previously discussed figure eight scenario. During the turns, the assumption of the motion center being located at the center of the rear axle is not valid any more. The tracking algorithm is able to compensate this shift of the center of motion when using a single sensor type only. Yet, the combination of a precise orientation estimate along with the Doppler measurements presumably counteracts the compensation and causes track losses in some turns. Orientation uncertainty upon reinitialization then causes increased orientation errors. Without this extremely dynamic scenario, the fusion results would also outperform the single sensor type constellations with respect to the orientation and speed.

The percentage of true cardinality estimates (i.e. the cardinality does not deviate more than 0.125 from the true value) in the fusion evaluation area is 90.5% for lidar-only and 95.1% for radar-only tracking[2]. When fusing radar and lidar measurements, the percentage of true cardinality estimates is 95.6% and achieves the performance of radar-only tracking.

[2] As there are few stationary clutter objects in the area in front of the ego vehicle, the cardinality estimates are considerably better than when considering the entire sensor FOVs.



(a) Scenario excerpt: ego-vehicle (filled rectangle), estimated trajectories (solid) and exemplary vehicle poses (solid rectangles), reference trajectories (dashed) and reference poses (dashed rectangles)

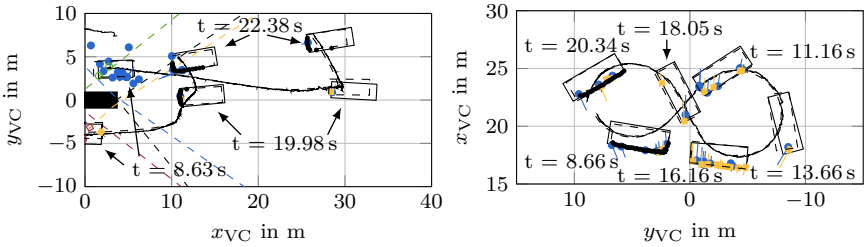
(b) Particle distributions: lidar (top), fusion (bottom)

Figure 6.24: Fusion of radar and lidar data in the occlusion scenario: lidar measurements (●), radar measurements from the front left (●), front right (■), rear left (○), and rear right (□) sensors

To analyze the effects of sensor fusion in occlusion situations, the two multi-object scenarios from the lidar evaluation are revisited. Figure 6.24 shows the estimation results in the occlusion scenario. Again, both vehicles are continuously tracked through occlusion, similarly to lidar-only tracking with activated occlusion model. However, a comparison of the occluded vehicle's particle distribution (see Fig. 6.24b) shows that the state uncertainty is considerably reduced and that the available radar measurements ensure precise tracking also in the absence of lidar measurements.

Furthermore, fusion of both sensors ensures that the front vehicle in the following scenario is continuously tracked through the occlusion period (see Fig. 6.25a) and does not deviate as in lidar-only tracking (cf. Fig. 6.22a).

To evaluate the behavior in the case of sensor failure, i.e. in the case a sensor stops delivering measurements, the lidar sensor was switched off in the figure eight scenario between 8.6 s and 20.3 s. The results are illustrated in Fig. 6.25b. As soon as the lidar measurements are missing, the filter keeps track of the vehicle only using the available radar measurements. After the lidar sensor is switched on again, it seamlessly continues using the measurements of both sensor types. This behavior is possible as the chosen centralized fusion architecture avoids elements of static fusion which require all sensors to be available at all times.



(a) Following scenario with two vehicles (b) Figure eight scenario with lidar failure

Figure 6.25: Two scenarios that illustrate fusion of radar and lidar: lidar measurements (●), radar measurements from the front left (●), front right (■), rear left (○), and rear right (□) sensors, sensor FOVs (dashed lines), ego-vehicle (filled rectangle), estimated trajectories (solid) and exemplary vehicle poses (solid rectangles), reference trajectories (dashed) and reference poses (dashed rectangles)

6.6.3 Fusion of Radar and Scene Labeling

Apart from lidar, camera images also provide rich information about the object geometry and orientation and thus constitute an interesting complement to radar data. Therefore, fusion of radar with scene labeling data is evaluated as a second sensor constellation in this section. The focus is again set on analyzing the effects on the tracking accuracy. Additionally, it is demonstrated how the occlusion model in the scene labeling module allows for processing large label patches despite the missing instance-level information.

The RMSE values for the single object data set are listed in Table 6.7. Again, sensor fusion has a positive impact on the tracking accuracy. The estimation errors are considerably reduced throughout the state vector. In particular, there is a notable improvement in the orientation estimates due to the additional information. The effect of sensor fusion on the percentage of correct cardinality estimates in the multi-object data set is rather small and remains with 95.3% at the same level as radar-only tracking (95.1%).

The fusion performance of radar and scene labeling is additionally illustrated in Fig. 6.26 for the public road scenario. Figure 6.26a shows the occlusion situation that causes a track loss of the rear vehicle when only using radar information. In contrast, the rear vehicle is continuously tracked through occlusion when using the

Table 6.7: Comparison of average RMSE values over 20 Monte Carlo runs when fusing radar and scene labeling data

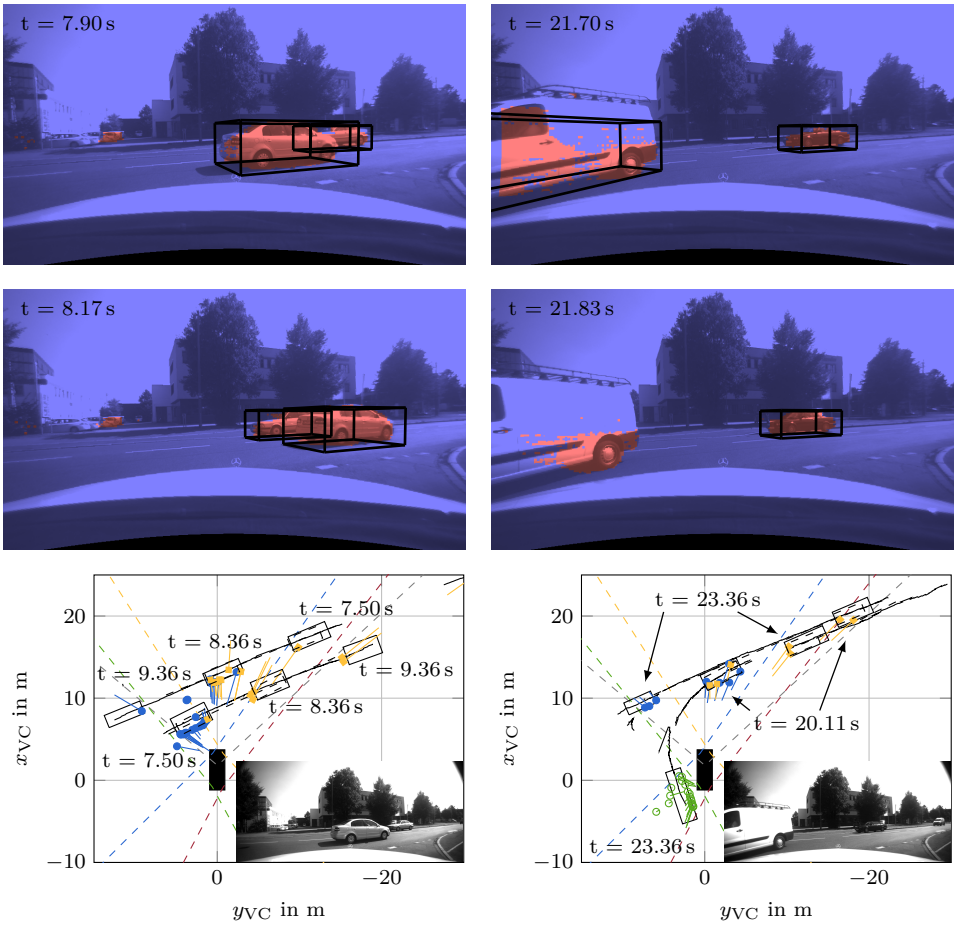
States	Variational Radar Model	Fusion	Relative Change
x_R in m	0.15	0.13	-17.69%
y_R in m	0.23	0.15	-33.95%
φ in °	6.55	4.00	-38.93%
v in m	0.35	0.30	-13.26%
ω in °/s	5.54	4.96	-10.48%
a in m	0.20	0.20	-0.87%
b in m	0.22	0.18	-18.77%

fusion approach. This is due to the prolonged availability of measurements from the rear vehicle in the labeled image and the occlusion model that was added to the separable likelihood model. The occlusion model allows the scene labeling module to internally split the patch of vehicle labels, which appears as one blob without any instance information, and to assign the pixels to the respective vehicles.

An example of large a scale scene labeling failure is depicted in Fig. 6.26b. The filter discards the vehicle track since a majority of the pixels showing the turning van is not correctly classified over several time steps. As shown in the tracking excerpt figure, the track is shortly after reinitialized by the radar sensors.

The overall tracking RMSE values with respect to the labeled ground truth is 0.23 m and 0.48 m for the position, 4.2° for the orientation, and 0.39 m as well as 0.54 m for the width and length. State estimates for the vehicles are available in the lidar FOV in 96.0% of the cases. Here, the major cause for this slight improvement in comparison to tracking with the variational radar model only is the avoidance of the track loss during the occlusion situation.

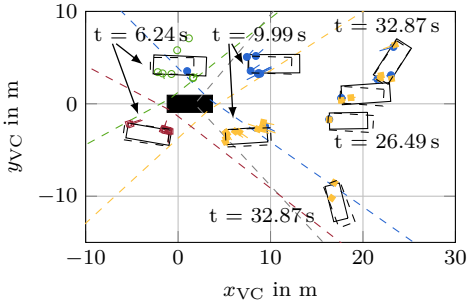
In a final experiment, the large impact that fusion of scene labeling information can have on the tracking accuracy is shown for the parallel scenario. Since the direct scattering model exhibited great difficulties in maintaining continuous tracks in this scenario (cf. Fig. 6.15a), it was used instead of the variational radar model. The results and an exemplary labeled image are illustrated in Fig. 6.27. Due to the information from the scene labeling data, the filter is able to clearly distinguish the two vehicles and achieves improved orientation estimates. Consequently, it continuously tracks both vehicles.



(a) Occlusion situation: Labeled images before and after occlusion (top) and tracking excerpt (bottom)

(b) Turning van and two other vehicles: large-scale labeling failures (top) cause a track loss

Figure 6.26: Radar and scene labeling fusion in the T-intersection scenario: front left (●), front right (■), rear left (○), and rear right (□) radar measurements, labeled images with vehicle (red) and background (blue) classes, sensor FOVs (colored dashed lines), ego-vehicle (filled rectangle), estimated trajectories (solid) and exemplary vehicle poses (solid rectangles / boxes), reference trajectories (dashed) and reference poses (dashed rectangles)



(a) Tracking excerpt with radar measurements from the front left (●), front right (■), rear left (○), and rear right (□) sensors, ego-vehicle (filled rectangle), exemplary vehicle poses (solid rectangles), and reference poses (dashed rectangles)

(b) Labeled image with vehicle (red) and background (blue) classes, as well as exemplary vehicle poses (boxes)

Figure 6.27: Parallel scenario when using scene labeling data and the direct scattering model

6.6.4 Fusion of Radar, Lidar, and Scene Labeling

In a final experiment, the data from all three sensor types is fused. Table 6.8 lists the RMSE values for the single object data set. The comparison with the results of radar and lidar as well as radar and scene labeling fusion shows that the use of all three sensor types is able to reduce the weaknesses of the two-sensor constellations. Apart from the x-position and the length, which only slightly deteriorate, all RMSE values are better than the worst estimate from the two comparison set-ups. Yet, the achieved results do for most states not reach the accuracy of the best value.

From a theoretical perspective, one would expect that data fusion always improves the accuracy. However, this supposes that all measurement and process models exactly represent the true densities and that inference is made in an exact and not approximate way. Thus, there are several issues that may cause the fact that the combination of three sensors does not clearly outperform the constellations with two sensors. Particularly the handcrafted measurement models may, for instance, contain modeling errors or information may be lost in the resampling step of the particle filter. Also, the use of all three sensor types results in an average of 67.5 measurement cycles per second in the area in front of the ego-vehicle. Therefore, the information gain of the temporally closely spaced measurements might be small in comparison to modeling errors. Scene labeling and lidar data furthermore exhibit similar strengths

Table 6.8: Comparison of average RMSE values over 20 Monte Carlo runs of the different fusion constellations

States	Radar + Lidar	Radar + Scene Labeling	All Sensors	Comparison to best	Comparison to worst
x_R in m	0.15	0.13	0.15	20.22%	2.65%
y_R in m	0.15	0.15	0.15	-2.61%	-3.24%
φ in °	3.25	4.00	2.33	-28.36%	-41.69%
v in m	0.37	0.30	0.35	16.73%	-5.02%
ω in °/s	5.01	4.96	4.85	-2.09%	-3.04%
a in m	0.16	0.20	0.16	1.46%	-21.71%
b in m	0.23	0.18	0.23	28.06%	0.57%

in comparison to radar data. They both provide valuable information about the orientation and size of objects. If used together, the contribution might not be significantly greater than in the individual cases. Nonetheless, the combination of all three sensor types increases the redundancy of the perception system and provides greater protection against sensor failure.

Lastly, fusing radar, lidar, and scene labeling data yields the best value for the correct cardinality estimate: The cardinality is correct in 95.9% of the time steps, as opposed to 95.6% and 95.3% for radar and lidar as well as radar and scene labeling fusion, respectively.

6.6.5 Discussion

In summary, the fusion evaluation experiments allow to draw several conclusions: Fusing radar and lidar combines the strengths of both sensors and particularly helps to merge the precision of the lidar in terms of the pose estimate with the accurate motion information from the radar sensors. Also, the lidar occlusion model proves to be valuable in such a fusion approach. It naturally models the absence of information in the lidar data and permits the filter to focus on the remaining radar information.

Similar to lidar and radar fusion, the use of a rather simple scene labeling model is able to greatly improve the tracking accuracy in comparison to radar-only tracking. Again, the occlusion model allows to keep track of vehicles that are occluded in the camera image and to separate the instance-free semantic information.

It was shown that fusion of data from three different sensor types and using extended

object models for all sensors is possible. The detailed extended object models closely emulate the measurement principles. They allow for proper consideration of sensor-specific characteristics, such as occlusion in lidar and scene labeling data, as well as the measurement uncertainty. In sensor fusion, it is particularly important that all measurement models reflect the true measurement uncertainty as good as possible. This avoids the introduction of overconfidence in the state estimates through particular sensor modules. Hence, the extension of the data-driven approach of the variational radar model to the other sensor types could further improve the performance.

Finally, it was demonstrated that the proposed system architecture is able to cope with sensor failures as it uses a centralized fusion architecture and does not rely on static fusion elements.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

The goal of this thesis was to derive a vehicle tracking framework that tackles the multi-object problem, the extended object problem, and sensor fusion in an integral probabilistic manner and directly operates on the sensors' raw measurements. As solution, the thesis proposed a framework that is based on finite set statistics (FISST) and provides a Bayesian end-to-end formulation for the entire problem. The multi-object state is described in form of a random finite set (RFS) and the framework aims at obtaining a posterior density of this multi-object state by applying the multi-object Bayes filter. Its equations are solved by modeling the multi-object densities as labeled multi-Bernoulli (LMB) and generalized labeled multi-Bernoulli (GLMB) distributions and using an LMB-filter-like structure. The FISST-based formulation not only provides a solution to the multi-object problem but also allows for straightforward incorporation of extended object models, which are able to process multiple measurements, closely model the sensor behavior, and aim at using all available information.

There are several contributions that this work makes to the current state of the art. First, it proposed a filter architecture that tackles the three mentioned problems in a fully Bayesian fashion and allows for easy adding and removing of sensor modules that accommodate entirely different measurement models and update routines. The filter cycle additionally includes a second prediction step that was theoretically derived and approximately avoids overlapping objects.

Secondly, several new extended object models were developed for vehicle tracking using radar, lidar, and scene labeling data. In particular, two extended object models for radar-based tracking were put forward in form of the direct scattering and variational radar models. The direct scattering model is based on expert knowledge,

whereas the variational radar model is the first extended object model for radar data that is directly learned from actual measurements. Moreover, the separable likelihood approach was transfered to tracking with lidar data. Thus, a new lidar measurement model was established and it was shown how ray-based lidar measurement models can be incorporated into FISST-based multi-object trackers. Lastly, an extended object measurement model for scene labeling data was developed. It considers the occlusion of objects and is able to cope with vision data that does not provide instance-level information.

Thirdly, this thesis presented the first approach that fuses data from three heterogeneous sensor types using different sensor-specific extended object models and it was demonstrated that the filter can be implemented using particle filtering techniques.

To assess the performance of the proposed methods, they were evaluated on multiple single object and multi-object scenarios as well as a public road scenario. The tracking accuracy was determined by comparing the estimates to precise ground truth data for each state component. Also, the cardinality estimates were assessed to measure the multi-object performance. Both radar models yield precise estimation results. Yet, a comparison between the two models revealed that the variational radar model for the most part outperforms the handcrafted direct scattering model. The applicability of the proposed approaches to different vehicle types was demonstrated using a public road scenario. In contrast to the radar models, the ray-based lidar model achieves preciser estimates of the vehicle pose but is less accurate in estimating the vehicle motion. Apart from tracking accuracy, the lidar evaluation analyzed the effects of the occlusion model on the multi-object performance. It showed that the model helps to continuously track vehicles through occlusions but that track losses still occur due to the increasing uncertainty if there are no measurements over longer occlusion periods. Finally, different sensor fusion constellations were compared and analyzed. It was shown that combining radar with lidar or scene labeling data improves the accuracy in both cases and that sensor fusion especially helps to diminish the weaknesses of the individual sensors. As an additional variant, all three sensor types where combined. This combination does not yield performance gains that are as big as those of the two-sensor constellations but adds to sensor redundancy. Also, it was demonstrated that the chosen architecture is able to deal with sensor failure and does not rely on particular sensor constellations. In the fusion experiments, the occlusion models proved to be particularly beneficial since they model the absence of information for one sensor and allow for focusing on remaining information from other sensors.

In summary, the developed framework and extended measurement models come with several advantageous characteristics. The FISST formulation provides a rigorous probabilistic formulation of the entire problem which clearly reveals approximations as well as assumptions. It also allows for a principled approach to track management

in terms of existence probabilities, which are obtained by systematically comparing vehicle hypotheses among each other and to clutter models. Additionally, the extended object models directly use all measurements and avoid preprocessing routines which face difficulties if meta-measurements cannot be clearly extracted and oftentimes require additional heuristics. For instance, the radar measurement models are able to use all available Doppler information in cases with many but also with only one radar detection. They are not restricted to particular maneuvers and the variational radar model even processes spurious wheel measurements. Also, the variational radar model avoids elaborate manual design and directly learns the likelihood function form actual data. Besides, the lidar model uses negative information, considers occlusions, and does not require special treatment at field of view (FOV) boundaries as it is often the case in classical bounding box preprocessing approaches.

Yet, there are also limitations to the developed methods that require consideration. A rather obvious but particularly important insight is that the fully probabilistic description only yields reasonable results if all probabilistic models are a sufficiently good approximation of reality. Deviations from the true densities directly entail consequences. For example, the assumption of independent and uniformly distributed clutter does not accurately model clutter measurements from stationary objects. This can lead to false tracks and to overconfidence in the cardinality estimate. Also, the particle implementation introduces approximation errors which may lead to imperfect posterior densities that do not capture the full uncertainty. Lastly, the prototype MATLAB implementation was not intended for real-time application and new fast implementations need to be developed for application in an autonomous vehicle.

7.2 Suggestions for Future Work

These limitations lead to several recommendations for future work. Among those, there are several aspects that could lead to further improvement of the methods themselves. For example, it would be worthwhile to find new and advanced clutter models which adapt themselves to the current conditions or directly estimate the clutter density under consideration of typical correlations. This could further improve the filter's ability in distinguishing true vehicles from false tracks. Additionally, the results of the variational radar model strongly suggest that learning sensor models from actual data is beneficial and should be transferred to the other sensor types. This would further contribute to using densities that closely approximate reality. The variational radar model itself has only been tested using a training data set of a single vehicle and obviously, using training data with multiple different

vehicles would lead to additional insights. Analyzing to what extent different vehicle types can be lumped into a single model and at which point a separation into multiple vehicle classes is better, would be particularly useful for the application to public road scenarios that, for example, comprise trucks. Apart from vehicles, a full-fledged environment perception system also needs to track other object types such as pedestrians or bicycles. This extension requires finding suitable combinations of class-specific extended object models that achieve good performance and work together well.

In terms of real-time capability, there are two ways that can be pursued. First implementations of the variational radar model in C++, that were not optimized, have achieved near real-time computations on a single central processing unit (CPU) core and suggest that real-time capability can be achieved through careful implementation using multiple cores. Certainly, the use of graphics processing units (GPUs) for computing the particle updates would solve this issue. As an alternative, the computational load of the particle implementation could be avoided by using other approximation techniques such as an unscented Kalman filter (UKF) implementation. For example, the Student's t-mixture of the variational radar model could be simplified to a Gaussian mixture and re-interpreted as a reflection point model. In combination with approximate association techniques that establish correspondences between reflection points and measurements, this would allow using UKFs.

One of the major limitations in extended object tracking research is the absence of common benchmark data sets that allow a fair comparison between the overwhelming variety of methods. This is particularly the case for radar data. Hence, the preparation of such a data set and a thorough evaluation of different extended object models would greatly help to obtain a conclusive comparison of the different methods' strengths and weaknesses.

Besides the development of extended object methods, there has been a drastic increase of deep learning for vehicle environment perception. Deep learning methods are transferred from computer vision tasks to other sensor types and are, for instance, used for object classification or detection. Such approaches draw their performance from large data sets and are thus able to infer plausible outcomes even if the current data is ambiguous. As they additionally avoid manual engineering, the quality of preprocessing results is currently improving. For example, fitting bounding boxes to 3D lidar data (see e.g. [LZX16]) yields promising results and could alleviate many of the typical hand-crafted box fitting algorithms.

In contrast to deep learning approaches, Bayesian extended object tracking focuses on using probabilistic measurement models that capture the uncertainty in the data. Therefore, they are appropriate whenever the ambiguity in the data of a single time step is so large that using preprocessing results, e.g. the average result from many

observations, become deficient and an accumulation of information over several time steps is necessary. At the current time, this appears to particularly be the case for radar data and interestingly, the development of deep learning approaches for radar sensors seems to lag behind the other sensor types. Nevertheless, bounding box regression using deep learning and radar data is most certainly forthcoming and for future work, a through comparison between such methods and extended object tracking would be highly interesting.

If high-performance preprocessing routines are available, the basic concept of the variational radar model could be used to bridge the cap between deep learning methods and Bayesian tracking. The idea of learning a measurement model using variational Gaussian mixtures (VGMs) can, for instance, be transferred to measurement models on bounding box level. Thus, the true uncertainty in the bounding boxes as well as different preprocessing characteristics that, for instance, depend on the aspect angle could be learned from data and used to achieve improved tracking performance.

Appendix A

Marginal and Conditional of the Multivariate Student's t-Distribution

For the sake of completeness, this appendix lists the marginal and conditional distributions of a multivariate Student's t-distribution as provided in [Rot13]. Additionally, the marginal and conditional densities of Student's t-mixtures are provided. The marginal distribution is necessary for computing the measurement likelihood of the variational radar model through (5.26). Alternatively, the conditional distribution can be used directly.

Let $x = [x_1^{\mathrm{T}}, x_2^{\mathrm{T}}]^{\mathrm{T}}$ be a random variable that is distributed according to a multivariate Student's t-distribution $\mathrm{St}(x|\mu, \underline{\Sigma}^{-1}, \nu)$ of the form (2.15) and with parameters $\mu = [\mu_1^{\mathrm{T}}, \mu_2^{\mathrm{T}}]^{\mathrm{T}}$,

$$\underline{\Sigma} = \begin{bmatrix} \underline{\Sigma}_{11} & \underline{\Sigma}_{12} \\ \underline{\Sigma}_{21} & \underline{\Sigma}_{22} \end{bmatrix}, \tag{A.1}$$

and ν. The marginal distribution over x_1 is

$$p(x_1) = \int p(x_1, x_2)\, \mathrm{d}x_2 = \mathrm{St}(x_1|\mu_1, \underline{\Sigma}_{11}, \nu). \tag{A.2}$$

The conditional distribution of x_1 given a particular value for x_2 is

$$p(x_1|x_2) = \mathrm{St}(x_1|\mu_{1|2}, \underline{\Sigma}_{1|2}, \nu_{1|2}) \tag{A.3}$$

with

$$\mu_{1|2} = \mu_1 + \underline{\Sigma}_{12}\underline{\Sigma}_{22}^{-1}(x_2 - \mu_2), \tag{A.4}$$

$$\underline{\Sigma}_{1|2} = \frac{\nu + (x_2 - \mu_2)^{\mathrm{T}}\underline{\Sigma}_{22}^{-1}(x_2 - \mu_2)}{\nu + n_{x_2}} \left(\underline{\Sigma}_{11} - \underline{\Sigma}_{12}\underline{\Sigma}_{22}^{-1}\underline{\Sigma}_{12}^{\mathrm{T}}\right), \tag{A.5}$$

and

$$\nu_{1|2} = \nu + n_{x_2}. \tag{A.6}$$

For a Student's t-mixture in the form of

$$p(x_1, x_2) = p(x) = \sum_{i=1}^{c} w_i p_i(x) = \sum_{i=1}^{c} w_i \,\mathrm{St}(x|\mu^{(i)}, \underline{\Sigma}_{(i)}^{-1}, \nu_i), \tag{A.7}$$

the marginal density is

$$\begin{aligned} p(x_1) &= \int p(x_1, x_2)\,\mathrm{d}x \\ &= \int \sum_{i=1}^{c} w_i p_i(x_1, x_2)\,\mathrm{d}x_2 \\ &= \sum_{i=1}^{c} w_i \int p_i(x_1, x_2)\,\mathrm{d}x_2 \\ &= \sum_{i=1}^{c} w_i p_i(x_1), \end{aligned} \tag{A.8}$$

where $p_i(x_1)$ is the marginal distribution of the i-th component according to (A.2). The conditional density is

$$p(x_1|x_2) = \frac{p(x_1, x_2)}{p(x_2)} = \frac{\sum_{i=1}^{c} w_i p_i(x_1, x_2)}{\sum_{j=1}^{c} w_j p_j(x_2)} = \sum_{i=1}^{c} \frac{w_i p_i(x_2)}{\sum_{j=1}^{c} w_j p_j(x_2)} p_i(x_1|x_2), \tag{A.9}$$

where the densities $p_i(x_2)$ and $p_i(x_1|x_2)$ are again the respective marginal and conditional densities from (A.2) and (A.3).

Appendix B

Update Equations for the Variational Gaussian Mixture

Computing the optimal approximation of the posterior over the latent variables from (2.39) involves iteratively maximizing the lower bound from (2.27) with respect to $q(L)$ and $q(w, M, H)$. That is, (2.31) is evaluated in turns to obtain the optimal solutions $q^*(L)$ and $q^*(w, M, H)$.

Inserting the joint distribution $p(Z, L, w, M, H)$ from (2.37) into (2.31), computing the expected value with respect to w, M, and H, and normalizing the result yields the optimal solution $q^*(L)$. It is again a product of m multinomial distributions of the form (2.33) [Bis13, pp. 476-477],

$$q^*(L) = \prod_{i=1}^{m} \text{Mult}(l^{(i)}|\tau^{(i)}). \tag{B.1}$$

Here, each $\tau^{(i)}$ is normalized such that $\sum_{j=1}^{c} \tau_j^{(i)} = 1$ and

$$\tau_j^{(i)} \propto \tilde{w}_j \tilde{h}_j^{\frac{1}{2}} \exp\left(-\frac{n_z}{2\beta_j} - \frac{\nu_j}{2}(z^{(i)} - \gamma^{(j)})^{\mathrm{T}} \underline{V}_{(j)} (z^{(i)} - \gamma^{(j)})\right), \tag{B.2}$$

with

$$\tilde{h}_j = \exp\left(\sum_{i=1}^{n_z} \Psi\left(\frac{\nu_j + 1 - i}{2}\right) + n_z \ln 2 + \ln\det(\underline{V}_{(j)})\right), \tag{B.3}$$

$$\tilde{w}_j = \exp\left(\Psi(\rho_j) - \Psi(\tilde{\rho})\right), \tag{B.4}$$

the diagamma function $\Psi(\cdot)$ [Bis13, p. 687], and where $\tilde{\rho} = \sum_{j=1}^{c} \rho_j$.

Similarly, the optimal solution $q^*(w, M, H)$ is obtained by inserting (2.37) into

(2.31) and computing the expected value with respect to L. This procedure yields a factorized optimal result in the form $q^*(w, M, H) = q^*(w)q^*(M, H)$. These factors are a Dirichlet distribution [Bis13, p.478]

$$q^*(w) = \mathrm{Dir}(w|\rho), \tag{B.5}$$

where the elements of the updated parameter are

$$\rho_j = \rho_0 + s_j, \tag{B.6}$$

and

$$s_j = \sum_{i=1}^{m} \tau_j^{(i)}. \tag{B.7}$$

The optimal solution for $q^*(M, H)$ is the product of c Gaussian-Wishart distributions [Bis13, p. 478]

$$q^*(M, H) = \prod_{j=1}^{c} \mathcal{N}(\mu^{(j)}|\gamma^{(j)}, \beta_j^{-1}\underline{H}_{(j)}^{-1})\mathcal{W}(\underline{H}_{(j)}|\underline{V}_{(j)}, \nu_j) \tag{B.8}$$

with

$$\beta_j = \beta_0 + s_j, \tag{B.9}$$

$$\gamma^{(j)} = \frac{1}{\beta_j}(\beta_0\gamma_0 + s_j\bar{z}^{(j)}), \tag{B.10}$$

$$\bar{z}^{(j)} = \frac{1}{s_j}\sum_{i=1}^{m} \tau_j^{(i)} z^{(i)}, \tag{B.11}$$

$$\underline{V}_{(j)}^{-1} = \underline{V}_0^{-1} + s_j\underline{S}_{(j)} + \frac{\beta_0 s_j}{\beta_0 + s_j}(\bar{z}^{(j)} - \gamma_0)(\bar{z}^{(j)} - \gamma_0)^{\mathrm{T}}, \tag{B.12}$$

$$\underline{S}_{(j)} = \frac{1}{s_j}\sum_{i=1}^{m} \tau_j^{(i)}(z^{(i)} - \bar{z}^{(j)})(z^{(i)} - \bar{z}^{(j)})^{\mathrm{T}}, \tag{B.13}$$

and

$$\nu_j = \nu_0 + s_j + 1. \tag{B.14}$$

Acronyms

δ-GLMB	δ-generalized labeled multi-Bernoulli 35
ADMA	Automotive Dynamic Motion Analyzer 93
CBMeMBer	cardinality-balanced multi-target multi-Bernoulli 39, 59
CCD	charge-coupled device 93
CNN	convolutional neural network 54, 126
CPHD	cardinalized probability hypothesis density 39, 59
CPU	central processing unit 140
CTRV	constant turn rate and velocity 62, 103, 154
DBSCAN	density-based spatial clustering of applications with noise 106
DGPS	differential global positioning system 93
EKF	extended Kalman filter 23, 55
FISST	finite set statistics ..... ix, 4, 22, 57, 67, 137
FOV	field of view 1, 26, 52, 85, 91, 139
GLMB	generalized labeled multi-Bernoulli 34, 59, 72, 137
GP	Gaussian process 44, 51
GPS	global positioning system 94
GPU	graphics processing unit 101, 140
i.i.d.	independent and identically distributed 31, 80
IMU	inertial measurement unit 93
JIPDA	joint integrated probabilistic data association 27, 55
JPDA	joint probabilistic data association 27
LMB	labeled multi-Bernoulli 33, 55, 73, 137

MAP	maximum a posteriori 7
MHT	multiple hypotheses tracking 27
MMSE	minimum mean square error 7, 95
NN	nearest neighbor 27
PDA	probabilistic data association 27
PHD	probability hypothesis density 30, 55
PMBM	Poisson multi-Bernoulli mixture 40, 55
RANSAC	random sample consensus 106
RBPF	Rao-Blackwellized particle filter 26
RFS	random finite set 4, 28, 57, 79, 137
RHM	random hypersurface model 44, 153
RMSE	root mean squared error 95
SIR	sequential importance resampling 24
SMC	sequential Monte Carlo 61
UKF	unscented Kalman filter 24, 55, 140
VGM	variational Gaussian mixture 16, 85, 98, 141

Symbols

General Notation

a	scalar or vector
a_i	i-th element of a vector
$\hat{a}$	estimate of the random variable a
$\bar{a}$	mean or expected value of the random variable a
$\underline{A}$	matrix
$\underline{A}^{-1}$	inverse of matrix $\underline{A}$
$\underline{A}^{\mathrm{T}}$	transpose of matrix $\underline{A}$
A	set
$a^{(i)}$	i-th element of a set A
$\underline{A}_{(i)}$	i-th element of a set of matrices A
$\lvert A \rvert$	cardinality of a set
$\mathbf{a}$	labeled vector
$\mathbf{A}$	labeled set
$\boldsymbol{a}(\cdot)$	function or density of a labeled quantity
$\mathbb{A}$	space
$\mathrm{d}a$	differential of a scalar or vector a
$\det(\underline{A})$	determinant of the matrix $\underline{A}$
$\mathbb{E}[\cdot]$	expectation operator
$\mathbb{E}_a[\cdot]$	expectation with respect to the random variable a
$\exp(\cdot)$	natural exponential function
$[h(\cdot)]^A$	multi-object exponential, i.e. product of the real-valued function $h(\cdot)$ applied to all elements of A
n_a	dimension of the vector a
$\mathrm{Tr}(\underline{A})$	trace of the matrix $\underline{A}$
δA	differential of a set A as used in set integrals

Densities

$f(\cdot)$	single object or multi-object transition density
$\boldsymbol{f}(\cdot)$	labeled multi-object transition density
$g(\cdot)$	measurement likelihood
$g_C(\cdot)$	density of the clutter measurement set
$p(a)$	density function over some random variable a
$p(a\|b)$	conditional density of a random variable a given b
$p_C(\cdot)$	density of a single clutter measurement
$q(\cdot)$	proposal density in importance sampling or auxiliary density in variational Bayes
$q^*(\cdot)$	optimal proposal density
$\pi(\cdot)$	mutli-object state density
$\boldsymbol{\pi}(\cdot)$	labeled mutli-object state density
$\varphi_i(\cdot)$	foreground likelihood of the i-th measurement in the separable likelihood model
$\psi(\cdot)$	background likelihood in the separable likelihood model
$\psi_{\mathcal{U}(Z)}(\cdot,\cdot\|\theta)$	generalized likelihood in the generalized standard multi-object likelihood for given partition $\mathcal{U}(Z)$ and association mapping θ

Distributions

$\mathrm{Ber}(x\|r)$	Bernoulli distribution with parameter r
$\mathrm{Dir}(x\|\rho)$	Dirichlet distribution with parameter ρ
$\mathrm{Mult}(x\|w)$	multinomial distribution with parameter vector w
$\mathcal{N}(x\|\mu,\underline{\Sigma})$	multivariate Gaussian distribution over the random variable a with mean μ and covariance matrix $\underline{\Sigma}$
$\mathrm{Poi}(x\|\lambda)$	Bernoulli distribution with rate parameter λ
$\mathrm{St}(x\|\mu,\underline{\Lambda},\nu)$	multivariate Student's t-distribution over the random variable a with parameters μ, $\underline{\Lambda}$, and ν degrees of freedom
$\mathrm{U}(x\|u_l,u_u)$	uniform distribution over the interval $[u_l,u_u]$
$\mathcal{W}(\underline{X}\|\underline{V},\nu)$	Wishart distribution over the random matrix $\underline{A}$ with parameter $\underline{V}$ and ν degrees of freedom

Functions

$1_A(a)$	inclusion function which is 1 if and only if $a \in A$
$A(\cdot)$	function that returns the area that is occupied by a vehicle in dependence on its state
$C(\rho)$	normalization factor of a Dirichlet distribution in dependence on the parameters ρ

$f_m(\cdot)$	transition function of the system model
$f_x(\cdot)$	state transformation function
$f_z(\cdot)$	measurement transformation function
$h_m(\cdot)$	measurement function of the system model
$k(\cdot,\cdot)$	Gaussian process kernel
$\mathrm{KL}(\cdot\|\|\cdot)$	Kullback-Leibler divergence
$B(\underline{V},\nu)$	normalization factor of the Wishart distribution
$\mathcal{L}(\boldsymbol{X})$	label projection function that retrieves all labels from a labeled set $\boldsymbol{X}$
$r(\cdot)$	radius function in contour models
$w(\cdot)$	weight of a multi-object hypothesis (label set as argument) or a particle / importance weight (state vector as argument)
$\tilde{w}(\cdot)$	importance weight that has not yet been normalized
$\Gamma(\cdot)$	gamma function
$\Delta(\boldsymbol{X})$	distinct label indicator which is 1 if and only if all labels in $\boldsymbol{X}$ are unique
$\delta(\cdot)$	Kronecker-delta function
$\eta(\cdot)$	normalization constant
$\theta(\cdot)$	association mapping
$\kappa(\cdot)$	intensity function of the clutter process
$\Psi(\cdot)$	diagamma function

Matrices

$\underline{H}$	precision matrix of the multivariate Gaussian distribution
$\underline{S}$	sufficient statistic in computing the optimal solution in VGMs
$\underline{V}$	matrix parameter of the Wishart distribution
$\underline{\Lambda}$	precision matrix of the multivariate Student's t-distribution

Probabilities

p_D	probability of detection
$p_D(\cdot)$	probability of detection as a function of the object state
$p_O(\cdot,\cdot)$	overlap probability for two objects as function of their labels
$p_{\mathrm{R,bgr}}$	probability of receiving a return in the background likelihood of the lidar model
$p_{\mathrm{R},j}$	probability of receiving a return in the foreground likelihood of the lidar model where $j = \{1,2,3\}$ for the three ray types
$p_S(\cdot)$	probability of persistence as a function of the object state
$q_D(\cdot)$	probability of misdetection as a function of the object state

$q_S(\cdot)$	probability of disappearance as a function of the object state

Subscripts

0	initial value
$1:k$	set of all measurements or measurement sets from the first to the k-th time step
B	density or variable corresponding to a newly-born object
k	time step
OC	object coordinates
SC	sensor coordinates
$\mathcal{U}(Z)$	for a given partition $\mathcal{U}(Z)$
VC	vehicle coordinates
$+$	prior quantity

Sets

H	set of precision matrices $\underline{H}$
I	label subset
J	label subset
L	set of latent variables l
M	set of Gaussian distribution means μ
$\mathcal{P}_i(Z)$	set of all measurement partitions with i clusters
$T(\cdot)$	set of affected measurement indices in the lidar and scene labeling measurement models
X	multi-object state
$\boldsymbol{X}$	labeled multi-object state
Z	measurement set
Z_D	training data set
Z_O	set of measurements that originate from a specific object
Φ	parameter set of the VGM model

Spaces

$\mathbb{L}$	label space
$\mathbb{N}$	natural numbers excluding zero
$\mathbb{N}_0$	natural numbers including zero
$\mathbb{X}$	state space
Θ	space of all possible associations
$\mathbb{Z}$	measurement space

Scalars and Vectors

a	vehicle width
a_i	Fourier series coefficient in the random hypersurface model (RHM)
b	vehicle length
b_i	Fourier series coefficient in the RHM
c	number of components in a mixture
d	range
l	assignment vector and latent variable in Gaussian mixture model
ℓ	object label
m	number of measurements or data points
n	number of objects
$\hat{n}_{\text{eff}}$	effective sample size in particle filters
n_p	number of particles
p	parameter vector of the RHM
r	parameter of the Bernoulli distribution and existence probability
s	scaling factor in the RHM
s_j	sufficient statistic in computing the optimal solution in VGMs
$s_{D,1}, s_{D,2}$	lumped state terms in the Doppler equation (5.16)
u_l, u_u	lower and upper interval bounds in a uniform distribution
v	vehicle speed
v_D	Doppler velocity
$\tilde{v}_D$	expected Doppler velocity
v_O	object velocity at the measurement location
w	parameter vector of the multinomial distribution which denotes the event probabilities and weight vector in the VGM model
x	latent variable or state vector
$\boldsymbol{x}$	labeled state vector
x'	transformed state vector
x_R	x-position of the rear axle center
y	measurement source in spatial distribution models
y_R	y-position of the rear axle center
z	measurement vector or observed variable
z_D	training data vector
$z_{\max}$	maximum sensor range in the lidar model
z'	transformed measurement vector
α	azimuth angle

β	scaling parameter in the Gaussian portion of the Gaussian-Wishart prior
γ	mean vector of the Gaussian portion in the Gaussian-Wishart prior
Δt	time difference between two filter cycles
ϵ	additive noise
ϵ_f	process noise in the discrete-time system model
ϵ_h	measurement noise in the discrete-time system model
ϵ_v	process noise of the speed in the constant turn rate and velocity (CTRV) model
ϵ_x	process noise of the x_R-position in the CTRV model
ϵ_y	process noise of the y_R-position in the CTRV model
ϵ_φ	process noise of the orientation in the CTRV model
ϵ_ω	process noise of the yaw rate in the CTRV model
ζ	extent state vector portion
θ	association mapping between objects and measurements
κ	angle of the connecting line between center and contour in star convex shapes
λ_C	expected number of clutter measurements
λ_T	expected number of measurements from an object
μ	mean of the multivariate Gaussian or multivariate Student's t-distribution
ν	degrees of freedom in a distribution
ξ	kinematic state vector portion
ρ	parameter of Dirichlet distribution
τ	responsibilities in VGMs
φ	vehicle yaw angle
ω	vehicle yaw rate

Miscellaneous

$\mathcal{F}$	event that a multi-object state is physically consistent
$\mathcal{L}(q(X))$	lower bound in variational Bayes which is a functional of the auxiliary density $q(X)$
$\mathcal{U}(Z)$	partition of the measurements Z

Bibliography

[AK65] Ando, Alberto and Kaufman, G. M.: *Bayesian Analysis of the Independent Multinormal Process. Neither Mean Nor Precision Known.* In: *Journal of the American Statistical Association*, volume 60, no. 309, pages 347–358, 1965.

[AMGC02] Arulampalam, M. Sanjeev; Maskell, Simon; Gordon, Neil J.; and Clapp, Tim: *A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking.* In: *IEEE Transactions on Signal Processing*, volume 50, no. 2, pages 174–188, 2002.

[ASKB12] Aeberhard, Michael; Schlichthärle, Stefan; Kämpchen, Nico; and Bertram, Torsten: *Track-to-Track Fusion with Asynchronous Sensors Using Information Matrix Fusion for Surround Environment Perception.* In: *IEEE Transactions on Intelligent Transportation Systems*, volume 13, no. 4, pages 1717–1726, 2012.

[ASW13] Adam, Christian; Schubert, Robin; and Wanielik, Gerd: *Radar-Based Extended Object Tracking Under Clutter using Generalized Probabilistic Data Association.* In: *Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems*, pages 1408–1415, 2013.

[Att00] Attias, Hagai: *A Variational Bayesian Framework for Graphical Models.* In: *Advances in Neural Information Processing Systems*, volume 12, pages 209–215, 2000.

[Att99] Attias, Hagai: *Inferring Parameters and Structure of Latent Variable Models by Variational Bayes.* In: *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence*, 1999.

[BA04] Borges, Geovany Araujo and Aldon, Marie-José: *Line Extraction in 2D Range Images for Mobile Robotics.* In: *Journal of Intelligent and Robotic Systems*, volume 40, no. 3, pages 267–297, 2004.

[Bar02] Bar-Shalom, Yaakov: *Update with Out-of-Sequnce Measurements in Tracking: Exact Solution.* In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 38, no. 3, pages 769–778, 2002.

[BDD17] Broßeit, Peter; Duraisamy, Bharanidhar; and Dickmann, Jürgen: *The Volcanormal Density for Radar-Based Extended Target Tracking*. In: *Proceedings of the 20th International IEEE Conference on Intelligent Transportation Systems*, 2017.

[Bea03] Beal, Matthew J.: *Variational Algorithms for Approximate Bayesian Inference*. Dissertation, University College London, London, UK, 2003.

[BH09] Baum, Marcus and Hanebeck, Uwe D.: *Random Hypersurface Models for Extended Object Tracking*. In: *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology*, pages 178–183, 2009.

[BH11] Baum, Marcus and Hanebeck, Uwe D.: *Shape Tracking of Extended Objects and Group Targets with Star-Convex RHMs*. In: *Proceedings of the 14th International Conference on Information Fusion*, 2011.

[BH14] Baum, Marcus and Hanebeck, Uwe D.: *Extended Object Tracking with Random Hypersurface Models*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 50, no. 1, pages 149–159, 2014.

[Bis13] Bishop, Christopher M.: *Pattern Recognition and Machine Learning*. From the series *Information science and statistics*. 11. (corr. printing), Springer, New York, 2013.

[BKBD15] Broßeit, Peter; Kellner, Dominik; Brenk, Carsten; and Dickmann, Jürgen: *Fusion of Doppler Radar and Geometric Attributes for Motion Estimation of Extended Objects*. In: *Proceedings of Sensor Data Fusion: Trends, Solutions, Applications*, 2015.

[Bla04] Blackman, Samuel S.: *Multiple Hypothesis Tracking For Multiple Target Tracking*. In: *IEEE Aerospace and Electronic Systems Magazine*, volume 19, no. 1, pages 5–18, 2004.

[BLK01] Bar-Shalom, Yaakov; Li, X. Rong; and Kirubarajan, Thiagalingam: *Estimation with Applications To Tracking and Navigation*. From the series *Electronic and electrical engineering*. Wiley, New York and Chichester, 2001.

[BML+17] Berthold, Philipp; Michaelis, Martin; Luettel, Thorsten; Meißner, Daniel; and Wünsche, Hans-Joachim: *Radar Reflection Characteristics of Vehicles for Contour and Feature Estimation*. In: *Proceedings of the 11th Symposium Sensor Data Fusion: Trends, Solutions, Applications*, 2017.

[BRAD16] Broßeit, Peter; Rapp, Matthias; Appenrodt, Nils; and Dickmann, Jürgen: *Probabilistic Rectangular-Shape Estimation for Extended Object Tracking*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 279–285, 2016.

[BRG+15] Beard, Michael; Reuter, Stephan; Granström, Karl; et al.: *A Generalised Labelled Multi-Bernoulli Filter for Extended Multi-target Tracking*. In: *Proceddings of the 18th International Conference on Information Fusion*, pages 991–998, 2015.

[BRG+16] Beard, Michael; Reuter, Stephan; Granström, Karl; et al.: *Multiple Extended Target Tracking With Labeled Random Finite Sets*. In: *IEEE Transactions on Signal Processing*, volume 64, no. 7, pages 1638–1653, 2016.

[BT08] Bertsekas, Dimitri P. and Tsitsiklis, John N.: *Introduction to Probability*. 2nd ed, Athena Scientific, Belmont, MA, 2008.

[BT75] Bar-Shalom, Yaakov and Tse, Edison: *Tracking in a Cluttered Environment With Probabilistic Data Association*. In: *Automatica*, volume 11, no. 5, pages 451–460, 1975.

[BTW11] Bar-Shalom, Yaakov; Tian, Xin; and Willett, Peter K.: *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, Storrs, CT, 2011.

[BY06] Bühren, Markus and Yang, Bin: *Simulation of Automotive Radar Target Lists using a Novel Approach of Object Representation*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 314–319, 2006.

[CSKR14] Cho, Hyunggi; Seo, Young-Woo; Kumar, B.V.K. Vijaya; and Rajkumar, Ragunathan: *A Multi-Sensor Fusion System for Moving Object Detection and Tracking in Urban Environments*. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1836–1843, 2014.

[DAK+15] Dickmann, Jürgen; Appenrodt, Nils; Klappstein, Jens; et al.: *Making Bertha See Even More: Radar Contribution*. In: *IEEE Access*, volume 3, pages 1233–1247, 2015.

[DdMR00] Doucet, Arnaud; de Freitas, Nando; Murphy, Kevin; and Russel, Stuart: *Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks*. In: *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, pages 176–183, 2000.

[Dez98] Dezert, Jean C.: *Tracking maneuvering and bending extended target in cluttered environment*. In: *Proceedings SPIE 3373, Signal and Data Processing of Small Targets*, 1998.

[DGA00] Doucet, Arnaud; Godsill, Simon J.; and Andrieu, Christophe: *On sequential Monte Carlo sampling methods for Bayesian filtering*. In: *Statistics and Computing*, volume 10, no. 3, pages 197–208, 2000.

[DGV+16] Duraisamy, Bharanidhar; Gabb, Michael; Vijayamohnan Nair, Aswin; Schwarz, Tilo; and Yuan, Ting: *Track Level Fusion of Extended Objects from Heterogeneous Sensors*. In: *Proceedings of the 19th International Conference on Information Fusion*, 2016.

[DH03] Daum, Fred and Huang, Jim: *Curse of Dimensionality and Particle Filters*. In: *Proceedings of the IEEE Aerospace Conference*, Vol. 4-1979–Vol. 4-1993, 2003.

[DRBU09] Darms, Michael S.; Rybski, Paul E.; Baker, Christopher; and Urmson, Chris: *Obstacle Detection and Tracking for the Urban Challenge*. In: *IEEE Transactions on Intelligent Transportation Systems*, volume 10, no. 3, pages 475–485, 2009.

[EAK16] Elfring, Jos; Appeldoorn, Rein; and Kwakkernaat, Maurice: *Multi-sensor Simultaneous Vehicle Tracking and Shape Estimation*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 630–635, 2016.

[Eff09] Effertz, Jan: *Autonome Fahrzeugführung in urbaner Umgebung durch Kombination objekt- und kartenbasierter Umfeldmodelle*. Dissertation, Technische Universität Braunschweig, Brunswick, Germany, 2009.

[EKSX96] Ester, Martin; Kriegel, Hans-Peter; Sander, Jörg; and Xu, Xiaowei: *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*. In: *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, pages 226–231, 1996.

[FB81] Fischler, Martin A. and Bolles, Robert C.: *Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography*. In: *Communications of the ACM*, volume 24, no. 6, pages 381–395, 1981.

[FBS83] Fortmann, Thomas E.; Bar-Shalom, Yaakov; and Scheffe, Molly: *Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association*. In: *IEEE Journal of Oceanic Engineering*, volume OE-8, no. 3, pages 173–184, 1983.

[FCNL13] Farabet, Clément; Couprie, Camille; Najman, Laurent; and LeCun, Yann: *Learning Hierarchical Features for Scene Labeling*. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 35, no. 8, pages 1915–1929, 2013.

[FEHP11] Forbes, Catherine. S.; Evans, Merran; Hastings, Nicholas; and Peacock, Brian: *Statistical Distributions*. From the series *Wiley series in probability and statistics*. 4th ed., Wiley, Hoboken, NJ, 2011.

[Fin97] Fink, Daniel: *A Compendium of Conjugate Priors*. Bozeman, MT, 1997.

[FR06] Fölster, Florian and Rohling, Hermann: *Lateral velocity estimation based on automotive radar sensors.* In: *Proceedings of the CIE International Conference on Radar*, 2006.

[GBR17] Granström, Karl; Baum, Marcus; and Reuter, Stephan: *Extended Object Tracking: Introduction, Overview, and Applications.* In: *Journal of Advances in Information Fusion*, volume 12, no. 2, pages 139–174, 2017.

[GCS+14] Gelman, Andrew; Carlin, John B.; Stern, Hal Steven; et al.: *Bayesian Data Analysis.* From the series *Chapman & Hall/CRC texts in statistical science.* Third edition, Boca Raton, FL, 2014.

[Ger89] Gertz, Jeffrey L.: *Multisensor Surveillance for Improved Aircraft Tracking.* In: *The Lincoln Laboratory Journal*, volume 2, no. 3, pages 381–396, 1989.

[GFS16] Granström, Karl; Fatemi, Maryam; and Svensson, Lennart: *Gamma Gaussian inverse-Wishart Poisson multi-Bernoulli Filter for Extended Target Tracking.* In: *Proceedings of the 19th International Conference on Information Fusion*, 2016.

[GG16] Gotzig, Heinrich and Geduld, Georg: *Automotive LIDAR.* In: Winner, Hermann; Hakuli, Stephan; Lotz, Felix; and Singer, Christina (editors): *Handbook of Driver Assistance Systems.* From the series *Springer reference*, pages 405–430, Springer, 2016.

[GGMS05] Gilholm, Kevin; Godsill, Simon J.; Maskell, Simon; and Salmond, David J.: *Poisson models for extended target and group tracking.* In: *Proceedings SPIE 5913, Signal and Data Processing of Small Targets.* 59 130R, 2005.

[GLO11] Granström, Karl; Lundquist, Christian; and Orguner, Omut: *Tracking Rectangular and Elliptical Extended Targets Using Laser Measurements.* In: *Proceedings of the 14th International Conference on Information Fusion*, 2011.

[GRMS14] Granström, Karl; Reuter, Stephan; Meißner, Daniel; and Scheel, Alexander: *A multiple model PHD approach to tracking of cars under an assumed rectangular shape.* In: *Proceedings of the 17th International Conference on Information Fusion*, 2014.

[GS05] Gilholm, Kevin and Salmond, David J.: *Spatial distribution model for tracking extended objects.* In: *IEE Proceedings - Radar, Sonar and Navigation*, volume 152, no. 5, pages 364–371, 2005.

[GSDB07] Gunnarsson, Joakim.; Svensson, Lennart; Danielsson, Lars; and Bengtsson, Fredrik: *Tracking vehicles using radar detections.* In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 296–302, 2007.

[GSS93] Gordon, Neil J.; Salmond, David J.; and Smith, A. F. M., *Novel approach to nonlinear/non-Gaussian Bayesian state estimation*. In: *IEE Proceedings F - Radar and Signal Processing*, volume 140, no. 2, pages 107–113, 1993.

[GT08] Gamarnik, David and Tsitsiklis, John N.: *Fundamentals of Probability: Derived Distributions*. Cambridge, MA, 2008.

[GWGS18] García-Fernández, Ángel F.; Williams, Jason L.; Granström, Karl; and Svensson, Lennart: *Poisson multi-Bernoulli mixture filter: direct derivation and implementation*. In: *IEEE Transactions on Aerospace and Electronic Systems*, 2018.

[HHR+15] Hirsenkorn, Nils; Hanke, Timo; Rauch, Andreas; et al.: *A Non-Parametric Approach for Modelling Sensor Behavior*. In: *Proceedings of the 16th International Radar Symposium*, 2015.

[HHR+16] Hirsenkorn, Nils; Hanke, Timo; Rauch, Andreas; et al.: *Virtual sensor models for real-time applications*. In: *Advances in Radio Science*, volume 14, pages 31–37, 2016.

[HLS12] Hammarstrand, Lars; Lundgren, Malin; and Svensson, Lennart: *Adaptive Radar Sensor Model for Tracking Structured Extended Objects*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 48, no. 3, pages 1975–1995, 2012.

[HLT13] Held, David; Levinson, Jesse; and Thrun, Sebastian: *Precision Tracking with Sparse 3D and Dense Color 2D Data*. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1138–1145, 2013.

[HSRD16] Hirscher, Tobias; Scheel, Alexander; Reuter, Stephan; and Dietmayer, Klaus: *Multiple Extended Object Tracking Using Gaussian Processes*. In: *Proceedings of the 19th International Conference on Information Fusion*, 2016.

[HSSS12] Hammarstrand, Lars; Svensson, Lennart; Sandblom, Fredrik; and Sörstedt, Joakim: *Extended Object Tracking using a Radar Resolution Model*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 48, no. 3, pages 2371–2386, 2012.

[HU91] Howson, Colin and Urbach, Peter: *Bayesian reasoning in science*. In: *Nature*, volume 350, no. 6317, pages 371–374, 1991.

[HVV13] Hoseinnezhad, Reza; Vo, Ba-Ngu; and Vo, Ba-Tuong: *Visual Tracking in Background Subtracted Image Sequences via Multi-Bernoulli Filtering*. In: *IEEE Transactions on Signal Processing*, volume 61, no. 2, pages 392–397, 2013.

[HVVS12] Hoseinnezhad, Reza; Vo, Ba-Ngu; Vo, Ba-Tuong; and Suter, David: *Visual Tracking of Numerous Targets via Multi-Bernoulli Filtering of Image Data*. In: *Pattern Recognition*, volume 45, no. 10, pages 3625–3636, 2012.

[Jen06] Jensen, Johan Ludwig William Valdemar, *Sur les fonctions convexes et les inégalités entre les valeurs moyennes*. In: *Acta Mathematica*, volume 30, 1906.

[JU97] Julier, Simon J. and Uhlmann, Jeffrey K.: *New extension of the Kalman Filter to Nonlinear Systems*. In: *Proceedings SPIE 3068, Signal Processing, Sensor Fusion, and Target Recognition VI*, 1997.

[Kal60] Kalman, Rudolf Emil: *A New Approach to Linear Filtering and Prediction Problems*. In: *Journal of Basic Engineering*, volume 82, no. 1, pages 35–45, 1960.

[KBD+13] Kellner, Dominik; Barjenbruch, Michael; Dietmayer, Klaus; Klappstein, Jens; and Dickmann, Jürgen: *Instantaneous Lateral Velocity Estimation of a Vehicle using Doppler Radar*. In: *Proceedings of the 16th International Conference on Information Fusion*, pages 877–884, 2013.

[KBD05] Kämpchen, Nico; Bühler, Matthias; and Dietmayer, Klaus: *Feature-Level Fusion for Free-Form Object Tracking using Laserscanner and Video*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 453–458, 2005.

[KBK+14] Kellner, Dominik; Barjenbruch, Michael; Klappstein, Jens; Dickmann, Jürgen; and Dietmayer, Klaus: *Instantaneous Full-Motion Estimation of Arbitrary Objects using Doppler Radar*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 324–329, 2014.

[KBK+16] Kellner, Dominik; Barjenbruch, Michael; Klappstein, Jens; Dickmann, Jürgen; and Dietmayer, Klaus: *Tracking of Extended Objects with High-Resolution Doppler Radar*. In: *IEEE Transactions on Intelligent Transportation Systems*, volume 17, no. 5, pages 1341–1353, 2016.

[KBS17] Krämer, Stefan; Bouzouraa, Mohammed Essayed; and Stiller, Christoph: *Simultaneous Tracking and Shape Estimation Using a Multi-Layer Laserscanner*. In: *Proceedings of the 20th International IEEE Conference on Intelligent Transportation Systems*, pages 1826–1832, 2017.

[KKD12] Kellner, Dominik; Klappstein, Jens; and Dietmayer, Klaus: *Grid-Based DBSCAN for Clustering Extended Objects in Radar Data*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 365–370, 2012.

[Kni15] Knill, Christina: *Extended Object Tracking for Vehicle Environment Perception Using High-Resolution Radars*. Master Thesis, Ulm University, Ulm, Germany, 2015.

[KNW+15] Kunz, Felix; Nuss, Dominik; Wiest, Jürgen; et al.: *Autonomous Driving at Ulm University: A Modular, Robust, and Sensor-Independent Fusion Approach*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 666–673, 2015.

[Koc04] Koch, Johann Wolfgang: *On 'Negative' Information in Tracking and Sensor Data Fusion: Discussion of Selected Examples*. In: *Proceedings of the Seventh International Conference on Information Fusion*, 2004.

[Koc08] Koch, Johann Wolfgang: *Bayesian Approach to Extended Object and Cluster Tracking using Random Matrices*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 44, no. 3, pages 1042–1059, 2008.

[KSD16] Knill, Christina; Scheel, Alexander; and Dietmayer, Klaus: *A Direct Scattering Model for Tracking Vehicles with High-Resolution Radars*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 298–303, 2016.

[KYY+15] Kim, Beomjun; Yi, Kyongsu; Yoo, Hyun-Jae; Chong, Hyok-Jin; and Ko, Bongchul: *An IMM/EKF Approach for Enhanced Multitarget State Estimation for Application to Integrated Risk Management System*. In: *IEEE Transactions on Vehicular Technology*, volume 64, no. 3, pages 876–889, 2015.

[LGO13] Lundquist, Christian; Granström, Karl; and Orguner, Omut: *An Extended Target CPHD Filter and a Gamma Gaussian Inverse Wishart Implementation*. In: *IEEE Journal of Selected Topics in Signal Processing*, volume 7, no. 3, pages 472–483, 2013.

[LSD15] Long, Jonathan; Shelhamer, Evan; and Darrell, Trevor: *Fully Convolutional Networks for Semantic Segmentation*. In: *Proceedings of the 2015 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[LZX16] Li, Bo; Zhang Tianlei, and Xia, Tian: *Vehicle Detection from 3D Lidar Using Fully Convolutional Network*. In: *Proceedings of Robotics: Science and Systems 2016*, 2016.

[Mah03] Mahler, Ronald P. S., *Multitarget Bayes Filtering via First-Order Multitarget Moments*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 39, no. 4, pages 1552–1178, 2003.

[Mah04] Mahler, Ronald P. S., *"Statistics 101" for Multisensor, Multitarget Data Fusion*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 19, no. 1, pages 53–64, 2004.

[Mah07a] Mahler, Ronald P. S., *Statistical Multisource-Multitarget Information Fusion*. From the series *Artech House information warfare library*. Artech House, Boston, MA, 2007.

[Mah07b] Mahler, Ronald P. S., *PHD Filters of Higher Order in Target Number*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 43, no. 4, pages 1523–1543, 2007.

[Mah09] Mahler, Ronald P. S., *PHD filters for nonstandard targets, I: Extended targets*. In: *Proceedings of the 12th International Conference on Information Fusion*, pages 915–921, 2009.

[Mah13] Mahler, Ronald P. S., *"Statistics 102" for Multisource-Multitarget Detection and Tracking*. In: *IEEE Journal of Selected Topics in Signal Processing*, volume 7, no. 3, pages 376–389, 2013.

[Mah14] Mahler, Ronald P. S, *Advances in Statistical Multisource-Multitarget Information Fusion*. From the series *Artech House electronic warfare library*. Artech House, Boston, MA, 2014.

[MBMW17] Michaelis, Martin; Berthold, Philipp; Meißner, Daniel; and Wünsche, Hans-Joachim: *Heterogeneous Multi-Sensor Fusion for Extended Objects in Automotive Scenarios Using Gaussian Processes and a GMPHD-Filter*. In: *Proceedings of the 11th Symposium Sensor Data Fusion: Trends, Solutions, Applications*, 2017.

[MCH+08] Miller, Isaac; Campbell, Mark; Huttenlocher, Dan; et al.: *Team Cornell's Skynet: Robust Perception and Planning in an Urban Environment*. In: *Journal of Field Robotics*, volume 25, no. 8, pages 493–527, 2008.

[MDM09] Munz, Michael; Dietmayer, Klaus; and Mählisch, Mirko: *A Sensor Independent Probabilistic Fusion System for Driver Assistance Systems*. In: *Proceedings of the 12th International Conference on Information Fusion*, pages 438–443, 2009.

[ME02] Mušicki, Darko and Evans, Rob: *Joint Integrated Probabilistic Data Association - JIPDA*. In: *Proceedings of the Fifth International Conference on Information Fusion*, pages 1120–1125, 2002.

[ME04] Mušicki, Darko and Evans, Rob: *Joint Probabilistic Data Association: JIPDA*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 40, no. 3, pages 1093–1099, 2004.

[MMD10] Munz, Michael; Mählisch, Mirko; and Dietmayer, Klaus: *Generic Centralized Multi Sensor Data Fusion Based on Probabilistic Sensor and Environment Models for Driver Assistance Systems*. In: *IEEE Intelligent Transportation Systems Magazine*, volume 2, no. 1, pages 6–17, 2010.

[MNM+13] Mertz, Christoph; Navarro-Serment, Luis E.; MacLachlan Robert, et al.: *Moving Object Detection with Laser Scanners.* In: *Journal of Field Robotics*, volume 30, no. 1, pages 17–43, 2013.

[Mur68] Murty, Katta G.: *An Aglorithm for Ranking all the Assignments in Order of Increasing Cost.* In: *Operations Research*, volume 16, no. 3, pages 682–687, 1968.

[NH98] Neal, Radford M. and Hinton, Geoffrey E.: *A View of the EM Algorithm That Justifies Incremental, Sparse, and Other Variants.* In: Jordan, Michael Irwin (editor): *Learning in graphical models.* From the series *NATO ASI series. Series D, Behavioural and social sciences*, pages 355–368, 1998.

[PT09] Petrovskaya, Anna and Thrun, Sebastian: *Model Based Vehicle Detection and Tracking for Autonomous Urban Driving.* In: *Autonomous Robots*, volume 26, pages 123–139, 2009.

[PVV+15] Papi, Francesco; Vo, Ba-Ngu; Vo, Ba-Tuong; Fantacci, Claudio; and Beard, Michael: *Generalized Labeled Multi-Bernoulli Approximation of Multi-Object Densities.* In: *IEEE Transactions on Signal Processing*, volume 63, no. 20, pages 5487–5497, 2015.

[RAG04] Ristic, Branko; Arulampalam, Sanjeev; and Gordon, Neil J.: *Beyond the Kalman Filter: Particle Filters for Tracking Applications.* From the series *Artech House radar library.* Artech House, Boston, MA, 2004.

[RBL+08] Rauskolb, Fred W.; Berger, Kai; Lipski, Christian; et al.: *Caroline: An Autonomously Driving Vehicle for Urban Environments.* In: *Journal of Field Robotics*, volume 25, no. 9, pages 674–724, 2008.

[Rei79] Reid, Donald B.: *An Algorithm for Tracking Multiple Targets.* In: *IEEE Transactions on Automatic Control*, volume AC-24, no. 6, pages 843–854, 1979.

[Reu14] Reuter, Stephan: *Multi-Object Tracking Using Random Finite Sets.* Dissertation, Ulm University, Germany, 2014.

[RKDW16] Roos, Fabian; Kellner, Dominik; Dickmann, Jürgen; and Waldschmidt, Christian: *Reliable Orientation Estimation of Vehicles in High-Resolution Radar Images.* In: *IEEE Transactions on Microwave Theory and Techniques*, volume 64, no. 9, pages 2986–2993, 2016.

[RKK+15] Roos, Fabian; Kellner, Dominik; Klappstein, Jens; et al.: *Estimation of the Orientation of Vehicles in High-Resolution Radar Images.* In: *Proceedings of the IEEE MTT-S International Conference on Microwaves for Intelligent Mobility*, 2015.

[RMM15] Rieken, Jens; Matthaei, Richard; and Maurer, Markus: *Toward Perception-Driven Urban Environment Modeling for Automated Road Vehicles*. In: *Proceedings of the 18th International IEEE Conference on Intelligent Transportation Systems*, pages 731–738, 2015.

[Rot13] Roth, Michael: *On the Multivariate t Distribution*. Linköping, Sweden, 2013.

[RS70] Raiffa, Howard and Schlaifer, Robert: *Applied Statistical Decision Theory*. From the series *Studies in managerial economics*. 5. print, Div. of Research, Graduate School of Business Administration, Harvard University, Boston, MA, 1970.

[RVVD14] Reuter, Stephan; Vo, Ba-Tuong; Vo, Ba-Ngu; and Dietmayer, Klaus: *The Labeled Multi-Bernoulli Filter*. In: *IEEE Transactions on Signal Processing*, volume 62, no. 12, pages 3246–3260, 2014.

[RW06] Rasmussen, Carl Edward and Williams, Christopher K. I., *Gaussian Processes for Machine Learning*. MIT Press, Camebridge, MA, 2006.

[RWD12] Reuter, Stephan; Wilking, Benjamin; and Dietmayer, Klaus: *Methods to Model the Motion of Extended Objects in Multi-Object Bayes Filters*. In: *Proceedings of the 15th International Conference on Information Fusion*, pages 527–534, 2012.

[RWW+13] Reuter, Stephan; Wilking, Benjamin; Wiest, Jürgen; Munz, Michael; and Dietmayer, Klaus: *Real-Time Multi-Object Tracking using Random Finite Sets*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 49, no. 4, pages 2666–2678, 2013.

[RX17] Ru, Jifeng and Xu, Cuichun: *Improvement on Velocity Estimation of an Extended Object*. In: *Proceedings of the 20th International Conference on Information Fusion*, 2017.

[SADD13] Schütz, Markus; Appenrodt, Nils; Dickmann, Jürgen; and Dietmayer, Klaus: *Simultaneous Tracking and Shape Estimation with Laser Scanners*. In: *Proceedings of the 16th International Conference on Information Fusion*, pages 885–891, 2013.

[SADD14a] Schütz, Markus; Appenrodt, Nils; Dickmann, Jürgen; and Dietmayer, Klaus: *Multiple extended objects tracking with object-local occupancy grid maps*. In: *Proceedings of the 17th International Conference on Information Fusion*, 2014.

[SADD14b] Schütz, Markus; Appenrodt, Nils; Dickmann, Jürgen; and Dietmayer, Klaus: *Occupancy Grid Map-Based Extended Object Tracking*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2014.

[SBR+18] Scheidegger, Samuel; Benjaminsson, Joachim; Rosenberg, Emil; Krishnan, Amrit; and Granström, Karl: *Mono-Camera 3D Multi-Object Tracking Using Deep Learning Detections and PMBM Filtering.* In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 433–440, 2018.

[SBW08] Schuler, Karin; Becker, Denis; and Wiesbeck, Werner: *Extraction of Virtaul Scattering Centers of Vehicles by Ray-Tracing Simulations.* In: *IEEE Transactions on Antennas and Propagation*, volume 56, no. 11, pages 3543–3551, 2008.

[Sch13] Scheel, Alexander: *Laser-Based Multi-Target Tracking for Autonomous Vehicle Perception using Particle Filters.* Diploma Thesis, Fraunhofer Institute for Manufacturing Engineering and Automation, Stuttgart. Germany, 2013.

[SD19] Scheel, Alexander and Dietmayer, Klaus: *Tracking Multiple Vehicles Using a Variational Radar Model.* In: *IEEE Transactions on Intelligent Transportation Systems*, volume 20, no. 10, pages 3721–3736, 2019.

[SG99] Salmond, David J. and Gordon, Neil J.: *Group and Extended Object Tracking.* In: *Proceedings of the IEE Colloquium on Target Tracking: Algorithms and Applications*, pages 16/1–16/4, 1999.

[SGRD17] Scheel, Alexander; Gritschneder, Franz; Reuter, Stephan; and Dietmayer, Klaus: *Fusing Radar and Scene Labling Data for Multi-Object Vehicle Tracking.* In: *11. Workshop Fahrerassistenzsysteme und automatisiertes Fahren*, pages 11–20, 2017.

[Sit64] Sittler, Robert W.: *An Optimal Data Association Problem in Surveillance Theory.* In: *IEEE Transactions on Military Electronics*, volume 8, no. 2, pages 125–139, 1964.

[SKD+11] Steinemann, Philipp; Klappstein, Jens; Dickmann, Jürgen; Wünsche, Hans-Joachim; and von Hundelshausen, Felix: *Determining the Outline Contour of Vehicles in 3D-LIDAR-Measurements.* In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 479–484, 2011.

[SKD+12] Steinemann, Philipp; Klappstein, Jens; Dickmann, Jürgen; von Hundelshausen, Felix; and Wünsche, Hans-Joachim: *Geometric-Model-Free Tracking of Extended Targets Using 3D-LIDAR-Measurements.* In: *Proceedings SPIE 8379, Laser Radar Technology and Applications XVII*, pages 83790C-1–83790C-12, 2012.

[Sko02] Skolnik, Merrill I.: *Introduction to Radar Systems.* 3rd Revised edition, McGraw-Hill Education - Europe, London, 2002.

[SKRD16] Scheel, Alexander; Knill, Christina; Reuter, Stephan; and Dietmayer, Klaus: *Multi-Sensor Multi-Object Tracking of Vehicles Using High-Resolution Radars*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 558–565, 2016.

[SRD16] Scheel, Alexander; Reuter, Stephan; and Dietmayer, Klaus: *Using Separable Likelihoods for Laser-Based Vehicle Tracking with a Labeled Multi-Bernoulli Filter*. In: *Proceedings of the 19th International Conference on Information Fusion*, 2016.

[SRD17] Scheel, Alexander; Reuter, Stephan; and Dietmayer, Klaus: *Vehicle Tracking Using Extended Object Methods: An Approach for Fusing Radar and Laser*. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 231–238, 2017.

[SRKW16] Schlichenmaier, Johannes; Roos, Fabian; Kunert, Martin; and Waldschmidt, Christian: *Adaptive Clustering for Contour Estimation of Vehicles for High-Resolution Radar*. In: *Proceedings of the IEEE MTT-S International Conference on Microwaves for Intelligent Mobility*, 2016.

[SRW08] Schubert, Robin; Richter, Eric; and Wanielik, Gerd: *Comparison and Evaluation of Advanced Motion Models for Vehicle Tracking*. In: *Proceedings of the 11th International Conference on Information Fusion*, 2008.

[SRW14] Schuster, Michael; Reuter, Johannes; and Wanielik, Gerd: *Tracking of Vehicles on Nearside Lanes using multiple Radar Sensors*. In: *Proceedings of the International Radar Conference*, 2014.

[SRW15] Schuster, Michael; Reuter, Johannes; and Wanielik, Gerd: *Probabilistic Data Association for Tracking Extended Targets Under Clutter Using Random Matrices*. In: *Proceddings of the 18th International Conference on Information Fusion*, pages 961–968, 2015.

[SSH74] Singer, Robert A.; Sea, Ronald G.; and Housewright, Kim B.: *Derivation and Evaluation of Improved Tracking Filters for Use in Dense Multitarget Environments*. In: *IEEE Transactions on Information Theory*, volume IT-20, no. 4, pages 423–432, 1974.

[SSSW17] Schlichenmaier, Johannes; Selvaraj, Niranjan; Stolz, Martin; and Waldschmidt, Christian: *Template Matching for Radar-Based Orientation and Position Estimation in Automotive Scenarios*. In: *Proceedings of the IEEE MTT-S International Conference on Microwaves for Intelligent Mobility*, pages 95–98, 2017.

[SWBH12] Schueler, Kai; Weiherer, Tobias; Bouzouraa, Mohammed Essayed; and Hofmann, Ulrich: *360 Degree Multi Sensor Fusion for Static and Dynamic Obstacles*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 692–697, 2012.

[TG17] Thom, Markus and Gritschneder, Franz: *Rapid Exact Signal Scanning With Deep Convolutional Neural Networks*. In: *IEEE Transactions on Signal Processing*, volume 65, no. 5, pages 1235–1250, 2017.

[TLG08] Tzikas, Dimitris G.; Likas, Aristidis C.; and Galatsanos, Nikolaos P.: *The Variational Approximation for Bayesian Inference*. In: *IEEE Signal Processing Magazine*, volume 25, no. 6, pages 131–146, 2008.

[TSL00] Tenenbaum, Joshua B.; deSilva, Vin; and Langford, John C.: *A Global Geometric Framework for Nonlinear Dimensionality Reduction*. In: *Science*, volume 290, no. 5500, pages 2319–2323, 2000.

[VA09] Vu, Trung-Dung and Aycard, Olivier: *Laser-based Detection and Tracking Moving Objects using Data-Driven Markov Chain Monte Carlo*. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3800–3806, 2009.

[VDN15] Vatavu, Andrei; Danescu, Radu; and Nedevschi, Sergiu: *Stereovision-Based Multiple Object Tracking in Traffic Scenarios Using Free-Form Obstacle Delimiters and Particle Filters*. In: *IEEE Transactions on Intelligent Transportation Systems*, volume 16, no. 1, pages 498–511, 2015.

[VMB+15] Vo, Ba-Ngu; Mallick, Mahendra; Bar-Shalom, Yaakov; et al.: *Multi-target Tracking*. In: Webster, John G. (editor): *Wiley Encyclopedia of Electrical and Electronics Engineering*, Wiley, Chichester, UK, 2015.

[VSD03] Vo, Ba-Ngu; Singh, Sumeetpal; and Doucet, Arnaud: *Sequential Monte Carlo Implementation of the PHD Filter for Multi-target Tracking*. In: *Proceedings of the Sixth International Conference on Information Fusion*, pages 792–799, 2003.

[VSD05] Vo, Ba-Ngu; Singh, Sumeetpal; and Doucet, Arnaud: *Sequential Monte Carlo Methods for Multi-Target Filtering with Random Finite Sets*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 41, no. 4, pages 1224–1245, 2005.

[VV13] Vo, Ba-Tuong and Vo, Ba-Ngu: *Labeled Random Finite Sets and Multi-Object Conjugate Priors*. In: *IEEE Transactions on Signal Processing*, volume 61, no. 13, pages 3460–3475, 2013.

[VVC09] Vo, Ba-Tuong; Vo, Ba-Ngu; and Cantoni, Antonio: *The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and Its Implementations*. In: *IEEE Transactions on Signal Processing*, volume 57, no. 2, pages 409–423, 2009.

[VVPS10] Vo, Ba-Ngu; Vo, Ba-Tuong; Pham, Nam-Trung; and Suter, David: *Joint Detection and Estimation of Multiple Objects from Image Observations*. In: *IEEE Transactions on Signal Processing*, volume 58, no. 10, pages 5129–5241, 2010.

[VWN12] Vanpoperinghe, Elodie; Wahl, Martine; and Noyer, Jean-Charles: *Model-based detection and tracking of vehicle using a scanning laser rangefinder: a particle filtering approach*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 1144–1149, 2012.

[Wan93] Wang, Y. H.: *On the Number of Successes in Independent Trials*. In: *Statistica Sinica*, volume 3, no. 2, pages 295–312, 1993.

[WC17] Wyffels, Kevin and Campbell, Mark: *Precision Tracking via Joint Detailed Shape Estimation of Arbitrary Extended Objects*. In: *IEEE Transactions on Robotics*, volume 33, no. 2, pages 313–332, 2017.

[WHWK17] Wheeler, Tim A.; Holder, Martin; Winner, Hermann; and Kochenderfer, Mykel J.: *Deep Stochastic Radar Models*. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 47–53, 2017.

[Wil15a] Williams, Jason L.: *Marginal Multi-Bernoulli Filters: RFS Derivation of MHT, JIPDA, and Association-Based MeMBer*. In: *IEEE Transactions on Aerospace and Electronic Systems*, volume 51, no. 3, pages 1664–1687, 2015.

[Wil15b] Williams, Jason L.: *An Efficient, Variational Approximation of the Best Fitting Multi-Bernoulli Filter*. In: *IEEE Transactions on Signal Processing*, volume 63, no. 1, pages 258–273, 2015.

[Win16] Winner, Hermann: *Automotive RADAR*. In: Winner, Hermann; Hakuli, Stephan; Lotz, Felix; and Singer, Christina (editors): *Handbook of Driver Assistance Systems*. From the series *Springer reference*, pages 325–403, Springer, 2016.

[WÖ15] Wahlström, Niklas and Özkan, Emre: *Extended Target Tracking Using Gaussian Processes*. In: *IEEE Transactions on Signal Processing*, volume 63, no. 16, pages 4165–4178, 2015.

[Wv00] Wan, Eric A. and van der Merwe, Rudolph: *The Unscented Kalman Filter for Nonlinear Estimation*. In: *Proceedings of the IEEE Symposium on Adaptive Systems for Signal Processing, Communications, and Control*, 2000.

[XGSF18] Xia, Yuxuan; Granström, Karl; Svensson, Lennart; and Fatemi, Maryam: *Extended Target Poisson Multi-Bernoulli Filter*. 2018. arXiv: `1801.01353v1 [eess.SP]`.

Publications

Parts of this thesis were pre-published in author's papers.

As First Author

Scheel, Alexander; Granström, Karl; Reuter, Stephan; Meißner, Daniel; and Dietmayer, Klaus: *Tracking and Data Segmentation Using a GGIW Filter with Mixture Clustering.* In: *Proceedings of the 17th International Conference on Information Fusion*, 2014.

Scheel, Alexander; Knill, Christina; Reuter, Stephan; and Dietmayer, Klaus: *Multi-Sensor Multi-Object Tracking of Vehicles Using High-Resolution Radars.* In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 558–565, 2016.

Scheel, Alexander; Reuter, Stephan; and Dietmayer, Klaus: *Using Separable Likelihoods for Laser-Based Vehicle Tracking with a Labeled Multi-Bernoulli Filter.* In: *Proceedings of the 19th International Conference on Information Fusion*, 2016.

Scheel, Alexander; Gritschneder, Franz; Reuter, Stephan; and Dietmayer, Klaus: *Fusing Radar and Scene Labling Data for Multi-Object Vehicle Tracking.* In: *11. Workshop Fahrerassistenzsysteme und automatisiertes Fahren*, pages 11–20, 2017.

Scheel, Alexander; Reuter, Stephan; and Dietmayer, Klaus: *Vehicle Tracking Using Extended Object Methods: An Approach for Fusing Radar and Laser.* In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 231–238, 2017.

Scheel, Alexander and Dietmayer, Klaus: *Tracking Multiple Vehicles Using a Variational Radar Model.* In: *IEEE Transactions on Intelligent Transportation Systems*, volume 20, no. 10, pages 3721–3736, 2019.

As Co-Author

Granström, Karl; Reuter, Stephan; Meißner, Daniel; and Scheel, Alexander: *A multiple model PHD approach to tracking of cars under an assumed rectangular shape.* In: *Proceedings of the 17th International Conference on Information Fusion*, 2014.

Kunz, Felix; Nuss, Dominik; Wiest, Jürgen; Deusch, Hendrik; Reuter, Stephan; Gritschneder, Franz; Scheel, Alexander; Stübler, Manuel; Bach, Martin; Hatzelmann, Patrick; Wild, Cornelius; and Dietmayer, Klaus: *Autonomous Driving at Ulm University: A Modular, Robust, and Sensor-Independent Fusion Approach.* In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 666–673, 2015.

Reuter, Stephan; Scheel, Alexander; and Dietmayer, Klaus: *The Multiple Model Labeled Multi-Bernoulli Filter.* In: *Proceddings of the 18th International Conference on Information Fusion*, pages 1574–1580, 2015.

Beard, Michael; Reuter, Stephan; Granström, Karl; Vo, Ba-Tuong; Vo, Ba-Ngu; and Scheel, Alexander: *A Generalised Labelled Multi-Bernoulli Filter for Extended Multi-target Tracking.* In: *Proceddings of the 18th International Conference on Information Fusion*, pages 991–998, 2015.

Reuter, Stephan; Scheel, Alexander; Beard, Michael; Granström, Karl; Vo, Ba-Tuong; Vo, Ba-Ngu; and Dietmayer, Klaus: *Environment Perception for Companion Systems.* In: *Proceedings of the 1st International Symposium on Companion-Technology*, pages 68–72, 2015.

Beard, Michael; Reuter, Stephan; Granström, Karl; Vo, Ba-Tuong; Vo, Ba-Ngu; and Scheel, Alexander: *Multiple Extended Target Tracking With Labeled Random Finite Sets.* In: *IEEE Transactions on Signal Processing*, volume 64, no. 7, pages 1638–1653, 2016 .

Knill, Christina; Scheel, Alexander; and Dietmayer, Klaus: *A Direct Scattering Model for Tracking Vehicles with High-Resolution Radars.* In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 298–303, 2016.

Hirscher, Tobias; Scheel, Alexander; Reuter, Stephan; and Dietmayer, Klaus: *Multiple Extended Object Tracking Using Gaussian Processes.* In: *Proceedings of the 19th International Conference on Information Fusion*, 2016.

Reuter, Stephan; Danzer, Andreas; Stübler, Manuel; Scheel, Alexander; and Granström, Karl: *A Fast Implementation of the Labeled Multi-Bernoulli Filter*

Using Gibbs Sampling. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 765–772, 2017.

Reuter, Stephan; Scheel, Alexander; Geier, Thomas; and Dietmayer, Klaus: *Environment Adaption for Companion-Systems*. In: Biundo, Susanne and Wendemuth, Andreas (editors): *Companion Technology*. From the series *Cognitive Technologies*, pages 301–320, Springer International Publishing and Imprint: Springer, Cham, 2017.

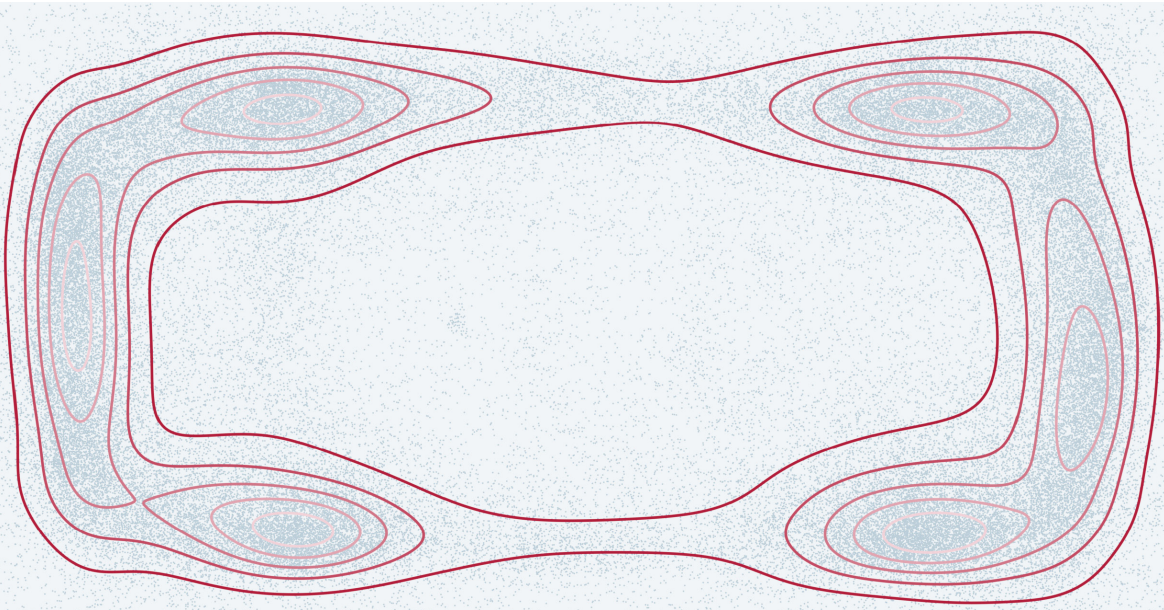
Supervised Theses

Brendle, Johannes: *Entwurf und Realisierung einer generischen Anwendung zur Markierung von Sensordaten.* Bachelor Thesis, Ulm University, Ulm, Germany, 2014.

Knill, Christina: *Extended Object Tracking for Vehicle Environment Perception Using High-Resolution Radars.* Master Thesis, Ulm University, Ulm, Germany, 2015.

Hirscher, Tobias: *Tracking mehrerer ausgedehnter Objekte mit Gauß-Prozessen.* Master Thesis, Ulm University, Ulm, Germany, 2016.

König, Markus: *Echtzeitfähige Verkehrszeichenerkennung für ein autonomes Modellfahrzeug.* Bachelor Thesis, Ulm University, Ulm, Germany, 2017.



ISBN 978-3-941543-45-4

ISBN 978-3-941543-46-1 (E-Book)