

Automotive Target Models for Point Cloud Sensors

Dissertation
for the award of the degree
“Doctor rerum naturalium” (Dr.rer.nat.)
at the Georg-August Universität Göttingen

within the doctoral program Ph.D. Programme in Computer Science (PCS)
of the Georg-August University School of Science (GAUSS)

submitted by
Hauke Kaulbersch

Göttingen, 2022

Thesis Advisory Committee

Prof. Dr.-Ing. Marcus Baum

Institute of Computer Science, Georg-August-Universität Göttingen, Germany

Dr. Jens Honer,

Driving Assistance Research Group, Valeo Schalter und Sensoren Bietigheim-Bissingen, Germany

Dr. Heinrich Gotzig,

Driving Assistance Product Group, Valeo Schalter und Sensoren Bietigheim-Bissingen, Germany

Members of the Examination Board

Referee: Prof. Dr.-Ing. Marcus Baum
Institute of Computer Science,
Georg-August-Universität Göttingen, Germany

Co-referee: Prof. Dr.-Ing. Alois Christian Knoll,
Institute of Computer Science,
Technische Universität München, Germany

Further Members of the Examination Board

Prof. Dr. Dieter Hogrefe,
Institute of Computer Science, Georg-August-Universität Göttingen, Germany

Prof. Dr. Carsten Damm,
Institute of Computer Science, Georg-August-Universität Göttingen, Germany

Prof. Dr. Fabian Sinz,
Institute of Computer Science, Georg-August-Universität Göttingen, Germany

Dr. Jens Honer,
Driving Assistance Research Group, Valeo Schalter und Sensoren Bietigheim-Bissingen, Germany

Date of the Oral Examination

10.12.2021

I hereby declare that I have written this thesis independently without any help from others and without the use of documents or aids other than those stated. I have mentioned all used sources and cited them correctly according to established academic citation rules.

Göttingen, 12.01.2020

Acknowledgment

Ich möchte mich an dieser Stelle einmal bei verschiedenen Gruppen für ihre Unterstützung während der letzten Jahre bedanken. Zuerst möchte ich meinen Dank an meine beiden Hauptbetreuer Professor Dr. Marcus Baum und Dr. Jens Honer aussprechen. Beide haben mich über den Zeitraum meiner Promotion mit ihrer Expertise, ihrem Feedback und diversen Diskussionen sehr unterstützt.

Weiterer Dank geht an Professor Dr. Alois Knoll als Gutachter und das Thesis Komitee für ihre Zeit und Bereitschaft der Verteidigung dieser Arbeit beizuwohnen.

Hinzu möchte ich meinen Dank an Dr. Heinz Gotzig als Mitglied meines Betreuungskomitees von Seite der Firma aussprechen. Ich bedanke mich auch bei allen Kollegen für die Zusammenarbeit und die Erfahrung die ich durch euch erhalten habe, im Speziellen Tobias Hirscher und Fabian Schmieder.

Weiterer Dank geht an die anderen Promotionsstudenten des Data Fusion Lab Göttingen: Shishan Yang, Fabian Sigges, Kolja Thormann und Laura Wolf. Speziell Kolja und Laura danke ich für das Feedback zu dieser Arbeit.

Zu guter Letzt bedanke ich mich bei meiner Familie und meinen Freunden für ihre konstante moralische Unterstützung dieser Zeit und insbesondere für die letzten anderthalb Corona Jahre.

Abstract

One of the major challenges to enable automated driving is the perception of other road users in the host vehicle's vicinity. Various automotive sensors that provide detailed information about other traffic participants have been developed to handle this challenge. Of particular interest for this work are Light Detection and Ranging (LIDAR) and Radio Detection and Ranging (RADAR) sensors, which generate multiple, spatially distributed, noise corrupted point measurements on other traffic participants. Based on these point measurements, the traffic participant's kinematic and shape parameters have to be estimated.

The choice of a suitable extent model is paramount to accurately track a target's position, orientation and other parameters. How well a model performs typically depends on the type of target that has to be tracked, e.g. pedestrians, bikes or cars, as well as the sensor's setup and measurement principle itself. This work considers the creation of extended object models and corresponding inference strategies for tracking automotive vehicles based on accumulated point cloud data.

We gain insights into the extended object model's requirements by analysing automotive LIDAR and RADAR sensor data. This analysis aids in the identification of relevant features from the measurement's spatial distribution and their incorporation into an accurate target model. The analysis lays the foundation for our main contributions.

We developed a constrained Spline-based geometric representation and a corresponding inference strategy for the contour of cars in LIDAR data.

We further developed a heuristic to account for the integration of the measurement distribution on cars, generated by LIDAR sensors mounted on the roof of the recording vessel.

Last, we developed an extended target model for cars based on automotive RADAR sensors. The model provides an interpretation of a learned Gaussian Mixture Model (GMM) as scatter sources and uses the Probabilistic Multi-Hypothesis Tracker (PMHT) to formulate a closed form Maximum a Posteriori (MAP) update.

All developed approaches are evaluated on real world data sets.

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 1 |
| 1.1 | Motivation | 1 |
| 1.2 | Problems And Contributions | 2 |
| 1.3 | Structural Outline Of The Thesis | 5 |
| 2 | Foundations | 7 |
| 2.1 | Bayesian Inference | 7 |
| 2.2 | Kalman Filter | 9 |
| 2.3 | Nonlinear Kalman Filter | 10 |
| 2.3.1 | Extended Kalman Filter | 10 |
| 2.3.2 | Unscented Kalman Filter | 11 |
| 2.4 | Extended Target Tracking | 12 |
| 2.4.1 | State Of The Art Extended Target Models | 13 |
| 2.4.2 | Update Strategies For Extended Targets | 15 |
| 2.4.3 | PMHT Multi Source Extended Target Update | 16 |
| 2.5 | Dynamic Models | 19 |
| 2.6 | Automotive Point Cloud Sensors | 20 |
| 2.7 | Automotive Point Cloud Data Sets | 21 |
| 2.8 | Metric for Extended Target Trackers Based On Bounding Box Annotations | 23 |
| 3 | Point Cloud Sensor Data Analysis | 25 |
| 3.1 | Problem Setting | 25 |
| 3.2 | State Of The Art Analysis Methods | 26 |
| 3.3 | Data Preparation And Analysis Strategy | 27 |
| 3.4 | Representative Data Sets | 29 |

| | | |
|----------|--|------------|
| 3.5 | Point Cloud Analysis | 31 |
| 3.5.1 | Bounding Box Size Distribution | 31 |
| 3.5.2 | Measurement Distribution | 34 |
| 3.6 | Discussion Of Target Types And Model Requirements | 38 |
| 4 | LIDAR Extent Modelling And Tracking | 41 |
| 4.1 | Problem Setting | 41 |
| 4.2 | State Of The Art: LIDAR Extended Target Models | 42 |
| 4.3 | A Spline Based Automotive Contour Model | 43 |
| 4.3.1 | B-Spline Model | 43 |
| 4.3.2 | Free B-Spline Model | 45 |
| 4.3.3 | Free B-Spline Update With The PMHT | 47 |
| 4.3.4 | A Constraint B-Spline Contour Function | 49 |
| 4.3.5 | Surface Point Association | 51 |
| 4.3.6 | Spline Model: Jacobian | 54 |
| 4.4 | Modelling the Measurement Distribution With Asymmetric Noise | 58 |
| 4.4.1 | Distribution Analysis For Radial Random Hypersurface Models | 58 |
| 4.4.2 | Proposed Asymmetric Noise Model | 59 |
| 4.4.3 | Application Of The Asymmetric Noise To Spline Target Model | 61 |
| 4.5 | Evaluation | 62 |
| 4.5.1 | Bumper Mounted LIDAR Experiments | 62 |
| 4.5.2 | Roof Mounted LIDAR Experiments | 67 |
| 4.6 | Discussion Of The Spline Model And The Asymmetric Noise Assumption For Cars In LIDAR Data | 75 |
| 5 | Radar Extent Modelling and Tracking | 77 |
| 5.1 | Problem Setting | 77 |
| 5.2 | State Of The Art: Radar Extended Target Models | 79 |
| 5.3 | Gaussian Mixture As An Extended Target Model | 81 |
| 5.4 | Learned Gaussian Mixture Model | 82 |
| 5.5 | Extent Model And PMHT Update | 83 |
| 5.6 | Evaluation | 86 |
| 5.6.1 | Single Target Evaluation | 87 |
| 5.6.2 | Large Scale Evaluation | 92 |
| 5.7 | Discussion Of The GMM_PMHT | 95 |
| 6 | Conclusion And Future Work | 97 |
| | Own Publications | 100 |
| | Bibliography | 102 |

List of Figures

| | | |
|-----|--|----|
| 1.1 | Example point clouds from Light Detection and Ranging (LIDAR) and Radio Detection and Ranging (RADAR) sensors. | 3 |
| 2.1 | Hidden Markov Model (HMM) of the typical state transition in recursive Bayesian estimation. The boxes describe hidden variables and the circles describe observed variables. | 8 |
| 2.2 | HMM for extended target tracking. | 13 |
| 2.3 | Measurement generation scenarios in extended target tracking. | 14 |
| 2.4 | A typical radial contour function. | 15 |
| 2.5 | Illustration of the Gaussian Wasserstein metric for rectangular extended targets. | 24 |
| 3.1 | Example of the coordinate transformation on a single frame of the nuScenes data set. | 28 |
| 3.2 | Illustrations of the aspect angle and the conditional ranges. | 29 |
| 3.3 | Histograms for pedestrian bounding box sizes. | 32 |
| 3.4 | Histograms for bicycle bounding box sizes. | 33 |
| 3.5 | Histograms for car bounding box sizes. | 33 |
| 3.6 | Accumulated data for the nuScenes data set. | 35 |
| 3.7 | Accumulated data for the Lyft level 5 data set. | 36 |
| 3.8 | Additional histograms for every LIDAR sensor in logarithmic scale for the marginal over the accumulated data for cars and bicycles. | 37 |
| 4.1 | Example of a closed quadratic B-spline. | 44 |
| 4.2 | Illustration of the basis functions | 46 |
| 4.3 | Example update for the free spline model using the Probabilistic Multi-Hypothesis Tracker (PMHT) | 48 |

| | | |
|------|---|----|
| 4.4 | Example of a minimal spline model representing the desired car model. | 49 |
| 4.5 | Logarithmically scaled heat map of all LIDAR points from cars in the nuScenes dataset in unit coordinates. | 59 |
| 4.6 | Histograms of the scaling factor s_c^z and the squared scaling factor $s_c^{z^2}$ | 60 |
| 4.7 | Comparison of modelling prior knowledge with spline and Gaussian process models. | 63 |
| 4.8 | Scaling differences in hand tuned Gaussian process with constrained shape but isotropic scaling and the spline model with anisotropic scaling. | 65 |
| 4.9 | Plot of every 4th update for spline and the Gaussian process model on the straight trajectory. | 68 |
| 4.10 | State parameters and their difference. Parameters of the spline model in blue, parameters of the Gaussian process in orange and the difference between them in green. | 68 |
| 4.11 | Measurements association and innovation log likelihood in the straight trajectory. | 68 |
| 4.12 | Plot of every 4th update for spline and the Gaussian process model on the curved trajectory. | 69 |
| 4.13 | State parameters and their difference in the curved trajectory. | 69 |
| 4.14 | Measurements association and innovation log likelihood in the curved trajectory. | 69 |
| 4.15 | Initialization effects for asymmetric noise. | 71 |
| 4.16 | Example frame from the nuScenes dataset illustrating the typical estimation bias. | 72 |
| 4.17 | Results over the entire batch of traces. | 74 |
| | | |
| 5.1 | Illustration of the representative spaces and measurement generation relations for extended target tracking with discrete measurement sources. | 79 |
| 5.2 | Illustration of the representative spaces and the measurement generation relations for multi target tracking. | 80 |
| 5.3 | Example Gaussian Mixture Model (GMM) densities over the histogram of the used training data consisting of 30x30 bins. | 83 |
| 5.4 | Schematic illustration of the update for the PMHT model. | 87 |
| 5.5 | Ground truth annotation data for the four picked scenarios in global coordinates. | 88 |
| 5.6 | Illustration of a RADAR algorithms on a subsequence of all updates. | 90 |
| 5.7 | Wasserstein distances for the key-frames of all scenarios. [1]©2019IEEE | 92 |
| 5.8 | Performance evaluation for different numbers of PMHT iterations. | 93 |
| 5.9 | Statistics over the compared algorithms. Shown are mean, median and 95th percentile | 94 |
| | | |
| 6.1 | Example accumulation for trucks in logarithmic scale. | 98 |

List of Tables

| | | |
|-----|--|----|
| 2.1 | Available automotive data sets. | 21 |
| 3.1 | General statistics of the annotation bounding boxes length and width. | 32 |
| 3.2 | Number of Frames per target and sensor type for both of the selected data sets. . . | 34 |
| 5.1 | Optimised parameters for the noise parameters σ_α and σ_a for both target models. . | 93 |
| 5.2 | Summary of the Wasserstein distance (WSD) statistics over all 650 traces. | 95 |

Nomenclature

| | |
|---------------------------------|---|
| a | Scalar |
| \mathbf{x} | Column vector |
| \mathbf{P} | Matrix |
| $\cos(a)$ | Cosine of a |
| $\sin(a)$ | Sine of a |
| $\arctan2(x_2, x_1)$ | Arctangent function |
| $\ \mathbf{x}\ _2$ | Euclidean norm of \mathbf{a} |
| $\mathbf{x} \cdot \mathbf{y}$ | Dot product between \mathbf{x} and \mathbf{y} |
| $\mathbf{x} \wedge \mathbf{y}$ | Cross product between \mathbf{x} and \mathbf{y} |
| $\mathbf{x} \otimes \mathbf{y}$ | Kronecker product between \mathbf{x} and \mathbf{y} |
| \mathbf{I}_d | Identity matrix with dimension d |
| $\mathbf{0}_d$ | column vector of length d , containing zeros |
| $p(x)$ | Probability density function of x |
| $\sphericalangle(\mathbf{x})$ | Angle of a given vector $\arctan2(x_2, x_1)$ |
| $\text{diag}(\mathbf{a})$ | Diagonal matrix with main diagonal consisting of the elements of \mathbf{a} |

| | |
|---|--|
| $\mathcal{N}(\hat{\mathbf{x}}, \mathbf{P})$ | Gaussian probability density function with mean $\hat{\mathbf{x}}$ and covariance \mathbf{P} |
| $\mathcal{N}(\mathbf{z}, \hat{\mathbf{x}}, \mathbf{P})$ | Likelihood of measurement \mathbf{z} for a Gaussian with mean $\hat{\mathbf{x}}$ and covariance \mathbf{P} |
| $p(\mathbf{x}, \mathbf{z})$ | Joint probability density function of \mathbf{x} and \mathbf{z} |
| $p(\mathbf{x} \mathbf{z})$ | Conditional probability density function of \mathbf{x} given \mathbf{z} |
| π | Pi |

List Of Abbreviations

| | | |
|--------------|--|----|
| AD | Automated Driving | 1 |
| ADAS | Advanced Driver Assistant Systems | 1 |
| EKF | Extended Kalman Filter | 10 |
| EM | Expectation Maximization | 17 |
| ETT | Extended Target Tracking | 12 |
| FMCW | Frequency Modulated Continuous Wave | 21 |
| GGIW | Gamma Gauss Inverse Wishart | 14 |
| GLMB | Generalized Labeled Multi-Bernoulli | 42 |
| GMM | Gaussian Mixture Model | 4 |
| HMM | Hidden Markov Model | 8 |
| IoU | Intersection over Union | 24 |
| LIDAR | Light Detection and Ranging | 1 |
| MAP | Maximum a Posteriori | 5 |
| MSE | Mean Squared Error | 23 |
| NURBS | non-uniform rational basis spline | 43 |
| OSPA | Optimal Subpattern Assignment | 24 |
| PDF | Probability Density Function | 8 |
| PHD | Probability Hypothesis Density | 42 |
| PMHT | Probabilistic Multi-Hypothesis Tracker | 5 |
| RADAR | Radio Detection and Ranging | 1 |
| RHM | Random Hypersurface Model | 14 |
| RM | Random Matrix | 14 |
| SAE | Society of Automotive Engineers | 42 |
| SUV | Sport Utility Vehicle | 26 |

| | | |
|------------|--|----|
| UKF | Unscented Kalman Filter | 10 |
| VGM | Variational Gaussian Mixture model | 80 |
| WSD | Wasserstein distance | 74 |

Contents

| | |
|--|---|
| 1.1 Motivation | 1 |
| 1.2 Problems And Contributions | 2 |
| 1.3 Structural Outline Of The Thesis | 5 |

1.1 Motivation

A topic of increasing interest in the automotive industry is the advancement to secure autonomous driving systems. This interest gave rise to Automated Driving (AD) and Advanced Driver Assistant Systems (ADAS). These systems are informing or warning the driver, or are taking control of some function of the vehicles, with respect to their level of automation. They are required to ensure a safe interaction with the vehicle's environment. Therefore, a major challenge that needs to be accounted for is the perception of the environment and, in specific, other traffic participants in the host vehicle's vicinity. To get information about the environment, a set of sensors have been developed. The most commonly used sensor systems are Ultrasonic, Radio Detection and Ranging (RADAR) and Light Detection and Ranging (LIDAR) sensors, and also cameras. Each of these sensors uses different measurement principles, leading to different information about the environment. Cameras provide images with varying resolution and color channels, the Ultrasonic sensor provides distance measurements, and RADAR and LIDAR sensors in general are point cloud sensors. Here RADAR sensors are desirable as it is less affected by weather conditions like rain, while LIDAR sensors provide a generally higher resolution with less noise. Each of these data types requires a tailored approach to allow for the detection and tracking of other traffic participants. The focus in this work is put on the point cloud sensors.

Modern LIDAR and RADAR sensor systems have an increasingly high resolution, which in turn provides a relatively large number of point measurements from the environment. This high resolution allows to receive multiple spatially distributed measurements from a single target as it can occupy multiple resolution cells. This stands in contrast to classical sensor systems. Air traffic RADARs, for example, provide at most a single measurement from a target. The additional information about the target's spatial occupancy in modern RADAR and LIDAR allows for the estimation of the target's shape as well as the kinematic parameters of a point target. This problem is typically referred to as the extended target tracking problem [MCS⁺14,GBR17a].

The choice of a suitable extended target model is of most importance to find a good estimate of a target's dimensions. It has to account for the measurement distribution on the target, prior knowledge about the target's shape, and measurement noise properties.

The properties of these measurements vary strongly in the respective aforementioned point cloud sensors. LIDAR sensors provide a larger amount of measurements distributed on the 3d contour of the target per time step. In contrast to that, current RADAR sensors provide only a limited number of planar measurements from material dependent scattering sources. Exemplary data of single frames from the respective sensor data in the nuScenes data set [CBL⁺19] are provided in Figure 1.1.

We will focus on the data driven development of suitable extended target models for tracking cars based on modern automotive LIDAR and RADAR point cloud sensor systems.

1.2 Problems And Contributions

This thesis considers the problem of extended target tracking for automotive point cloud sensor systems. The focus is specifically centered on the extended target models with respect to different point cloud sensors and their setup dependent incorporation of prior knowledge. Here a special interest is put on the measurement distribution and sizes of traffic participants as prior knowledge and the development of closed form inference strategies.

Suitable requirements for extended target models are often stated on general assumptions, e.g. that measurements from cars are generated from a 2d rectangular contour [GLO11,BRAD16] of the target or radially defined shape models with symmetry constraints [WÖ15,HSRD16], that measurements are generally following a uniform distribution on their volume or that the target's extent may be described by a general Gaussian distribution. Further works on the other hand have analysed the measurement distribution of traffic participants for single targets or a small set of recordings of multiple targets. These revealed general scan patterns and assumptions, such as approximately rectangular shapes for cars in LIDAR data [GLO11], a scatter source behaviour [BML⁺17,SD18] for cars in RADAR data and elliptical representations for bicycles in both sensor types [GLO11,HDKD18].

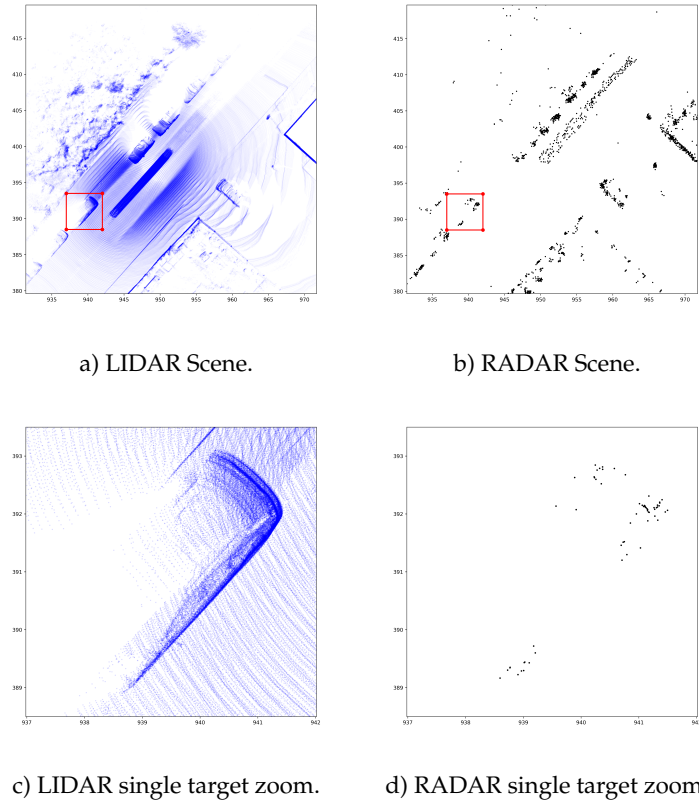


Figure 1.1: Example point clouds measured by LIDAR and RADAR sensors on a randomly picked scene of the nuScenes data set [CBL⁺19]. The points are accumulated over 3 seconds. The lower figures show the zoom on a single car, to illustrate the scattering data a single target generates.

We start with a minor contribution, by performing an analysis of point cloud sensor data on large publicly available point cloud data sets in Chapter 3. We specifically seek to confirm the measurement distributions found in the small scale or single target analysis and find further details that can be integrated into an extended target model, for pedestrians, bicycles, and cars. The analysis is based on the LIDAR and RADAR data from the nuScene [CBL⁺19] and LIDAR data Lyft level 5 [KUH⁺19] data set. The typical assumptions for pedestrians and bicycles yield for all sensor types, thus, they are not further considered in this thesis.

For cars in LIDAR data a general box with distinctly rounded corners can be determined. We further note that an anisotropic scaling of length and width is desired due to the size distribution of the analysed vehicles. These observations significantly differ from the corners of a rectangular bounding box approximation [GLO11, BRAD16]. Radially defined contour models with symmetry constraints [WÖ15, HSRD16] can handle these rounded corners well but they are not easily

constrained to a prior shape while introducing anisotropic scaling. Determining a model that contains the detailed shape features while providing anisotropic scaling would be desirable.

We further notice the influence of the mounting position of the LIDAR sensor on the measurement distribution on the car. Sensors mounted on the height of the recording vessels bumper mostly capture the 2d contour of the target car. For LIDAR sensors mounted on the roof, on the other hand, a complex distribution is visible, measurements are mostly still generated from the target's 2d boundary but a smaller but significant amount of measurements stems from the volume. Efficiently including this measurement distribution into the model would further be desirable, as typical scenarios assume measurements to stem purely from the contour or to be uniformly distributed over the complete volume of the target [GBR17b].

We further confirm the scatter source behaviour of RADAR data on a large scale. We note that an extended target model was proposed in [SD18]. The work incorporates the complex measurement generation based on Gaussian Mixture Model (GMM) densities learned on a large amount of recorded data from a single vehicle. This model performed an update using particle filter methods, in which the density served as a likelihood function for the particles themselves. This method, however, depends on sampling methods and suffers from the curse of dimensionality [BBL⁺08]. Thus, avoiding the particle filter would be desirable.

The research questions arising can be stated as:

1. How can the contour structure of cars in LIDAR data be effectively incorporated into the extended target model with a suitable inference strategy, while constraining the shape changes to anisotropic scaling?
2. How can the structured distribution of measurements be efficiently incorporated into the extended target model?
3. How can the complex distribution of scatter sources in RADAR data be incorporated into an extended target model with closed form updates instead of particle filter methods?

The contributions for these questions are:

1. A Cartesian defined B-spline contour model for the tracking of cars and the statement of a closed form update based on a radially defined measurement model. The focus of this contribution lies on the introduction of the rounded corners as prior known shape features, while constraining the shape changes to anisotropic scaling. This contribution is based on [2].
2. An asymmetric noise heuristic for the incorporation of the measurement distribution on the target's contour and volume based on the radially defined contour of a prior target estimate. This contribution is based on [3].
3. We provide an interpretation of a learned GMM as an extent representation with scatter sources. We relate the scatter source setup to multi target tracking and state a closed form

update based on the relation. We specifically use a Maximum a Posteriori (MAP) update with the Probabilistic Multi-Hypothesis Tracker (PMHT) for extended targets. This contribution is based on [1] and [4].

1.3 Structural Outline Of The Thesis

The remainder of this thesis is structured as follows. We start with the foundations in Chapter 2, in which we lay out Bayesian inference and extended target tracking. We further investigate available automotive point cloud sensors and corresponding data sets and provide a suitable metric for the corresponding annotations. Thereafter, we move to the data analysis of typical automotive point cloud sensors in Chapter 3, which lays out the structure of measurement on different target types and cars in specific, based on which we move on to extended target models. For these we have two sections which consider LIDAR and RADAR models separately, with respect to the previous analysis of the point cloud distribution. In Chapter 4 we provide an extended target contour model for LIDAR data based on Cartesian defined B-splines. The chapter also includes the asymmetric noise heuristic to incorporate the distribution of measurements that is generated by a typical roof mounted LIDAR sensor. In Chapter 5 we provide a suitable closed form update for a learned Gaussian mixture based extended target model by exploiting the relation of extended and multi target tracking. Finally, we provide a conclusion and an outlook as well as possible future work in Chapter 6.

Contents

| | | |
|-----|---|----|
| 2.1 | Bayesian Inference | 7 |
| 2.2 | Kalman Filter | 9 |
| 2.3 | Nonlinear Kalman Filter | 10 |
| 2.4 | Extended Target Tracking | 12 |
| 2.5 | Dynamic Models | 19 |
| 2.6 | Automotive Point Cloud Sensors | 20 |
| 2.7 | Automotive Point Cloud Data Sets | 21 |
| 2.8 | Metric for Extended Target Trackers Based On Bounding Box Annotations | 23 |

This chapter is meant to lay the foundations for all contributions of this thesis. We start with the basics to Bayesian tracking in Section 2.1 and a representation of linear and non-linear Kalman filtering methods in Section 2.2 and Section 2.3. After this short review of filtering methods, we continue with an introduction to extended target tracking and modelling in Section 2.4. We further dive into suitable motion models for tracking vehicles in Section 2.5 and provide a small list of available point cloud sensor types in Section 2.6 and public data sets that feature these sensor types in Section 2.7. We end the chapter with a short overview of metrics for the evaluation of extended targets in Section 2.8, where we focus on annotation bounding boxes which are typically used in automotive data sets.

2.1 Bayesian Inference

The aim in Bayesian inference is to provide an estimate of a target's true unobservable state \mathbf{x}_l by incorporating the information provided by a new measurement \mathbf{z}_k [Koc16].

The estimate in Bayesian estimation is generally represented as a Probability Density Function (PDF) $p(\mathbf{x}_l|\mathbf{z}_k)$ which incorporates all information available on time step l based on all measurements from time step 0 to time step k . The time varying relation between the state and the measurements can be described by a Hidden Markov Model (HMM) where the state \mathbf{x}_k at time k is a hidden variable that changes with transition function $f(\cdot)$ and emits dependent observations \mathbf{z}_k at time k with the measurement function $h(\cdot)$. A typical representation of the true state's behavior and the relation to observed measurements can be found in Figure 2.1.

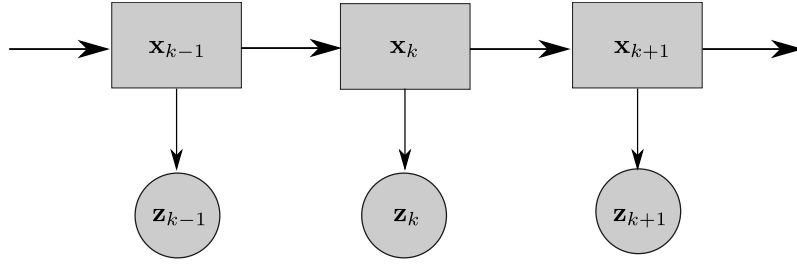


Figure 2.1: HMM of the typical state transition in recursive Bayesian estimation. The boxes describe hidden variables and the circles describe observed variables.

From this Markov process follows the application of the Markov property, such that each state is only dependent on the last state

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}, \dots, \mathbf{x}_0, \mathbf{z}_{k-1}, \dots, \mathbf{z}_0) = p(\mathbf{x}_k|\mathbf{x}_{k-1}) \quad (2.1)$$

and the assumption of conditional independence of measurements given the state on one time step

$$p(\mathbf{z}_k|\mathbf{x}_{k-1}, \dots, \mathbf{x}_0, \mathbf{z}_{k-1}, \dots, \mathbf{z}_0) = p(\mathbf{z}_k|\mathbf{x}_k) . \quad (2.2)$$

For the estimation of the hidden variables based on the observed variables, three steps for updating the knowledge about the state are generally available in the framework Bayesian.

The propagation of the estimated PDF from time $k - 1$ to time k is referred to as the prediction. This procedure is used for $l > k$ and by the use of the Markov process it can be formulated as the propagation of our PDF to the next time step, as no further measurements will be available between k and l , such that $l = k + 1$. Thus, the prediction can be performed via the Chapman-Kolmogorov equation [Koc16], such that

$$p(\mathbf{x}_{k+1}|\mathbf{z}_k, \dots, \mathbf{z}_0) = \int p(\mathbf{x}_{k+1}|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_k), d\mathbf{x}_k . \quad (2.3)$$

The incorporation of new measurements into the estimate is referred to as filtering or update and provides the integration of the information of new measurements on time step k , such that $l = k$.

This step is performed using the Bayes rule [Koc08] such that

$$p(\mathbf{x}_k | \mathbf{z}_k) = \frac{p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{z}_{k-1})}{\int p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{z}_{k-1}) d\mathbf{x}_k} . \quad (2.4)$$

Here $p(\mathbf{z}_k | \mathbf{x}_k)$ states the measurement likelihood.

The backwards propagation of knowledge from measurements is the optional final step in Bayesian tracking and is referred to as smoothing or retrodiction to update states in the past with information from future measurements. This back-propagation is generally performed using the prediction and filtering steps of future time steps. This propagation may be stated recursively, such that for $l < k$

$$p(\mathbf{x}_l | \mathbf{z}_k) = \int \frac{p(\mathbf{x}_{l+1} | \mathbf{x}_l) p(\mathbf{x}_l | \mathbf{z}_l)}{p(\mathbf{x}_{l+1} | \mathbf{z}_l)} p(\mathbf{x}_{l+1} | \mathbf{z}_k) d\mathbf{x}_l . \quad (2.5)$$

This approach is specifically useful for the estimation on a batch of time steps or offline estimation of entire trajectories.

2.2 Kalman Filter

The Kalman filter [Kal60] is the optimal Bayesian estimator for linear systems in which all random variables are assumed to be Gaussian distributed.

Thus, the problem may be state based on a process model with a linear transition function

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{v}_k \quad (2.6)$$

and respective linear measurement function

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{w}_k , \quad (2.7)$$

where the transition matrix \mathbf{F}_k and measurement matrix \mathbf{H}_k express the linear relation of the state propagation from time k to $k + 1$ and of measurement and state on time step k , respectively. The process noise \mathbf{v}_k and measurement noise \mathbf{w}_k are both assumed to be zero mean Gaussian white noise, such that $\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{R}_k)$ and $\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{Q}_k)$.

Further assuming our estimate \mathbf{x}_k on time step k to be another Gaussian distribution with $\mathbf{x}_k \sim \mathcal{N}(\hat{\mathbf{x}}_k, \mathbf{P}_k)$ yields a conjugate prior and results in the Kalman filter with the prediction equations

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_k \hat{\mathbf{x}}_{k-1|k-1} , \quad (2.8)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{Q}_k , \quad (2.9)$$

and update equations

$$\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k , \quad (2.10)$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \mathbf{S}_k^{-1} , \quad (2.11)$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}) , \quad (2.12)$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{H}_k \mathbf{P}_{k|k-1} . \quad (2.13)$$

These provide the best linear unbiased estimator. The proof may be found in the original work [Kal60].

2.3 Nonlinear Kalman Filter

In the case of nonlinear transition or generation models, the Kalman filter can not be applied directly as the nonlinearity, and related state dependency introduces arbitrary errors in the estimation. One of three commonly used approaches to handle these nonlinearities can be deployed: the Extended Kalman Filter (EKF), the Unscented Kalman Filter (UKF), and the particle filter. We will omit the particle filter as it is not required for the contributions in this thesis, but it should be mentioned for the sake of completeness.

The nonlinear transition function is typically described as

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k) + \mathbf{w}_k , \quad (2.14)$$

whereas, the measurement function can be stated as

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{v}_k . \quad (2.15)$$

2.3.1 Extended Kalman Filter

The Extended Kalman filter linearises the non-linear functions h and f with respect to the current state. While the linearisation removes any provable optimality, it is deployed in many applications.

The standard equations for the EKF with time index k are stated with the prediction equations

$$\hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}) , \quad (2.16)$$

$$\mathbf{P}_{k|k-1} = F_k \mathbf{P}_{k-1|k-1} F_k^T + \mathbf{Q}_k , \quad (2.17)$$

and the update equations

$$\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k , \quad (2.18)$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \mathbf{S}_k^{-1} , \quad (2.19)$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - h(\hat{\mathbf{x}}_k)) , \quad (2.20)$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} + \mathbf{K}_k \mathbf{H}_k \mathbf{P}_{k|k-1} , \quad (2.21)$$

in which the EKF uses the first order Taylor-series expansion to provide the Jacobians

$$\mathbf{H}_k = \left. \frac{\partial h}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_{k|k-1}} , \quad (2.22)$$

$$\mathbf{F}_k = \left. \frac{\partial f}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_{k|k}} . \quad (2.23)$$

We note that higher order Taylor series expansions can be used if appropriate.

2.3.2 Unscented Kalman Filter

The unscented Kalman filter uses a set of specific weighted samples referred to as sigma points $\{\mathcal{X}_i\}_{i=0}^{2L}$ and the actual mean. Here, L describes the state's dimension such that two sigma points are chosen per dimension. These are propagated through the nonlinear function, and the covariance and mean are then reconstructed based on their weighted average calculated.

The sigma points and corresponding weights are defined as

$$\mathcal{X}_0 = \hat{\mathbf{x}} , \quad (2.24)$$

$$\mathcal{X}_i = \hat{\mathbf{x}} + \left(\sqrt{(L + \lambda) \mathbf{P}} \right)_i , i = 1, \dots, L , \quad (2.25)$$

$$\mathcal{X}_i = \hat{\mathbf{x}} - \left(\sqrt{(L + \lambda) \mathbf{P}} \right)_i , i = L + 1, \dots, 2L , \quad (2.26)$$

$$W_0^m = \lambda / (L + \lambda) , \quad (2.27)$$

$$W_i^m = 1 / (2(L + \lambda)) , i = 1, \dots, 2L , \quad (2.28)$$

$$W_0^c = \lambda / (L + \lambda) + (1 - \alpha^2 + \beta) \quad (2.29)$$

$$W_i^c = W_i^m , i = 1, \dots, 2L , \quad (2.30)$$

where W_i^m are the first-order weights and W_i^c are the second-order weights, while $\left(\sqrt{(L + \lambda) \mathbf{P}} \right)_i$ denotes the i th row of square matrix root. Here $\lambda = \alpha^2(L + \kappa) - L$, such that α and β describe the spread of the sigma points with respect to the covariance matrix of the estimate.

With these weighted sigma points the prediction equations may be stated as

$$\hat{\mathbf{x}}_{k|k-1} = \sum_{i=0}^{2L} W_i^m f(\mathcal{X}_i) , \quad (2.31)$$

$$\mathbf{P}_{k|k-1} = \sum_{j=0}^{2L} W_j^c (f(\mathcal{X}_j) - \hat{\mathbf{x}}_{k|k-1})(f(\mathcal{X}_j) - \hat{\mathbf{x}}_{k|k-1})^T + \mathbf{Q}_k , \quad (2.32)$$

while the update may be stated as

$$\hat{\mathbf{z}}_k = \sum_{i=0}^{2L} W_i^m h(\mathcal{X}_i) , \quad (2.33)$$

$$\mathbf{S}_k = \sum_{i=0}^{2L} W_i^c (h(\mathcal{X}_i) - \hat{\mathbf{z}}_k)(h(\mathcal{X}_i) - \hat{\mathbf{z}}_k)^T + \mathbf{R} , \quad (2.34)$$

$$\mathbf{C}_k = \sum_{i=0}^{2L} W_i^c (\mathcal{X}_i - \hat{\mathbf{x}}_{k|k-1})(h(\mathcal{X}_i) - \hat{\mathbf{z}}_k)^T , \quad (2.35)$$

$$\mathbf{K}_k = \mathbf{C}_k \mathbf{S}^{-1} , \quad (2.36)$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \hat{\mathbf{z}}_k) , \quad (2.37)$$

$$\hat{\mathbf{P}}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T . \quad (2.38)$$

2.4 Extended Target Tracking

In classic target tracking, a single scan is assumed to generate at most a single point measurement on a target per time step. In extended target tracking, a single target may give rise to multiple spatially distributed measurements per time step. The measurements provide information about the target's spatial occupancy and, therefore, its extent and shape. An overview of the topic was presented in the surveys [MCS⁺14, GBR17b].

In Extended Target Tracking (ETT) targets can generate multiple spatially distributed noise corrupted measurements per time step, a set of measurements \mathcal{Z}^k on time step k can be described as

$$\mathcal{Z}^k = \{\mathbf{z}_i^k\}_{i=1}^{N^k} , \quad (2.39)$$

where $N^k \geq 0$ is the number of measurements at time k , while \mathbf{z}_i^k is the i th measurement this time step.

The aim is to estimate the PDF $p(\mathbf{x}|z)$ over the true state \mathbf{x} of the target consisting of the kinematic parameters x_{kin} and the extent parameters x_{ext} over time such that

$$\mathbf{x} = [\mathbf{x}_{\text{kin}}^T, \mathbf{x}_{\text{ext}}^T]^T. \quad (2.40)$$

Here, it is useful to expand the HMM which we saw in Section 2.1. The Markov process can now be modelled with multiple steps as the i th measurement is assumed to be generated via a source \hat{z}_i on the target's extent, with

$$\hat{\mathcal{Z}}^k = \{\hat{z}_j^k\}_{j=1}^{M_k}, \quad (2.41)$$

the set of all sources at time step k with cardinality M_k . This measurement generation process is illustrated in Figure 2.2.

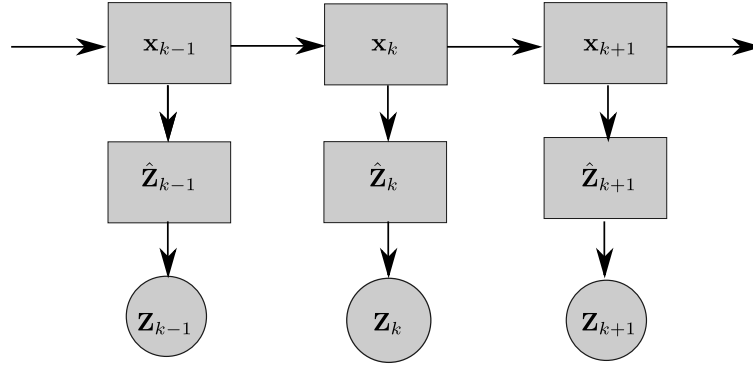


Figure 2.2: HMM for extended target tracking.

Three categories are often used for the description of the measurement sources. Measurements either stem from a set of discrete reflection points on the target, measurements can be emitted from the continuous contour of the target, or measurements can stem from the entire continuous volume of the target. The scenarios are illustrated in Figure 2.3. Each of these scenarios poses a different problem to be solved. We note that a model for tracking data in one of these scenarios needs to incorporate the respective assumption to provide an accurate estimate.

2.4.1 State Of The Art Extended Target Models

State of the art extended target models are typically divided into categories. One of the differentiation criteria for the target models is the degree of complexity. In [GBR17b] these complexity types are separated into three levels.

The simplest way to model an extended target is to not track the extent parameters and, hence, reduce the estimation to point targets that can generate multiple measurements per time step.

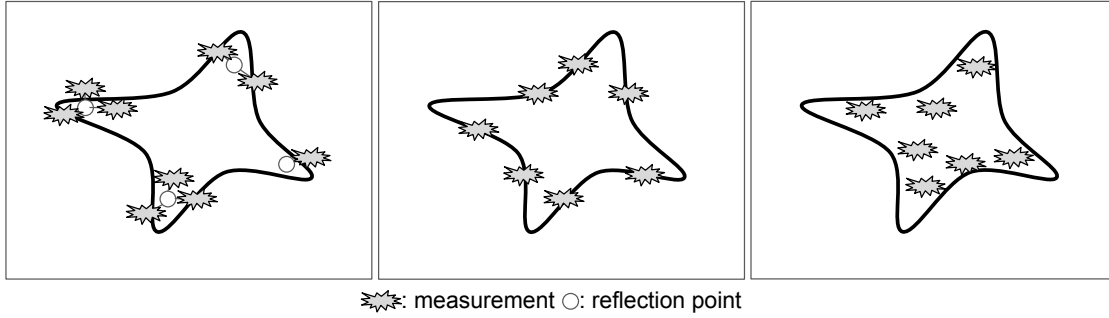


Figure 2.3: Measurement generation scenarios in extended target tracking. The figures from left to right show the scenarios with measurements originating from the contour, measurements originating from the volume and measurements from discrete sources.

Another group is tracking the extent with basic geometric representations. A typical example for this type is the representation as an ellipse, which is often represented by a density model based on a Gaussian distribution. These models typically use the spread of the measurements to estimate the extent of the target directly. One of the first developed and most commonly used elliptical approaches are the Random Matrix (RM) approach [Koc08] and its extension [FFK10] which track the target's extent by using the measurement spread of new measurements and treating the extent parameters independently from the target's kinematics. Other elliptical representations have been developed which focus on different aspects of the tracking problem. There are elliptical models which focus on the inclusion of a measurement rate for the target in the Gamma Gauss Inverse Wishart (GGIW) filter [GO12], models for the introduction of structured measurements on an elliptical target's boundary with the volcanormal density [BDD17] or the application of truncated Gaussian distributions [XWB⁺20b], and models with the focus on the representation of the extent via the semi axes length and orientation as separate state parameters in a joint state vector in the MEM-EKF and MEM-EKF* [YB19]. Other simple geometric representations are circles, line like models or rectangles [PMGA11, GLO11, BRAD16, BFH12, GS05].

Complex models can handle a large variety of shapes and measurement distributions and are not necessarily constrained to basic geometries. Complex models can be based on a set of ellipses joined by a shared kinematic state [GLO11]. A more general approach is the Random Hypersurface Model (RHM), first introduced for elliptical targets in [BH09] and extended to general star-convex targets in [BH14]. It is a general description of extended targets via a radial contour function $h(\mathbf{x}_{ext}, \alpha)$ with respect to a center point \mathbf{m} , where α is the angle from center to surface. This representation allows for modelling arbitrary star-convex shapes, based on the chosen contour function as illustrated in Figure 2.4. Different contour functions have been developed for the RHM. The original star-convex RHM [BH14] used Fourier coefficients to describe the contour, but other approaches have been deployed, with the most notable being the Gaussian-process model [WÖ15]. The radially defined RHM allows for inference based on measurements from contour and volume

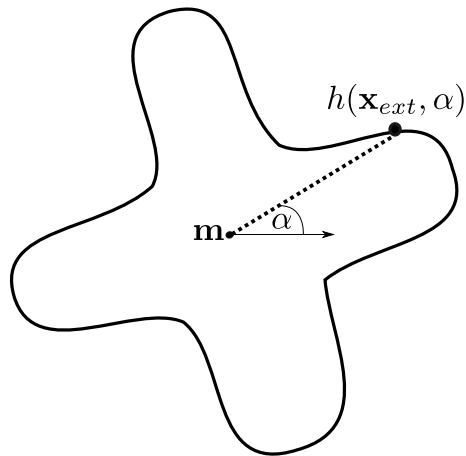


Figure 2.4: A typical radial contour function.

by using a scaling factor s on the contour function $h(\mathbf{x}_{ext}, \alpha)$ where measurement sources on a continuous contour can be modelled. The RHM represents measurement sources on the targets contour by using a scaling factor of 1. For measurements originating from the volume of the shape, it is assumed that the squared scaling factor is uniformly distributed between 0 and 1, as a property of the radial definition of the contour, can be deployed. This can be introduced to the target by approximating the scaling factor with a Gaussian distribution as described in [BH14]. Further non star-convex models are developed with level set RHMs in [ZFBH16], deformation functions [CLL17] on a set of control points as well as random polytopes in [Hei20], which is effectively using a set of joint half spaces to model the extended target.

2.4.2 Update Strategies For Extended Targets

One of the important steps in extended target tracking is the update of the state parameters with a set of new measurements. This requires to define a relation between state, measurement source and measurement.

The description of the target's volume and therefore the space of sources may be provided by a probability density with prime examples like the RM approaches [Koc08, FFK10] in which the extent will be updated with the spread matrices of all measurements on a time step, effectively using the statistical distribution of measurements to update the extent parameters.

For continuous contour-based descriptions of the target's extent, a continuous space of sources is available. For these, an association to a single source, which is optimal in a determined measure for example the Euclidean distance, is typically performed. A prime example where this is the case is the radially defined RHMs [BH11, WÖ15]. The chosen measure is the angular distance from mean to measurement and mean to contour. The radial contour function has a unique solution at

the angle, defined by the line from mean to measurement with distance 0 for every measurement's position except the mean itself.

Other models assume fixed point sources with respect to sensor properties, for which an association problem is established that has to be solved. An example of this is [GRMS14] in which multiple sources on a contour are determined based on sensor cells.

One specific example of resolving these problems is the PMHT filter for extended target tracking. The approach was used with discrete grid-based measurements and as the Histogram-PMHT in [WD14], with RHM discretised contour functions in [5,6] and [TLTK19], and with fixed, hand-crafted measurement sources on a rigid target in [BWS⁺17]. We will elaborate on the PMHT for extended target tracking in the following section as it will be used in the LIDAR and RADAR target model contributions.

2.4.3 PMHT Multi Source Extended Target Update

We will refer to the use of the single extended target version of the PMHT from our prior works [5], [6]. The setup of measurement generation considered with the extended target PMHT may be described with the following scenario. For a batch of K time steps, a time-varying state represented as a set of state realisations \mathcal{X} emits a set of measurements \mathcal{Z} from a set of state-dependent sources, where the set of emitting sources is described via the set $\hat{\mathcal{Z}}$.

The quantities we are interested in, with respect to the batch nature of the problem, may be described as

$$\begin{aligned}\mathcal{X} &= \{\mathbf{x}_k\}_{k=0}^K, \\ \mathcal{Z} &= \{\mathcal{Z}_k\}_{k=0}^K, \\ \hat{\mathcal{Z}} &= \{\hat{\mathcal{Z}}_k\}_{k=0}^K,\end{aligned}$$

in which the sets of measurements and sources on a single time step k are

$$\begin{aligned}\mathcal{Z}_k &= \{\mathbf{z}_i^k\}_{i=1}^{N_k}, \\ \hat{\mathcal{Z}}_k &= \{\hat{\mathbf{z}}_j^k\}_{j=1}^{M_k}.\end{aligned}$$

Here, N_k is the number of measurements and M_k is the number of measurement sources on time step k , such that \mathbf{z}_i^k describes the i th measurement on time step k , while $\hat{\mathbf{z}}_j^k$ is the j th potential measurement source on time step k . Each measurement source is assumed to be state-dependent via a potentially nonlinear function $\hat{\mathbf{z}}_j^k = h(\mathbf{x}_k, j)$ such that the measurement equation for a single measurement may be described as

$$\mathbf{z}_i^k = h(\mathbf{x}_k, q_i) + \mathbf{w}_{q_i}^k. \quad (2.42)$$

Here, $\mathbf{w}_{q_i}^k$ is the measurement noise with which the i 'th measurement is generated on time step k , where the noise is assumed to be zero-mean Gaussian white noise such that $\mathbf{w}_{q_i}^k \sim \mathcal{N}(0, R_{q_i}^k)$ with covariance matrix $R_{q_i}^k$. We introduce the index q_i , which denotes the real source $\hat{\mathbf{z}}_{q_i}^k$, responsible for generating the measurement \mathbf{z}_i^k . We note, that q_i^k is inherently unobservable, thus, we must assume all possible associations to measurement with $q_i^k \in [1, \dots, M_k]$. Here we will assume that each source can generate multiple measurements and denote the set of all possible assignments as

$$\mathcal{A} = \{q_i^k\} . \quad (2.43)$$

We further denote the probability of a source j generating a measurement as

$$P(q_i^k = j) = \pi_j , \quad (2.44)$$

such that the sources are treated as independent random variables with respect to the measurement generation.

The stated random variables may be described in the sense of a hidden Markov model. The *unknown data* to be estimated is our set of states \mathcal{X} , the *observed data* is our set of measurements over all time steps \mathcal{Z} , which express the dependent sources $\hat{\mathcal{Z}}$ over all time steps, while the *unobserved data* is the set of unobservable parameters \mathcal{A}_k which denote the associations to a source. We will here assume that the set of sources $\hat{\mathcal{Z}}$ is fully described by the relation to the set of states and is not treated as a random variable.

The PMHT uses the Expectation Maximization (EM) [NKM12] formulation for the calculation of a MAP estimate of the state parameters

$$\mathcal{X}_{MAP} = \arg \max_{\mathcal{X}} p(\mathcal{X} | \mathcal{Z}) , \quad (2.45)$$

which via marginalization over hidden parameters and Bayes rule, be expressed as

$$\mathcal{X}_{MAP} = \arg \max_{\mathcal{X}} \int_{\mathcal{A}} p(\mathcal{X}, \mathcal{A} | \mathcal{Z}) = \arg \max_{\mathcal{X}} \int_{\mathcal{A}} p(\mathcal{Z} | \mathcal{X}, \mathcal{A}) . \quad (2.46)$$

The EM algorithm is an alternating optimization algorithm based on an initial guess. The two steps are the Expectation and Maximization step.

- The Expectation step as the name suggests calculates the expectation of the log likelihood function $Q(\mathcal{X}^{l+1}; \mathcal{X}^l)$ based on the current estimate \mathcal{X}^l referred to as the Q -function based on an initial estimate.
- The Maximization step then seeks to find the parameters that maximize the Q -function.

This procedure is repeated until the algorithm converges or is stopped after a fix set of iterations. The important part is the statement of the Q -function and the provision of a suitable maximisation

strategy. Based on this formulation the EM's optimization function may be stated as

$$Q(\mathcal{X}^{l+1}; \mathcal{X}^l) = \sum_{\mathcal{A}} \log \left(p(\mathcal{X}^{l+1}, \mathcal{A} | \mathcal{Z}) \right) p(\mathcal{A} | \mathcal{X}^l, \mathcal{Z}) , \quad (2.47)$$

where l describes the EM iteration. The elements in \mathcal{X}^l will be denoted as \mathbf{x}_k^l . Here, the posterior probability may be expressed using Bayes rule

$$p(\mathcal{A} | \mathcal{X}, \mathcal{Z}) = \frac{p(\mathcal{Z}, \mathcal{X}, \mathcal{A})}{p(\mathcal{Z}, \mathcal{X})} , \quad (2.48)$$

in which the joint probabilities can be stated with the following proportionalities

$$\begin{aligned} p(\mathcal{Z}, \mathcal{X}, \mathcal{A}) &\propto p(\mathbf{x}_0) \prod_{k=1}^K p(\mathbf{x}_k | p(\mathbf{x}_{k-1})) \\ &\times \prod_{k=1}^K \prod_{i=1}^{N_k} \pi_{q_i} \mathcal{N}(\mathbf{z}_i^k; h(\mathbf{x}_k, q_i), R_{q_i}^k) , \end{aligned} \quad (2.49)$$

$$\begin{aligned} p(\mathcal{Z}, \mathcal{X}) &\propto p(\mathbf{x}_0) \prod_{k=1}^K p(\mathbf{x}_k | p(\mathbf{x}_{k-1})) \\ &\times \prod_{k=1}^K \prod_{i=1}^{N_k} \left(\sum_{j=1}^{M_k} \pi_j \mathcal{N}(\mathbf{z}_i^k; h(\mathbf{x}_k, j), R_{q_i}^k) \right) . \end{aligned} \quad (2.50)$$

By inserting the Equations (2.49) and (2.50) into Equation (2.48) we get the conditional

$$p(\mathcal{A} | \mathcal{Z}, \mathcal{X}) = \prod_{k=1}^K \prod_{i=1}^{N_k} \omega_{i, q_i}^k , \quad (2.51)$$

with

$$\omega_{i, j}^k = \frac{\pi_j \mathcal{N}(\mathbf{z}_i^k; h(\mathbf{x}_k, j), R_j^k)}{\sum_{t=1}^{M_k} \pi_t \mathcal{N}(\mathbf{z}_i^k; h(\mathbf{x}_k, t), R_t^k)} . \quad (2.52)$$

Each $\omega_{i, j}^k$ here has the interpretation of the posterior probability that the i th measurement was generated by the j th source. Based on this reformulation, we are able to state the Q -function of the

PMHT as

$$\begin{aligned}
Q(\mathcal{X}^{l+1}; \mathcal{X}^l) &= \sum_{\mathcal{A}} \log(p(\mathcal{X}^{l+1}, \mathcal{A}|Z)) \prod_{k=1}^K \prod_{i=1}^{N_k} w_{i,q_i}^n \\
&= \log \left(p(\mathbf{x}_0^{l+1}) \prod_{k=1}^K p(\mathbf{x}_k^{l+1} | p(\mathbf{x}_{k-1})^{l+1}) \right) \\
&\quad + \sum_{\mathcal{A}} \sum_{k=1}^K \sum_{j=1}^{M_k} \log(\pi_{q_i} \mathcal{N}(\mathbf{z}_i; h(\mathbf{x}_k, q_i), R_{q_i}^k) \omega_{i,j}^k) .
\end{aligned} \tag{2.53}$$

This function includes all possible associations and reacquires the complete calculation of every single one of them, which is not computationally feasible. However, following the same derivations as in the classic PMHT [SL95], a new optimization function \tilde{Q} can be stated, which is equivalent to Q in its optimization, with

$$\begin{aligned}
\tilde{Q}(\mathcal{X}^{l+1}; \mathcal{X}^l) &= \log(p(\mathcal{X}^{l+1}, C|Z)) \\
&\quad - \frac{1}{2} \sum_{j=1}^{M_k} (\tilde{z}_j - h(\mathbf{x}_k, j))^T \tilde{R}_j^k (\tilde{z}_j - h(\mathbf{x}_k, j)) ,
\end{aligned} \tag{2.54}$$

where the pseudo measurements and covariance matrices are used with

$$\tilde{z}_j = \frac{\sum_{i=1}^{|Z|} \omega_{i,j}^k \tilde{z}_i^t}{\sum_{i=1}^{|Z|} \omega_{i,j}^k}, j \in [1, \dots, M_k] , \tag{2.55}$$

$$\tilde{R}_j^k = \frac{R_j^k}{\sum_{i=1}^{|Z|} \omega_{i,j}^k}, j \in [1, \dots, M_k] . \tag{2.56}$$

This calculation provides us with an alternative Q function in the Expectation step of the EM algorithm with only a single pseudo measurement and pseudo covariance per measurement source, leaving the Maximization step to be performed. The optimisation is relatively easy, as the given function \tilde{Q} states a problem that may be optimised using respective Kalman smoothers [WRS02], which allows for the calculation of

$$\arg \max_{\mathcal{X}^{l+1}} \tilde{Q}(\mathcal{X}^{l+1}; \mathcal{X}^l) . \tag{2.57}$$

2.5 Dynamic Models

To predict an estimate between measurement updates, a process model needs to be stated. This process model has to be suitable to describe the target's motion properties. A wide range of motion models with different targets has been developed [LJ05].

A free mass model often describes the motion of pedestrians, and the motion of bicycles is often modelled using a nearly constant velocity model. For this thesis, however, only one motion model will be used to represent the motion of cars, the coordinated turn model with a polar velocity [RHG14]. The kinematic parameters of the coordinated turn can be described as

$$\mathbf{x}_{kin} = [x, y, \phi, v, \omega]^T, \quad (2.58)$$

with the position $[x, y]$ the yaw ϕ , the polar velocity v along the target's yaw and the yaw rate ω . It describes the transition from time $k - 1$ to k under the assumption that the torque and acceleration can be modeled as zero-mean Gaussian white noise. The transition function in this model is

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + g(\mathbf{x}_{k-1})v_k, \quad (2.59)$$

where $v_k \sim \mathcal{N}(0, \mathbf{Q})$, in which

$$\mathbf{Q} = \begin{pmatrix} \sigma_{\dot{v}} & 0 \\ 0 & \sigma_{\dot{\omega}} \end{pmatrix} \quad (2.60)$$

is the covariance matrix modelling the assumption that the derivative of velocity and yaw rate is modelled as independent zero-mean white noise. These are mapped to the state dimension based on this relation with function

$$g(\mathbf{x}_{k-1}) = \begin{pmatrix} 0.5 \cos(\phi)t^2 & 0 \\ 0.5 \sin(\phi)t^2 & 0 \\ t & 0 \\ 0 & t \\ 0 & 0.5t^2 \end{pmatrix}. \quad (2.61)$$

The state transition function in this model is stated as

$$f(x) = \begin{bmatrix} x + \frac{2v}{\omega} \sin(\frac{\omega t}{2}) \cos(\phi + \frac{\omega t}{2}) \\ y + \frac{2v}{\omega} \sin(\frac{\omega t}{2}) \sin(\phi + \frac{\omega t}{2}) \\ \phi + \omega t \\ v \\ \omega \end{bmatrix}. \quad (2.62)$$

2.6 Automotive Point Cloud Sensors

The sensors we are interested in are point cloud generating sensors. They provide spatially distributed point measurements of varying quality and quantities, based on their working principle and mounting position. For this work, we will only categorise the sensors based on their type and

work with the data on point cloud level and not on the low-level measurement principles as we focus on the data fusion aspect.

Two major types of LIDAR sensors have been developed for automotive purposes, scanning LIDARs with rotating mirrors and flash LIDARs which requires no moving parts [War19]. As of now LIDAR sensors deployed in public data sets are mostly scanning LIDARs. These provide 3d point measurements from the environment. Their major differentiation comes from the number of layers, with 4, 16, 32 or 64 layers and in special cases a variation of the mounting position.

The deployed automotive RADARs are Frequency Modulated Continuous Wave (FMCW) systems [HTS⁺12]. The RADAR sensors used in public data sets are long-range RADAR sensors with a 76 - 81 GHz band which can provide multiple measurements per target. These radars are often planar sensors in contrast to the commonly used 3d LIDAR sensor systems.

The measurement patterns on the targets of interest need to be understood and accounted for in the extended target model to allow for a correct estimation and inference strategy. This includes the degree of freedom in the shape representation that needs to be modelled to accurately represent the target, the distribution of the measurements' origins on the target and sensor specific quantities such as the noise which is present in the measurement generation.

2.7 Automotive Point Cloud Data Sets

| data set | LIDAR | RADAR | Camera | Annotation boxes |
|--|-------|-------|--------|------------------|
| Kitti [GLSU13] | ✓ | × | ✓ | ✓ |
| Oxford RobotCar [MPLN17] [BGM ⁺ 20] | ✓ | ✓ | ✓ | × |
| nuScenes [CBL ⁺ 20] | ✓ | ✓ | ✓ | ✓ |
| Lyft Level 5 [KUH ⁺ 19] | ✓ | × | ✓ | × |
| Argoverse [CLS ⁺ 19] | ✓ | × | ✓ | ✓ |
| Woodscape [YHH ⁺ 19] | ✓ | × | ✓ | ✓ |
| H3D [PMGC19] | ✓ | × | ✓ | ✓ |
| Astyx [MK19] | ✓ | ✓ | ✓ | ✓ |
| A2D2 [GKM ⁺ 20] | ✓ | × | ✓ | ✓ |
| Radiate [SDPM ⁺ 21] | ✓ | × | ✓ | ✓ |
| A*3D [PSP ⁺ 20] | ✓ | × | ✓ | ✓ |
| Toronto-3D [TQM ⁺ 20] | ✓ | × | ✓ | ✓ |
| Waymo [SKD ⁺ 20] | ✓ | × | ✓ | ✓ |
| PixSet [DMT ⁺ 21] | ✓ | ✓ | ✓ | ✓ |

Table 2.1: Available automotive data sets with attribution of available sensor types and the availability of annotation bounding boxes.

In recent years a list of data sets with different point cloud sensors and sensor setups has been released. These data sets are generated with different goals in mind.

There are general data sets which provide 3d point cloud objects, e.g. the Shapenet data set [CFG⁺15]. These data sets provide data of whole scenes or objects. They focus on a full representation of a large set of objects as 3d point clouds.

We, however, are particularly looking for automotive data sets that feature point cloud sensors and annotations of other traffic participants. For these a multitude of data sets has been provided over the past decade shown in the following listing, in which we included only data sets that feature at least one point cloud sensor.

- The Kitti data set [GLSU13] was released as an early contender featuring a car equipped with a 64 layer LIDAR sensor on the roof and multiple front facing cameras, with bounding box annotations for different traffic participants and with recent extensions of the semantic labels in Semantic Kitti [BGM⁺19].
- The Oxford RobotCar data set [MPLN17] uses two 2d LIDAR sensors as well as a 4 layer 3d LIDAR sensor on the roof and 6 cameras and its RADAR extension [BGM⁺20] features a high resolution 3d, 76 to 77 GHz RADAR.
- The nuScenes data set [CBL⁺20] featuring 32 layer LIDAR on the roof, a set of 5 77 GHz RADAR sensors and a set of cameras with recent updates including point wise semantic information and an environmental map. It further provides annotation bounding boxes with a frequency of 5 Hz.
- The Lyft Level 5 data set [KUH⁺19] contains three LIDAR sensors in different mounting positions, one 64 layer LIDAR on the roof and two 40 layer LIDAR sensors in the front bumper. This data set is annotated as well and provides bounding boxes with a frequency of 5 Hz.
- The Argoverse data set [CLS⁺19] features 7 cameras and data from two 32 layer LIDAR sensors combined with information from an environmental map. It further features bounding box annotations for other traffic participants.
- The WoodScape data set [YHH⁺19] provides data from a 64 layer LIDAR sensor as well as from a set of 4 fisheye cameras. It further adds annotation bounding boxes.
- The H3D data set [PMGC19] which features 3 cameras and a 64 layer LIDAR on the roof has a focus on crowded scenarios and provides further 3d annotation bounding boxes.
- The A2D2 data set [GKM⁺20] features a set of 6 cameras and 5 16 layer LIDAR sensors on the vessel's roof with the focus on semantic information in the camera data and a mapping on the respective LIDAR information. No bounding box annotations are available.
- Radiate [SDPM⁺21] is a data set that uses a high resolution 79 GHz RADAR, a 32 layer LIDAR sensor and a stereo camera, with focus on different weather conditions. It also provides 3d bounding boxes.

- The A*3D data set [PSP⁺20] features a spinning 64 layer LIDAR and two front facing cameras. It has one of the highest annotation frequencies with 0.2 Hz.
- The Astyx data set [MK19] provides data from a 16 layer LIDAR sensor and RADAR data from their own sensors. In addition to camera data, annotations are provided. We note that only a small set of annotated frames is available yet.
- The Toronto-3D data set [TQM⁺20] also provides data from a 32 layer LIDAR sensor and 5 panoramic cameras, with the aim of semantic segmentation. As the semantic segmentation is the focus no annotation bounding boxes are provided.
- The Waymo open data set is one of the largest available data sets. It features a set of 5 LIDARs in different mounting positions and five cameras [SKD⁺20]. The data set is notable as it uses separate annotations on LIDAR and camera instead of joint annotations.
- The PixSet data set [DMT⁺21] is one of the newest data sets available with a 64 layer LIDAR and a 76 – 81 GHz FMCW RADAR as well as 3 cameras. The LIDAR in this data set is a flash LIDAR which stands in contrast to all aforementioned data sets which use LIDAR sensors with rotating mirrors. It further features annotation bounding boxes.

An overview of automotive data sets with point cloud sensors is listed in Table 2.1. It shows the availability of sensor types and annotations in the corresponding data set to provide a short summary of the aforementioned data sets.

2.8 Metric for Extended Target Trackers Based On Bounding Box Annotations

One of the most common annotations used for LIDAR and RADAR data are bounding box annotations. These are typically generated based on camera and LIDAR data and require a lot of manual effort from a team of annotators. While the bounding box itself is no perfect estimate of the target's shape it is the most deployed approach in the available data set.

To compare an estimate of an extended target to this bounding box different metrics can be used, which provide different advantages and drawbacks. Even though the work is mostly considering elliptical targets, a good overview over typical metrics is provided in [YBG16].

A simple and direct comparison of the state parameters is often performed using the Mean Squared Error (MSE). This approach assumes, however, that the state can be represented in the same manner as the ground truth bounding box's parameters. A center point can typically be defined for every extent model, it is, however, no suitable metric for extended target models that exceed basic geometries and especially for models with incompatible shape parameters.

From computer vision the Intersection over Union (IoU) is often deployed to determine the similarity of rectangular estimates. If no overlap exists between an estimate with the ground truth this score will always be 0. Even if one shape has a perfect estimate of the target's shape and another is completely different, there will be no difference in the score itself. We further note that calculating this score for non polygon shapes can be a complex task.

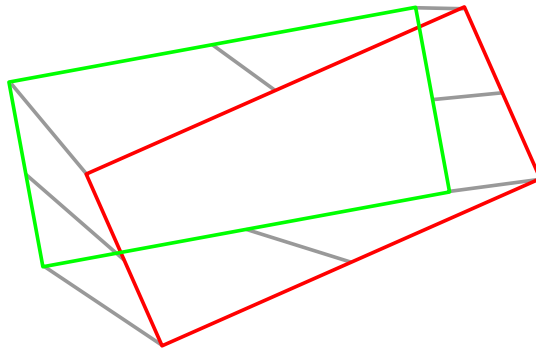


Figure 2.5: Illustration of the Gaussian Wasserstein metric for rectangular extended targets. For two different shapes the best association is provided.

A rather recent work for extended target metrics for elliptical targets was proposed in [YBG16]. In this work the Gaussian Wasserstein [GS84] distance and more specifically a discretised version based on the Optimal Subpattern Assignment (OSPA) [SVV08] distance was proposed to evaluate the similarity of elliptical extended targets. For two sets of discretised points $\{p_{\mathbf{x}}^1, \dots, p_{\mathbf{x}}^n\}$ and $\{p_{\hat{\mathbf{x}}}^1, \dots, p_{\hat{\mathbf{x}}}^n\}$ on the surface, defined by state \mathbf{x} and the estimate $\hat{\mathbf{x}}$, the OSPA distance can be expressed as

$$d_{\text{OSPA}} = \min_{\pi \in \Pi} \left(\frac{1}{n} \sum_{i=1}^n \|p_{\mathbf{x}}^i - p_{\hat{\mathbf{x}}}^{\pi(i)}\|^p \right)^{\frac{1}{p}}. \quad (2.63)$$

This definition allows to be used on other shapes as well. While the paper indicates that the use of the 4 boundary points on the ellipses main axis can be used it also proposes to use the metric for other shapes such as rectangles.

For this work we resort to using the Wasserstein distance for large scale evaluations. The discretised points chosen for it are the corners of the rectangular bounding box as well as the points on the middle of each side which is exemplarily illustrated in Figure 2.5. We will further use the $p = 1$ order Wasserstein distance as it is commonly used in the application of the OSPA metric in multi-target-tracking [HVV15].

Point Cloud Sensor Data Analysis

Contents

| | |
|---|----|
| 3.1 Problem Setting | 25 |
| 3.2 State Of The Art Analysis Methods | 26 |
| 3.3 Data Preparation And Analysis Strategy | 27 |
| 3.4 Representative Data Sets | 29 |
| 3.5 Point Cloud Analysis | 31 |
| 3.6 Discussion Of Target Types And Model Requirements | 38 |

Modelling extended targets typically depends on the sensor and the target type to be represented. Here, we are interested in the distribution and patterns that automotive cloud sensors generate on a specific target type and desirable features an extended target model has to fulfill to allow for accurate inference strategies. We will first state the problem in Section 3.1. Next, an overview on state of the art strategies for the analysis of measurement distributions for automotive point cloud sensors is provided in Section 3.2. The analysis strategy we use in this chapter will then be presented in Section 3.3. We will use representative data sets namely nuScenes [CBL⁺20] and Lyft level 5 [KUH⁺19]. This choice and a more detailed overview of the data sets are discussed in Section 3.4. The analysis executed on the chosen data sets is provided in Section 3.5. It is performed with respect to typical target types of interest, which will be bicycles, pedestrians and cars. Based on the analysis, we will state the requirements of a suitable extent model for the respective target type in Section 3.6 and compare it to a state of the art radially defined extended target model.

3.1 Problem Setting

Recursive estimation typically tries to incorporate as much prior information into the target models as possible to improve the estimation. This includes information about the measurement generation

function as well as the process models. Examples from classical point target tracking are the incorporation of sensor and target-dependent noise terms as well as physically motivated motion models. In extended target tracking, more complex assumptions about the target's measurement generation process can be made, as the spatial distribution of measurements on the target needs to be accounted for. This includes the distribution of measurements on the target, which is an additional factor that needs to be incorporated and can vary with respect to the target type and sensor used, resulting in different signatures and patterns. It is important to note that the measurement distribution in this work will be considered in a bird-eye projection of the measurements as it is commonly considered in extended target tracking.

Modelling extended targets and incorporating the distribution of measurements into their extent becomes a more conquerable task with the availability of diverse automotive perception data sets. Therefore, a large-scale analysis of the measurement distribution concerning the most prominent sensor setups needs to be investigated to correctly incorporate it into the estimation procedure. This analysis should provide information about the general requirements for specific target types in the perception of different point cloud sensors and configurations. It includes the general size that needs to be covered, the expected number and the distribution of measurements on the target, and a determination of the measurement generation scenario for extended target tracking. Analysing the distribution of the measurements for targets with the same type but different sizes can be challenging as a shared space for a quantitative analysis is required.

Therefore, the problems we are facing in the analysis of the measurement of a specific target type are the accumulation and extraction of point clouds belonging to the specified target type, the determination of a space in which this data can be analysed over all instances of the same type, and the establishment of the measurement scenario in extended target tracking. We will orient ourselves on state of the art analysis strategies to tackle these problems and aim for the statement of requirements an extended target model needs to incorporate to suit the data of a specific target type well.

3.2 State Of The Art Analysis Methods

In [BML⁺17], the data points from a 77 GHz RADAR sensor reflected on a single standing target Sport Utility Vehicle (SUV) have been accumulated in the SUVs ego coordinate system. This allowed the authors to analyse the distribution of measurements on the car to find characteristics on it with respect to reflection centers, penetration depth and relative distribution concerning the angle under which the car was seen. The work emphasised that scattering centers, especially on the car's wheel housings, are prominent features that need to be accounted for and that RADAR measurements are not simply generated on the surface of the target, as the RADAR is able to penetrate the materials of the car. Similarly, the works of [SD18] accumulated short range RADAR measurements in the targets coordinate system but accounts for the targets potentially varying size

by scaling all measurements into a unit coordinate system, with the targets rotational center as the origin, providing information about the cars shape independent from its extent. This work focused on density representations of the data and also found the scatter source behaviour especially on the wheelhouses, similar to the results from [BML⁺17]. This analysis system was further used in [HDKD18] for a set of RADAR sensors as well as a 4 layer LIDAR sensor, providing further statistical measurement distributions for a single car, a set of bikes, and a set of pedestrians in scaled coordinates. In contrast to the aforementioned works [BML⁺17, SD18], [HDKD18] did not focus on the aspect angles and the conditionals but used a single marginal over all measurements instead. It is of further note that the marginal of the LIDAR data was split into the 4 layers of the sensor to provide an insight into the specific behaviour available in this datatype. These strategies all work with accurately annotated data which makes it possible to provide all measurements for the same target by using the annotation bounding box on the scan to extract measurements belonging to the target. In addition to these works, we note the work of [MWW⁺20] in which the data from the nuScenes data set is accumulated and post-processed locally to allow for the generation of synthetic measurements from the accumulated data for single target instances.

3.3 Data Preparation And Analysis Strategy

Our aim is the analysis of the point cloud data for a set of target types, namely pedestrians, bicycles, and cars. Based on the state of the art strategies shown in Section 3.2, we will state our analysis with respect to cuboidal bounding boxes. Given annotated data sets, we will use all annotations belonging to a specific type to provide information about its model requirements.

First, we aim to calculate some basic statistics for cuboidal bounding box annotations to the general dimension of each target type. This provides a general statistic of the expected extent in length, width and height and, therefore, the scaling behaviour the model requires. We, however, note that the cuboidal bounding box does not provide the actual structure, distribution or expected number of detections posed by the measurements and instead serves as an approximation of the contour of the target. A new coordinate system is required to represent the distribution of measurements over all targets of the same type. For this purpose we use the coordinate system proposed in the work [SD18] by Scheel et. al. in which all measurements are accumulated in the coordinate system of the target's bounding boxes and are scaled with respect to their annotation's length and width centered around the car's rear axis, providing all measurements in a joint coordinate system. In contrast to this approach, we use a bounding box centric coordinate system that removes the direct correlation to the car's rotation center but provides all measurements relative to the car's center. This step is necessary as we want to work with a large number of different targets, for which no such common center exists.

Thus, given a bounding box with translation $\mathbf{m}_{bb} = [x, y, z]^T$, the 3d orientation with roll, pitch and yaw $\phi = [\alpha, \beta, \gamma]$ and size $\mathbf{S}_{bb} = \text{diag}(l, w, h)$, where the sizes are the targets half length l ,

half width w and half height h , we can transform a single frame into the vehicle coordinates. The relation from the global frame into the coordinates of an annotation can be described as

$$\mathbf{z}_g = \mathbf{R}_{\phi_{bb}}(\mathbf{S}_{bb}\mathbf{z}_l) + \mathbf{m}_{bb} \quad (3.1)$$

where \mathbf{z}_g is the measurement in global coordinates, \mathbf{z}_l describes the measurement in the annotation coordinates while $\mathbf{R}_{\phi_{bb}}$ is the rotation matrix. The inverse transformation, therefore, provides the function from local to global coordinates. This transformation is exemplarily shown for a single frame in Figure 3.1 which provides this transformation from a bird-eye perspective.

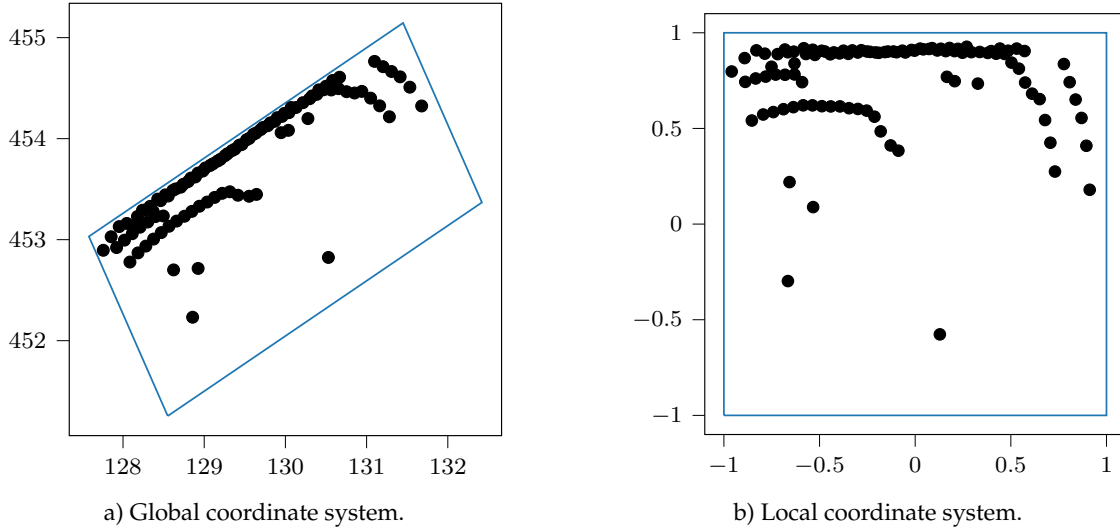


Figure 3.1: Example of the coordinate transformation on a single frame of the nuScenes data set.

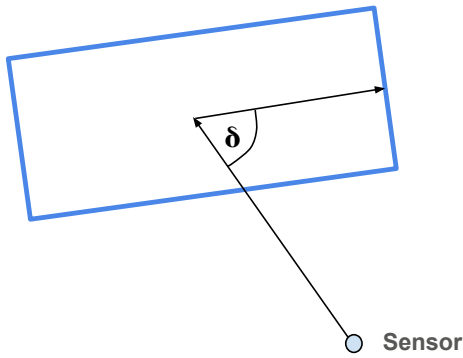
In principle, the complete data from the scene can be transformed into this coordinate system. We, however, are only looking for data belonging to the target. We enforce this by using a threshold on all target dimensions in scaled coordinates filtering measurements that are in the range $[-1.2, 1.2]$ in all three dimensions. We note that this may include ground detections and further set a lower threshold in height to effectively remove these detections based on their distance to the annotation bounding box's ground plane. The third dimension is dropped for detections from planar sensors but works analogously in the data extraction and requires no ground filtering otherwise. Based on this data, we can provide the statistical distribution as histograms of measurements in scaled coordinates allowing us to provide an approximation of the properties mentioned above, such as structure and measurement distribution. We can further provide the number of extracted detections on the target concerning the distance to the sensor.

We note that this histogram will include all data points in the data independent of the position of the sensor. While it provides the general measurement pattern, it obfuscates effects like the potential self-occlusion with respect to the relative viewpoint from the sensor to the annotated

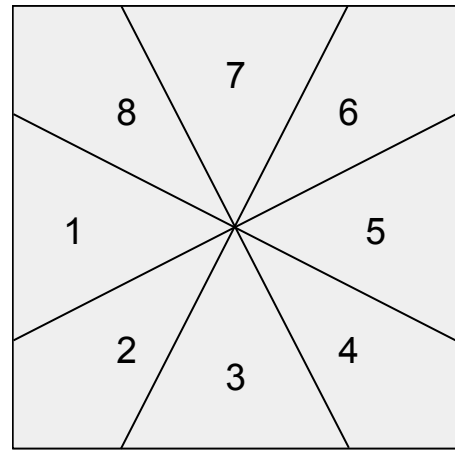
target, as it is visible in the single frame shown in Figure 3.1. To avoid this obfuscation, we will further create conditional distributions for the aspect angle δ , the angle between the bounding boxes orientation vector and the vector from the sensor to the annotations' center in the unit coordinate system, shown in Figure 3.2a. Here, we decided to specifically use eight conditionals over data that fall into a specific aspect angle interval, each spanning a range of 0.25π . The regions are centred around

$$[-\pi, -0.75\pi, -0.5\pi, -0.25\pi, 0, 0.25\pi, 0.5\pi, 0.75\pi] \quad (3.2)$$

as illustrated in Figure 3.2b.



a) Illustration of the aspect angle δ for a 2d bounding box.



b) Illustration of the angular ranges by the index of center angles indicated in Equation (3.2).

Figure 3.2: Illustrations of the aspect angle and the conditional ranges.

These statistics and accumulations then allow for an interpretation and description of requirements that an extended target model needs to provide for the combination of sensor setup and target type. Here we specifically focus on the current view on the target concerning self-occlusion and similar phenomenons.

3.4 Representative Data Sets

For this work, we will focus on two representative data sets from the list of available ones in Section 2.7: the nuScenes data set [CBL⁺20] and the Lyft level 5 [KUH⁺19] data set.

The nuScenes data set provides a total of 750 scenes, with each covering approximately 20 seconds of recordings. The sensors it features are

- one 32 layer LIDAR sensor on the roof of the car which captures data with a frequency of 30 Hz, and

- five 77 GHz FMCW RADAR sensors that provides planar detections with a 13 Hz capture frequency .

In addition to the sensor data, we are provided with target annotations on keyframes where a keyframe occurs with a frequency of 2 Hz. Each of these annotations belongs to a single time-varying target in the scene, which is referred to as an instance. These instances have class labels, e.g. car, pedestrian, or bicycle.

This data set was chosen as it features both LIDAR and RADAR sensors. It further provides the necessary cuboidal annotations that allow for interpolation of single instances to all sensor data frames in between. Of these sensor types, the availability of RADAR data is a rare feature over all automotive data sets, as most data sets focus on camera and lidar sensor systems. Only three other data sets featuring the sensor type are available. As of today, there are, to our knowledge, only the Oxford robotics data set [BGM⁺20], which provides no annotations, the Astyx data set [MK19], which provides only a minimal amount of annotations and the newly released [DMT⁺21], which was not considered, as the release was too recent.

The Lyft level 5 data set, on the other hand, provides 180 scenes of approximately 20 seconds of recordings. The sensors it features are

- one 64 layer LIDAR sensor on the roof of the car which captures data with a frequency of 10 Hz, and
- five 40 layer LIDAR sensors in front at the height of the front bumper with a frequency of 10 Hz.

This data set provides keyframe annotations and instances in the same fashion as the nuScenes data set. The keyframes occur with a frequency of 5 Hz and cover compatible classes to the nuScenes data set. We note, however, that only annotated frames are provided, in contrast to nuScenes which features all updates of every sensor, effectively reducing the available update frequency of the sensors from 10 Hz to 5 Hz.

The Lyft level 5 data set is desirable for its variety in the sensors mounting positions. Roof-mounted LIDAR sensors are commonly used in publicly available data sets. However, a setup with a set of bumper-mounted sensors represents another setting that may be used. It is desirable as classic automotive sensor setups like Ultrasonic already use it. Therefore, the data set allows for evaluating the influence the mounting position can have on the typical measurement distribution available.

We summarize the data set choice of Lyft level 5 and nuScenes with the following points:

- The data sets feature a large set of scenes and in turn a large set of frames to analyze.
- Both of these data sets provide instance-based cuboidal bounding box annotations for their respective sensors, which, as described in the state of the art in Section 3.2, is commonly used.

- Representative LIDAR and RADAR sensors are available.
- The influence of the mounting position on the LIDAR measurements on a target may be observed.

With this choice of data sets we can proceed to the point cloud data analysis.

3.5 Point Cloud Analysis

We will provide the statistics shown in Section 3.4 for pedestrians, bicycles and cars for the data of the LIDAR and RADAR sensors in the nuScenes [CBL⁺20] and Lyft level 5 [KUH⁺19] data set.

First, we will provide a statistical distribution over the width and length of each target type's general bounding boxes sizes, which will be aggregated jointly over both data sets in Section 3.5.1. We illustrate histograms for length and width and jointly over both dimensions to illustrate the ranges and correlations to indicate the model's scaling behaviour.

Next, an analysis based on the extracted data points for every sensor will be generated. The threshold for ground removal in LIDAR data in scaled coordinates, described in Section 3.3, was chosen as 0.8 for all models after visual inspection of the accumulated point clouds. The accumulated point cloud data will be shown as histograms with each sensor type's marginal and all eight conditionals. The histograms are spanned over the selected dimensions in scaled coordinates covering the extracted spaces. The resolution of the histograms was tuned by hand with the aim to capture visible features if possible. For LIDAR data, the histogram is created with a resolution of 500x500 bins, while the resolution for RADAR data was selected with 100x100 bins. The dimensions for the histogram will be length and width as it is available in all sensor types and spans the typical projection used in extended target tracking. For cars and bicycles in LIDAR data, an additional histogram view in logarithmic scale will be added, allowing to gain further insights into the measurement distribution. The logarithmic scale is chosen to reveal secondary measurement sources, as the Birdseye projection might obfuscate.

3.5.1 Bounding Box Size Distribution

We provide general statistics over the ranges covered on the bounding box ranges in Table 3.1. This table lists the number of instance bounding boxes used by the target type as well as the respective interval, mean and standard deviation of length and width. The histograms over length and width are shown in Figure 3.3 for pedestrians, in Figure 3.4 for bicycles and in Figure 3.5 for cars.

Pedestrians For pedestrians, 12223 instances are used. The dimensions for pedestrians cover lengths from 21 cm to 2.24 m with a 95-percentile of 1.04 m and widths from 22 cm to 1.97 m with a 95-percentile of 0.91 m. We note that the sizes in the extrema of the ranges are kids as the smaller end and pedestrian that carry or pull objects, as they are included in the annotation bounding box.

| Target Type | #Instances | Length (m) | Width (m) |
|-------------|------------|----------------------------------|---------------------------------|
| | | Range, mean, std , 95-percentile | Range, mean, std ,95-percentile |
| Pedestrian | 12223 | [0.21,2.24], 0.71, 0.19, 1.04 | [0.22,1.97], 0.66, 0.14, 0.91 |
| Bicycle | 1759 | [0.81,4.40], 1.77, 0.27, 2.08 | [0.21,1.77], 1.67, 0.20, 0.88 |
| Car | 21578 | [2.24,7.83], 4.65, 0.45, 5.48 | [0.92,3.48], 1.92, 0.17, 2.22 |

Table 3.1: General statistics of the annotation bounding boxes length and width.

Both dimensions follow an approximately Gaussian distribution and are close to each other in mean and variance as visible in the histograms over both dimensions and a difference of 5 cm in the mean and 5 cm in the standard deviation. A general correlation can be found in the histogram over both dimensions as the joint distributions orientation is in general about 0.25π , suggesting a direct relation of length and width. It is, however, noteworthy that it is slightly skewed towards longer than broader targets. This skew can be attributed to the motion of arms and legs, which provides a generally larger range in motion along with the orientation of the pedestrian.

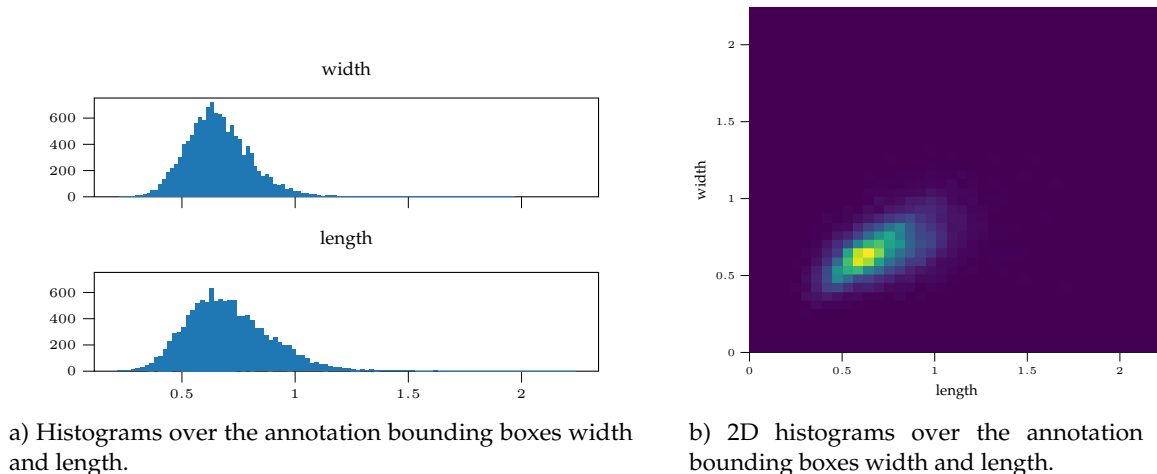


Figure 3.3: Histograms for pedestrian bounding box sizes.

Bicycles For bicycles, 1759 instances are available, which is the smallest number of instances for all analyzed target types. Therefore, we expect potential biases in the aspect angle regions and coverage of the targets. A significant difference in length and width is shown with the length ranging from 81 cm to 4.40 m with a 95 percentile of 2.08 m and the width ranging from 21cm to 1.77 m with a 95 percentile of 0.88 m. The significant outliers in this distribution are unexpected at first. However, it is explained with the annotation rules as a pedestrian walking and pushing his bicycle as well as additional object's on it are also included in the bounding box, leading to a larger range of widths and length. The distribution is not simply Gaussian but seems to consist of an overlap of different distributions emphasizing a larger variety in the annotated targets' sizes.

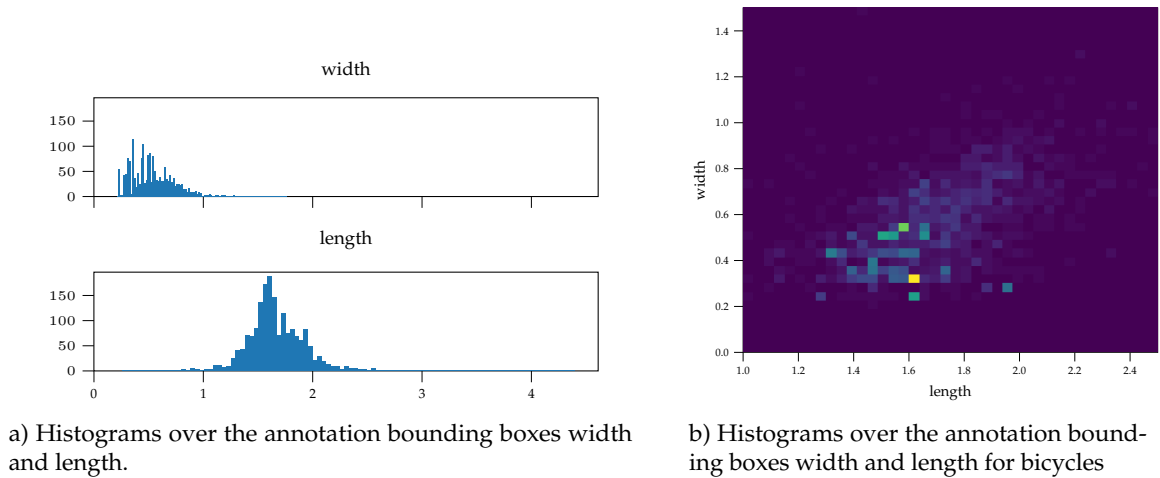


Figure 3.4: Histograms for bicycle bounding box sizes.

Cars The largest number of 21578 instances is available for cars. The space we cover in length and width is significantly different from each other, as the length ranges from 2.24 m to 7.83 m and provides a 95-percentile of 5.48 m. In comparison, the width ranges from 0.92 m to 3.48 m providing a 95-percentile of 2.22 m, with the upper limits being extreme outliers of atypical cars. For the relation of length and width, we note that the length can vary significantly more substantially than the width. This is expressed in the parameter wise and joint distribution where the orientation of the distribution is nearly axis-aligned, suggesting a lower correlation between the dimensions and, therefore, a larger independent range.

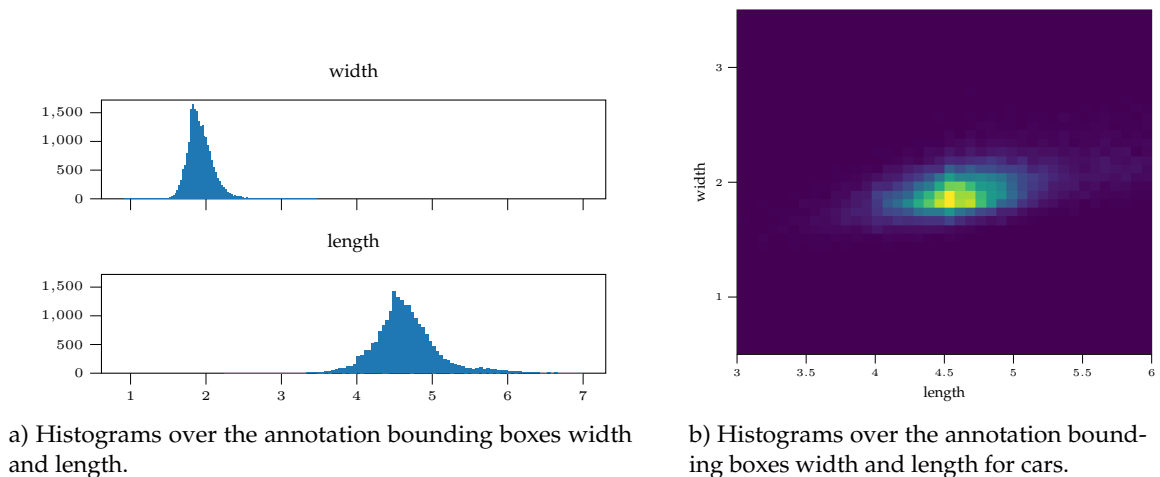


Figure 3.5: Histograms for car bounding box sizes.

3.5.2 Measurement Distribution

We provide the histograms in scaled coordinates for our selected data sets. For nuScenes in Figure 3.6 and for Lyft level 5 in Figure 3.7. For both data sets, a group of distributions for each combination of sensor and target type is generated. For every combination, nine histograms are provided. These are eight aspect angle conditionals ordered according to the illustration in Figure 3.2b and with the marginal in the center. The sensor types will specifically be the complete set of RADAR sensors and the single roof-mounted LIDAR sensor in the nuScenes data set, as well as the bumper- and roof-mounted LIDAR sensor from the Lyft level 5 data set. We further add the histograms in logarithmic scale over the cars and bicycles marginals, and exemplary conditionals in Figure 3.8 to illustrate the measurement distribution with respect to secondary structures as the dominating part of measurements can obfuscate secondary measurement sources in these targets. The number of extracted frames containing measurements provided per sensor and the target type are listed in Table 3.2, which is strongly dependent on the sensor system and the data set size.

| nuScenes | | | |
|---------------|------------|---------|-----------|
| | Pedestrian | Bicycle | Car |
| Roof LIDAR | 1.816.693 | 94.800 | 3.311.451 |
| 77 GHz RADARs | 153.715 | 18.180 | 1.755.623 |

| Lyft | | | |
|--------------|------------|---------|---------|
| | Pedestrian | Bicycle | Car |
| Front LIDARs | 5.895 | 3.482 | 93.914 |
| Roof LIDAR | 27.411 | 22.998 | 570.362 |

Table 3.2: Number of Frames per target and sensor type for both of the selected data sets.

We will go through the target types in the following, describe the distribution available per target type, and clarify sensor-specific similarities and details.

Pedestrians For pedestrians a general circular extent in scaled coordinates is available in the marginal over all sensor types shown in Figures 3.6a, 3.6b, 3.7a, 3.7b. Comparing the LIDAR sensors to each other reveals that the general measurement distribution over the conditionals is similar as well. Each conditional shows a bias towards the sensor as the pedestrian is self occluding the other side such that only half the pedestrian's body generates measurements. The major difference between the LIDAR sensors is the density of the measurements. The roof-mounted sensors with 32 and 64 layers provide a dense coverage of the target, in the bumper-mounted 40 layer sensor a less dense coverage of the shape is visible in the bumper-mounted. This can be attributed to the number of measurements accumulated which is smaller than that of the higher resolution sensors, as well as a stronger coverage of the legs which are more dynamic and provide a less rigid measurement source than the upper body. Comparing the RADAR data to the LIDAR data instead reveals major differences in the density. The general density in scaled coordinates covers more space and is less precise in covering the target's true contour. The most interesting

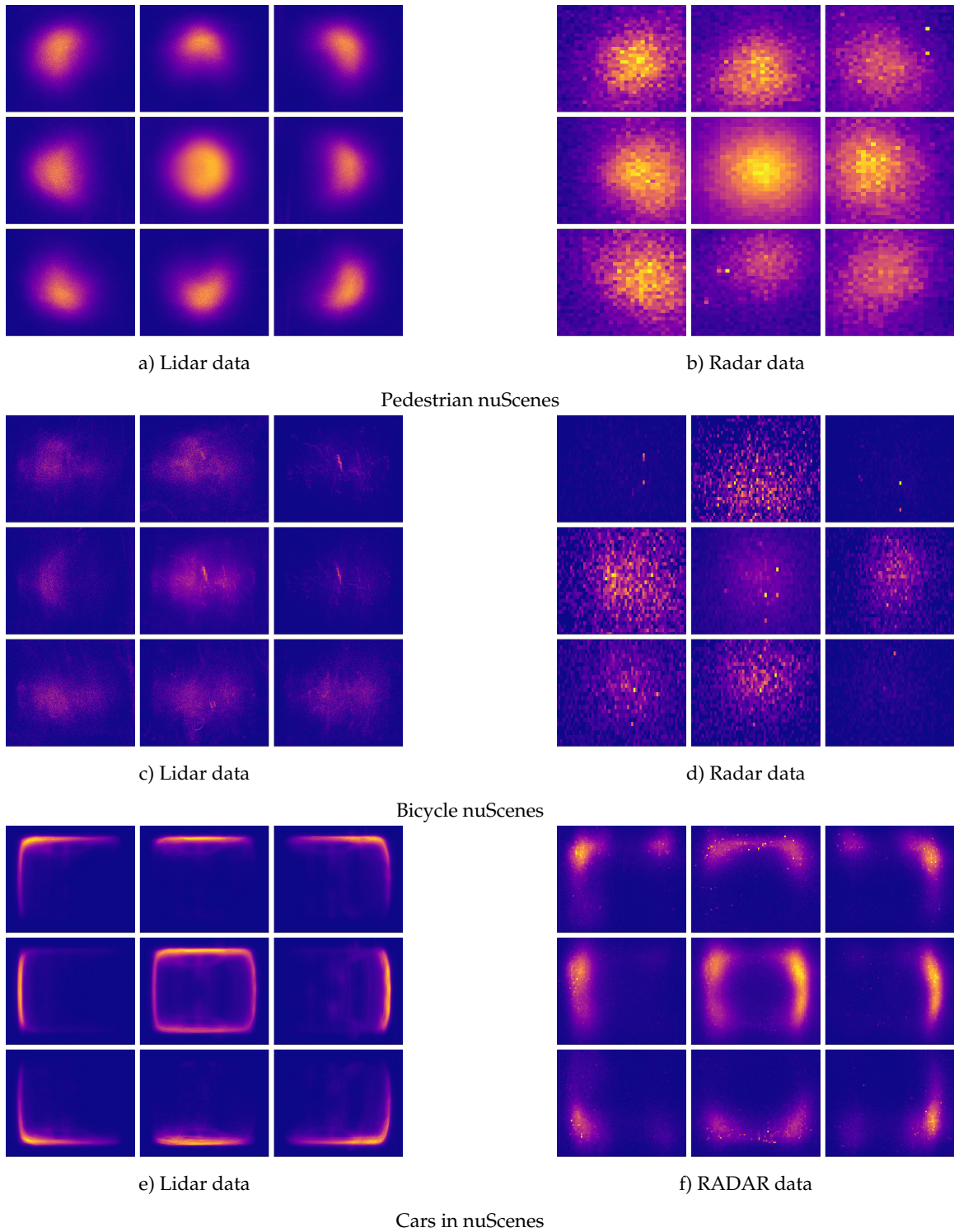
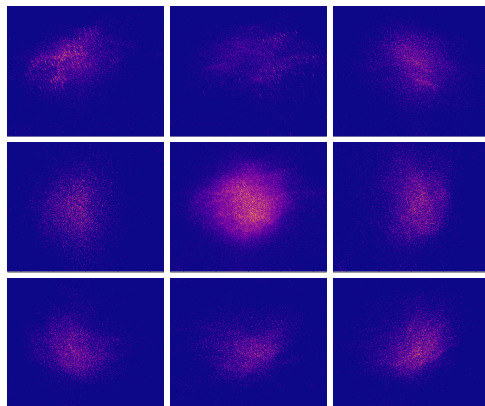
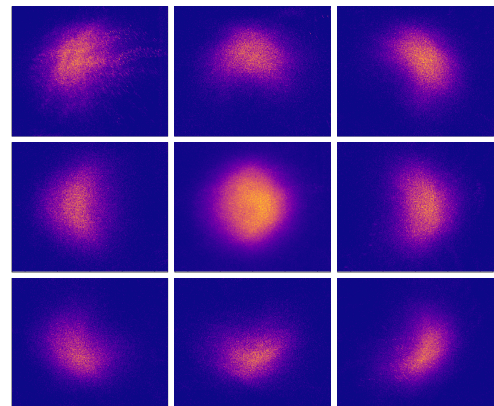


Figure 3.6: Accumulated data for the nuScenes data set.

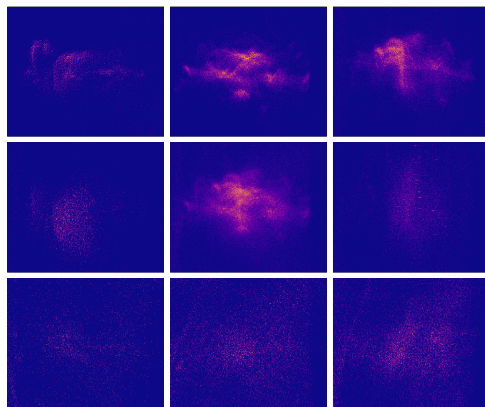


a) Bumper-mounted LIDAR data

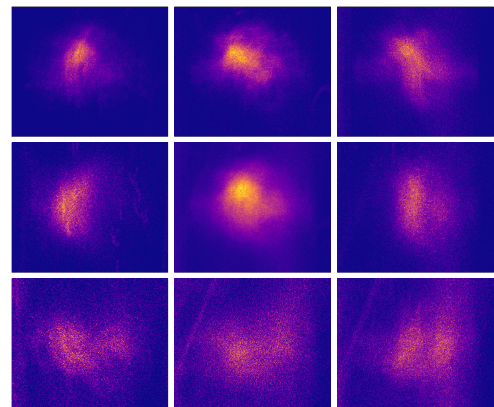


b) Roof mounted LIDAR data

Pedestrian Lyft level 5

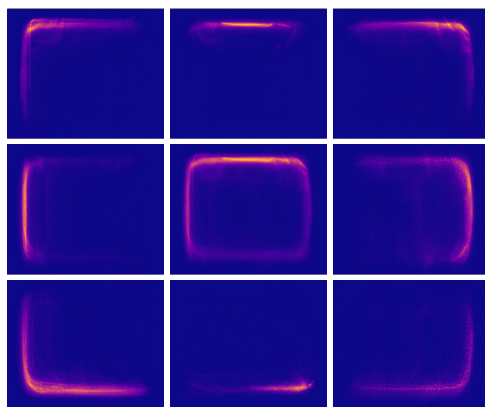


c) Bumper mounted LIDAR data

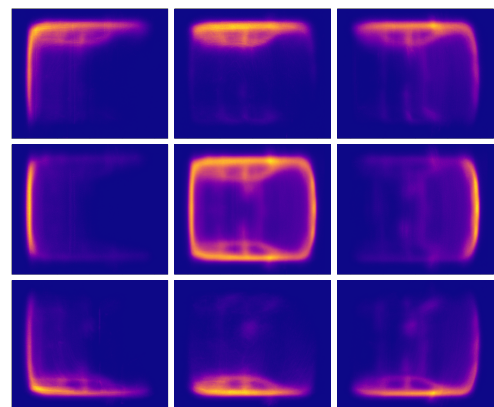


d) Roof mounted LIDAR data

Bicycle Lyft level 5



e) Bumper mounted LIDAR data.



f) Roof mounted LIDAR data

Cars Lyft level 5.

Figure 3.7: Accumulated data for the Lyft level 5 data set.

observation is however found in the conditionals. RADAR measurements, in contrast to the LIDAR sensor measurements, are biased towards the direction of the self occluded side instead of the sensor.

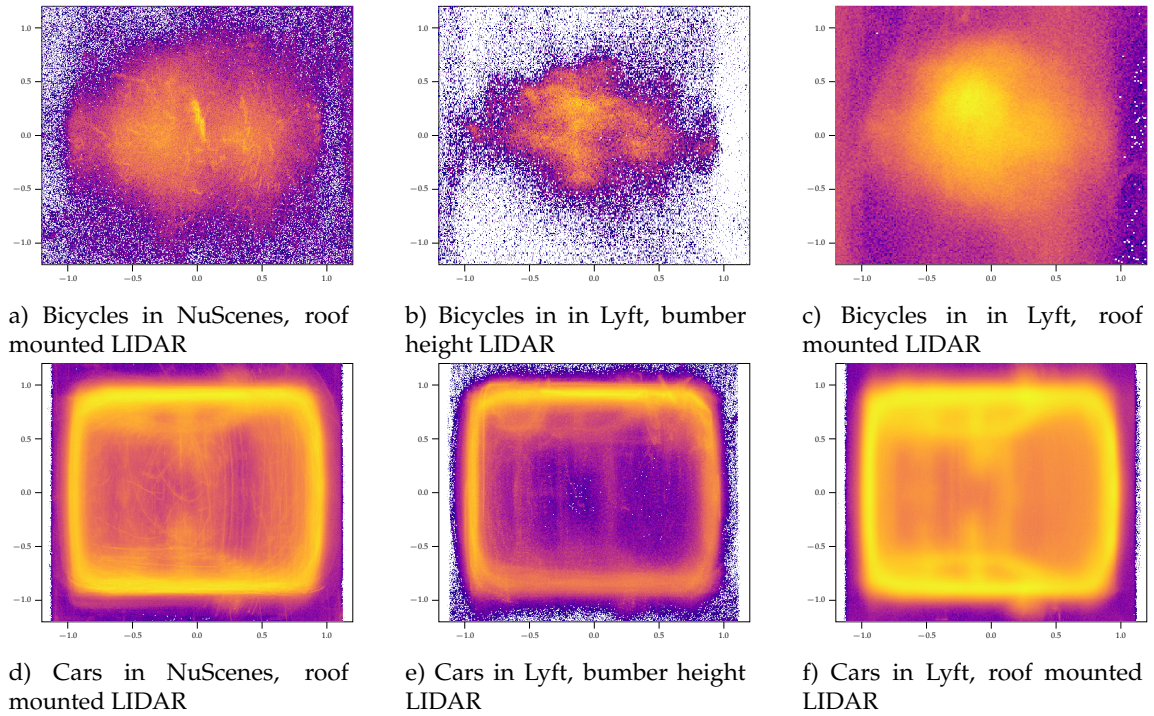


Figure 3.8: Additional histograms for every LIDAR sensor in logarithmic scale for the marginal over the accumulated data for cars and bicycles.

Bicycles The histograms for bicycles are shown in Figures 3.6c, 3.6d, 3.7c, 3.7d, the additional logarithmic scaled heatmaps are found in Figure 3.8. As the number of bicycle measurements is the lowest, we first acknowledge that biases in the visibility ranges are clearly available, especially for the conditionals 4, 6 and 8 in the nuScene’s RADAR data and the conditionals 1, 2, 3 and 5 in the Lyft level 5’s front LIDARs only sparser measurements. We note that LIDAR and RADAR behave significantly different. All LIDAR sensors provide visible structures from the driver and bicycle with respective self occlusions based on the aspect angle. The relative distribution on the bike and the driver is further changing as the driver provides a larger surface in conditionals 6 and 8. No such structure can be found in the RADAR data. The data represents a mostly unstructured distribution around the center, while general biases towards the occluded side are visible in the conditionals.

Cars Cars are rigid targets in contrast to the other target types with freely moving limbs. The data for these sensors is shown in Figures 3.6e, 3.6f, 3.7e, 3.7f, the additional logarithmic scaled heatmaps

are found in Figure 3.8. The data in all LIDAR sensors provides strong measurements from the targets' 2d boundary as it provides the largest horizontal surface. An astonishing feature in this accumulation is the clear boundary and the structure of the corners visible in scaled coordinates. As we scaled a large number of different cars with different length a larger variance in these features would be expected. The data further shows that additional structures are available in the measurement distribution for the mounting position and number of layers, specifically in the Lyft Level 5 data set but also in the nuScenes data set. We can clearly see structures of the upper part of the vehicle such as the A, B, and C Pillar and parts of the roof in the histogram for the roof-mounted sensor while they are essentially not captured by the sensors mounted in bumper height. In the LIDAR sensor of the nuScenes data set, these structures are also visible, but the dominant part of measurements still stems from the targets' boundary.

3.6 Discussion Of Target Types And Model Requirements

With the statistical analysis of the general extent and measurement distribution, we can discuss requirements for extended target models for each analysed target type.

Based on the bounding box statistics, pedestrians may be modelled with a small extent, which seems to be correlated in length and width. The complete accumulated pattern seen in scaled coordinates provides an approximately circular measurement distribution. The conditionals suggest that, with the relative orientation, a partial view on this circle is found in LIDAR, where based on self-occlusion, only the side directed to the camera is visible, while an inverse effect is visible in RADAR data. The distribution of pedestrians with respect to the aspect angle, in essence, represents crescent density for LIDAR and skewed ellipses for RADAR. This allows for the use of ellipse models, but a bias will be introduced due to the self-occlusion represented. However, the error introduced is small in relation to the true extent and could be incorporated as a symmetry effect or bias on the extent with respect to the sensor position. We further note that the partial visibility based on the sensor field of view makes decoupling the motion direction from the orientation of the shape desirable as the distribution is not oriented with the motion direction. The extent models suitable for this are the elliptical models like the RM approach [Koc08, FFK10] and newer models like the GGIW [GO12] or the MEM-EKF* [YB19]. We emphasize that a model that may handle single measurements well would be recommended. We further note that, for RADAR data perceived with a higher distance tracking the extent will mostly state a problem because mostly a single measurement will be perceived from the target.

Bicycles are a complex type in comparison to pedestrians, as they are a combination of two objects. We first note that the general length and width suggest a strong difference in the targets dimensions. For LIDAR data a structured distribution of measurements may be found. While the general measurements are distributed on both objects, a more significant amount is generated from the pedestrians body as it vertically occupies more space. The bicycle, however, is still providing

us with a significant amount of measurements which further allows for the reception of a clearly elongated target with smaller width on the front and back. For RADAR data, no clear structure was found. We still see penetration effects, especially on the side views, but the analysis suggests that skewed elliptical distributions are found. The target's orientation is here similar to the target's general motion direction which allows them to be coupled. Thus, the distribution of bicycles can again be described by an ellipse for different reasons than the pedestrian. The extent models suitable for this are the already mentioned elliptical models. This time we are suggesting the use of a motion model that ties orientation and main axis of the ellipse together while higher moments of the orientation change can be considered, such that e.g. a polar constant velocity or coordinated turn model might be suitable. This is, however, not directly supported in the typical ellipse models as extent and kinematic parameters are estimated without correlation.

The distribution for cars is rigid and more structured than the distribution of the former two target types. Based on the scaling behaviour, an anisotropic scaling model is desirable as length and width are, in general, independent. For LIDAR data, the general boundary of cars can be described as a rectangle with rounded corners. The corners are significant features that should be included. We further note that a set of secondary sources on the volume of the target is available due to the structures identified on the upper part of the vehicle. These pose a measurement distribution that needs to be incorporated into the target model to avoid biases in the estimation of the size of the target. Cars based on RADAR data, however, pose a completely different scenario regarding the measurement origin. While the LIDAR data provides measurements from the complete car's 2d boundary and structure on the volume due to the projection of 3d structures, measurements from RADAR stem from specific scatter sources. Thus, the scenario provides measurement origins that are neither just the surface nor the general volume. Instead, a model with distinct measurement origins based on the accumulated measurements is desirable.

LIDAR Extent Modelling And Tracking

Contents

| | |
|--|----|
| 4.1 Problem Setting | 41 |
| 4.2 State Of The Art: LIDAR Extended Target Models | 42 |
| 4.3 A Spline Based Automotive Contour Model | 43 |
| 4.4 Modelling the Measurement Distribution With Asymmetric Noise | 58 |
| 4.5 Evaluation | 62 |
| 4.6 Discussion Of The Spline Model And The Asymmetric Noise Assumption For Cars In LIDAR Data | 75 |

Remark. This chapter is based on the conference publication [2] which create a cartesian B-spline model for automotive vehicle tracking based on LIDAR data and compares it with Gaussian process based extended target models with respect to model requirements. It also includes the asymmetric measurement noise heuristic for the data from roof mounted LIDAR sensors to reduce estimating biases based on the conference publication [3].

4.1 Problem Setting

We aim to create an extended target model that captures the features of a car provided in LIDAR scans. Since the data consist of dense point clouds, we are aiming mainly to create a model which incorporates a robust approximation of the vehicle's shape and measurement distribution on the target. In addition, we want this model to be constrained to the features required without creating an unnecessary degree of freedom.

Based on the accumulated data from Chapter 3 we can identify the rounded corners of the vehicle as the most prominent feature that the model needs to incorporate, while the sides and the front

are in general straight lines. We further determined that an independent scaling of length and width is desired to capture the variance in the vehicles' extent. Thus we are looking for a state that is described by the shape representation with

$$\mathbf{x}_{ext} = [l, w]^T, \quad (4.1)$$

and a shape model that can describe the desired shape features relative to these parameters. These requirements cover the geometric features of the model. However, the distribution of measurements needs to be covered with respect to the sensor's mounting position. The data provided from a low mounted LIDAR scanner mostly consists of measurements from the 2d boundary of the vehicle and, therefore, conforms with typical extended target tracking assumptions. The measurement distribution for a sensor with a high mounting position differs significantly from one with a low mounting position. Measurements from specific areas of the 3d boundary are projected into the 2d coordinates which generate substantial measurements on the targets volume which are not evenly distributed. This distribution is rather uncommon and needs to be understood well to avoid the introduction of biases.

4.2 State Of The Art: LIDAR Extended Target Models

We listed the general extended target models of interest in Section 2.4 but emphasize on automotive LIDAR based related work in this section. Tracking of extended targets based on LIDAR data is a topic of interest for the scientific community as well as the automotive industry with respect to automated driving assistance functionality of higher Society of Automotive Engineers (SAE) levels. In this area different model types are employed in state of the art research.

First, there are models with simple shape representations which describe the target's extent with a birds eye view projection of measurements. In [GLO11] the authors explored dedicated extended target models for different traffic participant types in LIDAR data. For pedestrians they proposed the use of an elliptical extent model and for cars the authors provided a rectangle approximation of the shape. In [GRMS14] another rectangular extended target model was used to generate the origin of LIDAR measurements on the target's extent to incorporate expected measurement sources and exploit LIDAR sensor properties, like the implicit order of measurements, in the source association. In [LOG10] a Gaussian mixture Probability Hypothesis Density (PHD) filter for extended target tracking has been investigated on simulated data and in [BRG⁺15] the tracking of pedestrian targets in automotive LIDAR data has been explored using the Generalized Labeled Multi-Bernoulli (GLMB) with the GGIW as a target model.

Then, there are complex shape models applied to laser scanner data in the birds eye projection. With prime examples of the RHM using the Gaussian process model which have been deployed for vehicle tracking in [HSRD16] as well as in [WÖ15]. These works use, in general, free form

star-convex models to estimate the target's extent. In addition to the planar case, extended target models for automotive targets in 3d space have been explored in [EW19] and [KÖ19], in which 3d Gaussian processes have been deployed to model the target's extent. Additionally, a 3d unclosed non-uniform rational basis spline (NURBS) surface was used instead to model the extent in [NBW19].

These models are typically simplified shapes with rectangular or elliptical approximations or they are essentially free shape models which have an increasing degree of complexity with increasing amount of extent parameters and dimensions. Rectangular approximations make the introduction of prior knowledge of specific shape parts such as the identified corners very hard. While radial functions with more expressiveness require a large number of free shape parameter which all require to be tracked and updated, even though the shape is approximately known in advance. We further note that constraining a radial function to a desired shape is possible but tedious due to the radial function parameters that model a Cartesian contour and the non-Cartesian definition makes the introduction of anisotropic scaling very complex with respect to the shape parameter.

4.3 A Spline Based Automotive Contour Model

To model the shape of extended vehicles with little complexity but the necessary expressiveness, we propose to use a constrained spline surface as the contour function of an RHM [2]. To model and track contours on dense data, such as images, splines have been used to great success [BI12].

We will start by stating a general spline contour model for RHMs and provide an update strategy for dense data. On this free spline model, we introduce constraints forcing it to retain star-convex shapes while keeping the shape features that are visible in the collected vehicle data.

4.3.1 B-Spline Model

A spline is, in general, a function consisting of polynomial segments of order d . While multiple different representations of varying complexity can be defined, we will focus on B-splines.

B-splines $B(\tau)$ of order d are described as the weighted sum of N_b basis functions $B_{n,d-1}(\tau)$, $n \in \{0, \dots, N_b - 1\}$, such that

$$B(\tau) = \sum_{n=0}^{N_b-1} B_{n,d-1}(\tau) \cdot P_n, \quad (4.2)$$

with weights P_n and a running parameter $\tau \in T := [0, N_b]$ and with the additional constraint that

$$\sum_{n=0}^{N_b-1} B_{n,d-1}(\tau) = 1, \forall \tau \in T. \quad (4.3)$$

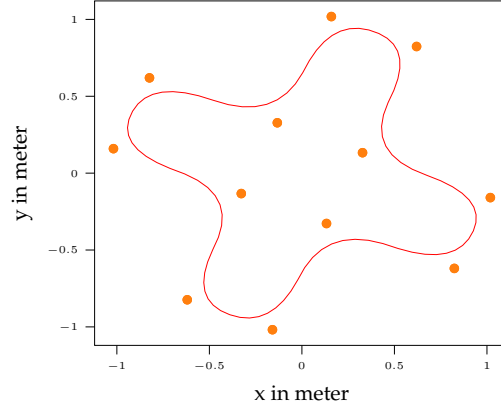


Figure 4.1: Example of a closed quadratic B-spline. We show the weights P_n as orange dots and the corresponding surface $B(\tau)$ in red.

This sum can be expressed as a matrix multiplication

$$B(\tau) = [P_0, \dots, P_{N_b-1}] \cdot B_\tau, \quad (4.4)$$

with $B_\tau = [B_{0,d-1}(\tau), \dots, B_{N_b-1,d-1}(\tau)]^T$. An example of a closed Quadratic B-spline is given in Figure 4.1.

Here, each basis function is defined with finite support over segments of the running parameter space. These segments are defined by their start and end point via *knots* $\{k_0, \dots, k_{N_b}\}$ with $k_j \in T, \forall j$.

In the following, we will list a selection of properties that are relevant for the context of this thesis. Detailed references can be found in [BI12].

- A B-Spline is called *uniform*, if all *knots* are equally spaced over the interval $[0, N_b]$.
- A B-Spline is called *periodic*, if the basis function, weights and the corresponding *knots* are defined *periodically*, such that the last knots act as copies of the first. In order to model the transition in the period an additional $d - 1$ *knots* are required, i.e. a total of $N_b + d - 1$, with d the order of the polynomial represented by $B_n(\tau)$. However, $B_{N'_b+i}(\tau) = B_i(\tau)$ and $P_{N'_b+i} = P_i$ for all $i \in \mathbb{N}$ with the altered number of knots

$$N'_b = N_b + d - 1. \quad (4.5)$$

Note that in the context of *periodic* functions we will refer to N'_b as N_b for the sake of simplicity.

- B-splines possess $d - 1$ continuous derivatives. In the case of quadratic B-splines, we enforce one continuous derivative.

- Splines allow to reduce their order and, in turn, its continuity at the connection between two segments by assigning the same weights to two consecutive basis functions such that the same *knot* is defined twice. For example defining the two of the three active knots in a quadratic B-spline, thus a B-spline of order 3 as the same knot results in a line segment. Thus, polygon contour models are a special case of quadratic spline models where each weight is defined twice, resulting in line segments.

In the general case the Coxde Boor recursion [DB72] can be used to construct the B-spline of order d as

$$B_{i,0}(\tau) = \begin{cases} 1, & \text{if } k_i \leq \tau \leq k_{i+1} \\ 0, & \text{otherwise} \end{cases} \quad (4.6)$$

$$B_{i,d}(\tau) = \frac{\tau - k_i}{k_{i+d} - k_i} B_{i,d-1}(\tau) + \frac{k_{i+d+1} - \tau}{k_{i+d+1} - k_{i+1}} B_{i+1,d-1}(\tau) . \quad (4.7)$$

However, for the remainder of this paper we will focus on a contour definition with quadratic B-splines. Therefore, we will omit the index for the order such that $B_n(\tau) = B_{n,2}(\tau)$.

Evaluating the recursive formulation in Equation (4.7) for quadratic basis functions leads to a simple definition of a uniform quadratic basis function [BI12], such that

$$B_n(\tau) = B_0(\tau - n), \quad (4.8)$$

$$B_0(\tau) = \begin{cases} 0.5 \cdot \tau^2, & \text{if } 0 \leq \tau < 1 \\ 0.75 - (\tau - 1.5)^2, & \text{if } 1 \leq \tau < 2 \\ 0.5 \cdot (\tau - 3)^2, & \text{if } 2 \leq \tau < 3 \\ 0, & \text{otherwise,} \end{cases} \quad (4.9)$$

with N_b knots defined at the integers $n \in \{0, 1, \dots, N_b\}$.

For the basis function $B_0(\tau)$, the defining knots are $[0, 1, 2, 3]$ while the defining knots for each other basis function are shifted by their respective index.

Here, we note that exactly three basis functions are active for any given τ . An example of the basis functions of a periodic quadratic B-spline is given in Figure 4.2. With these basis functions, a contour in the dimensions of the weights can be constructed.

4.3.2 Free B-Spline Model

With the definitions in Section 4.3.1 we can formulate a Cartesian RHM contour function $C(\tau)$ via periodic uniform quadratic B-splines, which are of order 3.

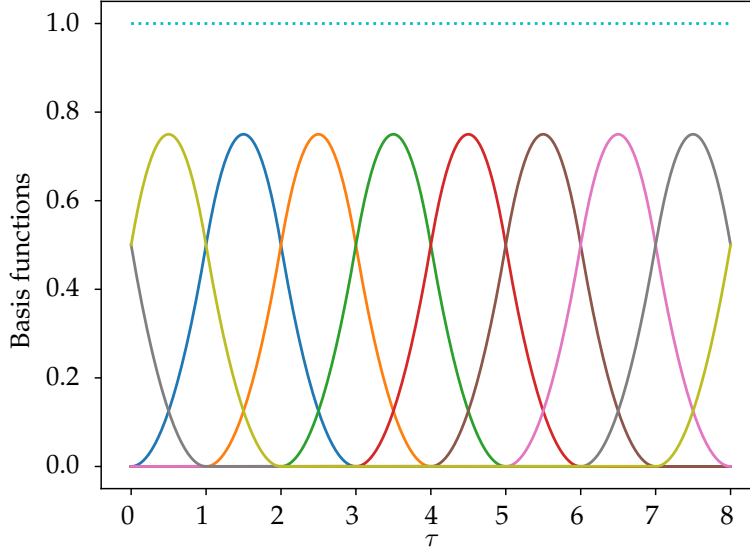


Figure 4.2: Eight basis functions in different colors over the space of τ . The sum over all basic functions is shown as the dotted line illustrating the sum property.

In contrast to typical RHM the running parameter τ will be used as a nuisance parameter in contrast to the typical radial definition via an angle.

The surface in local coordinates is, in turn, defined via

$$C(\tau) = \mathbf{P} \cdot B_\tau = \begin{pmatrix} P_{x,0}, \dots, P_{x,N_b-1} \\ P_{y,0}, \dots, P_{y,N_b-1} \end{pmatrix} \cdot B_\tau, \quad (4.10)$$

where \mathbf{P} are the Cartesian weights which will be referred to as *basis points* consisting of the x and y coordinates, such that a single basis point is determined by $P_{x,i}$ and $P_{y,i}$.

In the context of extended target tracking, we are provided with extent parameters

$$\mathbf{x}_{ext} := \mathbf{P}, \quad (4.11)$$

which contain the ordered weights of the spline in x - and y -dimension and fully define the shape of the contour.

This definition of spline-based contours can, in general, represent non star-convex shapes as it is done in image processing [BI12]. An update with contour fitting approaches can be formulated, assuming that an appropriate measurement association scheme can be provided. An example is the use of the PMHT update procedure with discretised surface points from our prior work [6].

4.3.3 Free B-Spline Update With The PMHT

Using the update approach in our prior work [6] and using the PMHT formulation for extended target tracking from Section 2.4.3, we may state a set of fix measurement origins determined by specified run parameters $[\tau_1, \dots, \tau_n]$. The n hypotheses on the spline surface may be assumed to generate measurements with equal prior probability $\pi_i = \frac{1}{n}$. It is then straightforward to state the necessary measurement equation for a single measurement \mathbf{z} using the contour function from Equation (4.10):

$$\mathbf{z} = C(\tau) + \mathbf{w} \quad , \quad (4.12)$$

with the assumption of additive Gaussian white noise, such that $\mathbf{w} \sim \mathcal{N}(0, R)$.

To illustrate the free spline model's behaviour, we used a simple non star-convex surface modelling the letter H. We randomly generate 20 equally distributed measurements from the surface per time step, which are corrupted with zero mean Gaussian white noise with different covariance sizes $R = \text{diag}(0.1^2, 0.1^2)$, $R = \text{diag}(0.2^2, 0.2^2)$, $R = \text{diag}(0.5^2, 0.5^2)$ and $R = \text{diag}(1, 1)$. We further set the batch size in the update to one for the sake of simplicity as it leads to a simple Kalman filter update as the measurement equation based for a fix source, denoted by τ , is linear as stated in Equation (4.10). The process noise covariance matrix that is described by a diagonal matrix where the variance 0.01^2 is used for all state extent parameters in between updates, and we initialize the basis points on a circle of radius 1. The results for all three noise terms after 200 time steps is shown in Figure 4.3.

The shape in the scenarios is well represented in the first three scenarios. In the fourth scenario, no shape could be extracted anymore as the noise's standard deviation exceeds the size of each volume segment and makes the determination of the features very hard. Here, only a generic rectangular shape can be determined.

We see that this strategy provides acceptable estimates on simple scenarios concerning different noise terms. However, the model has three significant drawbacks, as is expected with free shape models.

First, formulating the model such that it works relative to a position is not trivial as defining this point is very complicated for free form models. The typical star-convex RHM [BH11] uses the radial definition of the contour, which forces it to provide a relative shape with respect to the center of the target. Introducing such constraints to a free moving contour in the same space as the center point is not trivially possible, as it makes the relation between any basis point and a potential center point ambiguous, as their relation is, without further constraints, linear. The ambiguity, in turn, only depends on the choice of noise terms and does not guarantee that the correct interpretation can simply be kept. On this note, it is also important to mention that no constraint removes the

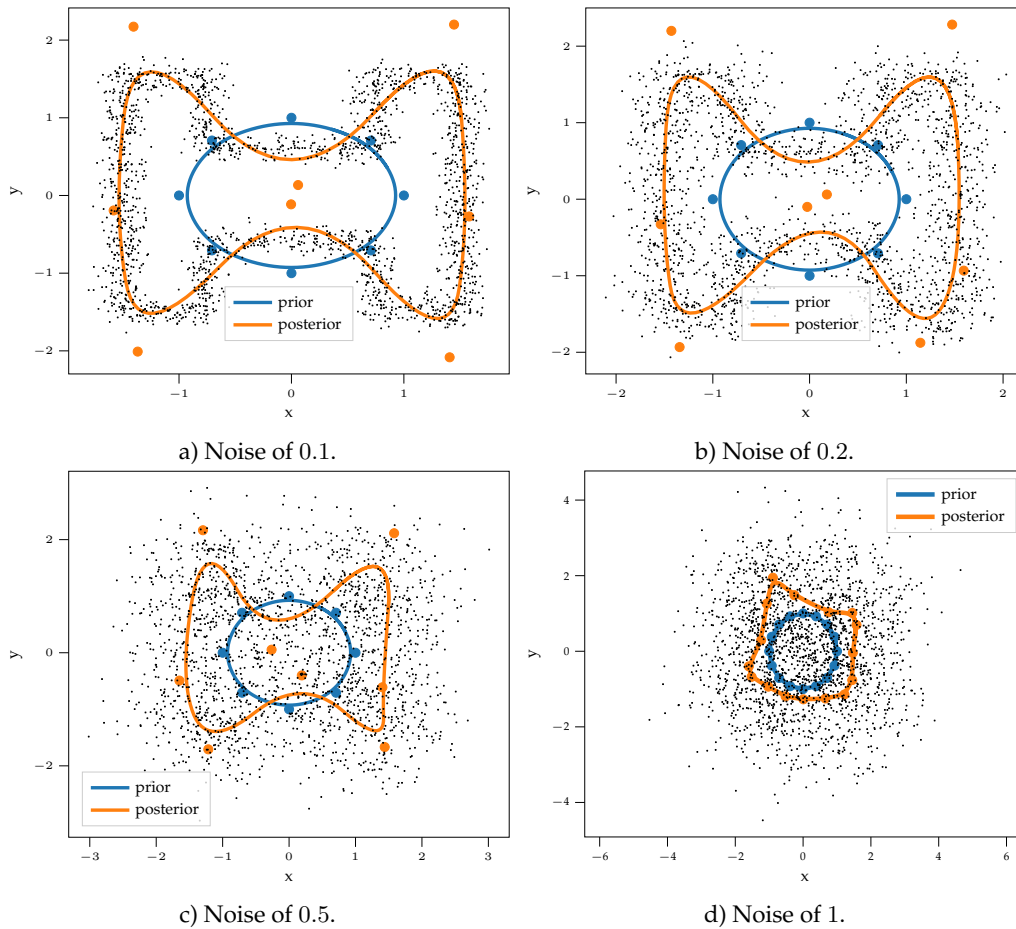


Figure 4.3: Measurement noise scenarios over 200 time steps with 10 measurements on each instance, with an illustration of the prior on the first and posterior of the last time step. These estimates include the basis points and contour.

self-intersection of the contour. This is a problem that may even occur in rare cases in typical radial RHMs, if a negative radius occurs, which was noted in [WÖ15].

Second, using the volume scattering has to be adapted significantly, since the radial relation makes approximations of a uniform distribution possible. Solving this issue would need to lean on a similar approach to level set RHMs used for polygon tracking approaches presented in [ZFH16].

The third problem is the determination of the number of necessary base points to describe a given shape, as too few points do not allow for a precise representation of the shape. At the same time, too many points can easily lead to overfitting, which in turn may lead to self-intersections or unassociated segments as well as computational load.

While solving these problems is theoretically appealing, we will focus on the application of the expressive spline formulation to the tracking of vehicles and introduce prior shape knowledge, which includes the star-convex property of the target.

4.3.4 A Constraint B-Spline Contour Function

Constraining a spline to a single desired shape is trivial. By placing an appropriate amount of basis points in desired places with respect to the target's position, we can introduce handcrafted shape features. Thus, using splines to model the rounded corners of the vehicle can be done with relative ease by generating a model that fits the contour of accumulated measurements from Section 3.5 with a minimal number of state parameters but the desired geometrical representation. An example of a minimal model with rounded corners is provided in Figure 4.4 in which we orient ourselves on the boundary visible in the accumulated data for cars in the unit coordinate system from the data analysis in Section 3.5, specifically the marginals. The model uses 8 basis points allowing for the design of sides and corners.

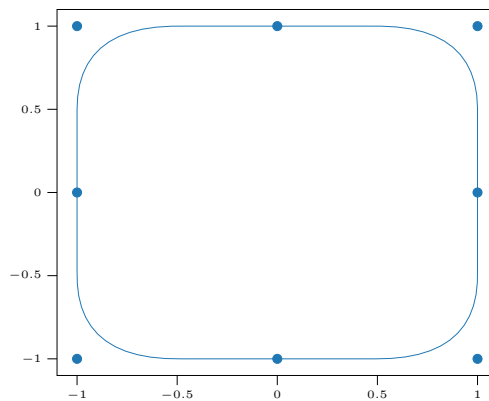


Figure 4.4: Example of a minimal spline model representing the desired car model.

However, the general treatment of the basis points as state parameters would inherently lead to the same problems we described in the free spline model. Constraining the points with respect to the dimension of the general shape is desirable. It would reduce the problem and number of actual state parameters, as the general shape is known and already well described in scaled coordinates.

To introduce the constraints required for the contour function, we focus on a spline surface that uses a fixed set of basis points with respect to a reference coordinate system. Thus, instead of keeping the basis points as free state parameters, they are constant, and the only extent parameters are the parameters of the parent system.

To incorporate the desired anisotropic scaling of the model, we choose the basis points as relative positions on a bounding box where the bounding boxes half length l and width w are kept as free

state parameters. Thus, the state can be represented as

$$\mathbf{x}_{kin} = [\mathbf{m}^T, \phi]^T, \quad (4.13)$$

$$\mathbf{x}_{ext} = [l, w]^T, \quad (4.14)$$

where the kinematic parameters are the position $\mathbf{m} = [x, y]$ and ϕ as the target's yaw angle. This state provides us with the definition of a yaw oriented bounding box, which may be extended to a coordinated turn motion model as stated in Section 2.5 by introducing higher moments of the kinematic parameters with the radial velocity and yaw rate. The higher parameters are omitted as they are not relevant for the extent model.

This state definition allows our contour to be described as scaled versions of a set of fix basis points with

$$\mathbf{S} C(\tau) = \mathbf{S} \mathbf{P} \mathbf{B}_\tau, \quad (4.15)$$

where $\mathbf{S} = \text{diag}(l, w)$ is the scaling of the bounding box itself.

Based on this equation, we can state a general measurement equation for a measurement from the surface as

$$\mathbf{z}_i = h(\mathbf{x}, \tau_i) + \mathbf{w} \quad (4.16)$$

$$= \mathbf{R}_\phi \mathbf{S} C(\tau) + \mathbf{m} + \mathbf{w}, \quad (4.17)$$

where \mathbf{w} again represents additive Gaussian white noise with $\mathbf{w} \sim \mathcal{N}(0, \mathbf{R})$ in which \mathbf{R} is the measurement noise covariance matrix and where \mathbf{R}_ϕ is the 2 dimensional rotation matrix for the yaw angle ϕ with

$$\mathbf{R}_\phi = \begin{pmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{pmatrix}. \quad (4.18)$$

We note that other coordinate systems and alternative relations based on the geometries of targets could be introduced, but for modelling an appropriate shape for other targets or scaling behaviour, we keep the system simple as it fits the desired shapes. We further note that as long as the basis points are chosen as a convex shape, every anisotropic shape scaled with $l > 0$ and $w > 0$ is a convex shape. This behaviour is due to the fact that every segment of the spline is contained in the convex hull determined by the three active basis points. Thus, we can immediately incorporate features from a star-convex RHM such as the use of a scaled surface for the representation of volume scattering. While the shape can now be represented as a star-convex RHM we are missing an efficient measurement-to-surface association technique as the running parameter of our model

differs from classical radially defined RHMs. Thus, we need to define a consistent association procedure for our running parameter.

4.3.5 Surface Point Association

Our model fully describes a specific measurement origin on the boundary of the target via the running parameter τ . The relation between measurement and origin on the target is a known challenge in extended target tracking. The origin of a measurement and, therefore, the respective running parameter τ on an extended target is inherently a hidden variable, as it cannot be captured by a sensor system [GBR17b]. The choice of either a single parameter or a set of parameter hypotheses is necessary for the measurement association to formulate a complete measurement equation for the update of the current shape's estimate.

In typical star-convex RHMs the nuisance parameter for a new measurement z_i is selected by finding the hypothesis τ_i and the corresponding position on the contour of the target $h(\tau_i, x)$ which minimizes a defined score

$$\tau_i := \arg \min_{\tau} f(\tau, \mathbf{z}_i) \quad , \quad (4.19)$$

with respect to a given score function $f(\tau, \mathbf{z})$.

As most star-convex shape models use a radially defined contour model [BH09, WÖ15], this definition can be exploited to choose a convenient score function $f(\cdot)$ via the radial difference of origin and measurement. Thus, an association can be performed via the relative angle from the shape's center to the measurement. This association is convenient as the minimal distance of f can uniquely be found as 0 and is easy to calculate, as the hidden parameter states a quantity that already needs to be calculated. More general associations would be the use of the minimal distance to the shape. However, this approach does not necessarily yield a unique hypothesis as multiple points on the shape can have the same distance and requires optimization algorithms to find a hypothesis for detailed shape models.

Alternatively, multiple hypotheses of measurement origins can be used with respect to the sensor's measurement principle but they introduce an association problem instead [GRMS14].

In this thesis we will derive a radial association scheme with respect to the abstract running parameter τ and establish the relationship between measurement and measurement origin via the association angle. We make this choice for the sake of simplicity and to provide a better comparison to state of the art radially defined models.

Association Derivation

We will consider a single measurement association for this derivation and drop the index i as the association is the same for each measurement. Thus, we define the measurements in global

coordinates as \mathbf{z} and the expected source on the surface as \mathbf{y} . It is convenient to work in the target coordinate system for this derivation. Hence, we will further denote the respective points in the target's local coordinate system as $\bar{\mathbf{z}}$ and $\bar{\mathbf{y}}$, respectively, and state their relation by rotation and translation via the equations

$$\bar{\mathbf{z}} = \mathbf{R}_\phi^{-1}(\mathbf{z} - \mathbf{m}), \quad (4.20)$$

$$\bar{\mathbf{y}} = \mathbf{R}_\phi^{-1}(\mathbf{y} - \mathbf{m}). \quad (4.21)$$

In the following, we are assuming that there is a bijective relation between the angle and the hidden running parameter concerning a single point on the surface of the measurement, which we enforced by the placement of the basis points. Exploiting this assumption, we can determine the knots between which τ lies and find the active basis functions. Thus, we can determine the knot as

$$k = j, \text{ where } \sphericalangle(C(j)) \leq \sphericalangle(\bar{\mathbf{z}}) < \sphericalangle(C(j+1)), j \in [0, \dots, N_B], \quad (4.22)$$

with $\sphericalangle(\mathbf{z}) = \arctan2(z_2, z_1)$ and $\mathbf{z} \in \mathbb{R}^2$. This determines the segment of the spline and imposes the constraint that $k \leq \tau < k+1$. By defining

$$\tau_0 = \tau - k \quad (4.23)$$

such that $0 \leq \tau_0 < 1$ we can plug it into the definition of the basis functions Equation (4.8) which allows for a commonly used reformulation of the contour function as a matrix multiplication [BI12]:

$$C(\tau_0) = \mathbf{P}_k \mathbf{M} [\tau_0^2, \tau_0, 1]^T, \quad (4.24)$$

where $\mathbf{P}_k \in \mathcal{R}$ is a matrix of the three active basis points from the complete matrix from Equation (4.10). Here, the evaluation of the basis functions, shifted with the according knot, leads to the coefficient matrix

$$\mathbf{M} = \begin{pmatrix} 0.5 & -1 & 0.5 \\ -1 & 1 & 0.5 \\ 0.5 & 0 & 0 \end{pmatrix} \quad (4.25)$$

and the vector of the running parameter's polynomial variables $[\tau_0^2, \tau_0, 1]^T$.

Thus, we can write the expected measurement source in local coordinates as

$$\bar{\mathbf{y}} = \mathbf{S} \mathbf{P}_k \mathbf{M} [\tau_0^2, \tau_0, 1]^T = \mathbf{S} C(\tau_0), \quad (4.26)$$

simply by applying the measurement Equation 4.17. We further define the full multiplication of the constant terms as

$$[\mathbf{a}, \mathbf{b}, \mathbf{c}] := \mathbf{P}_k \mathbf{M} . \quad (4.27)$$

With these definitions we can state the association model via the condition that the predicted measurement \mathbf{y} on the surface lies on the vector from the center point $\mathbf{m} = [x, y]$ to the predicted measurement z , which can be expressed via setting the cross product of these vectors to 0:

$$0 = (\mathbf{y} - \mathbf{m}) \wedge (\mathbf{z} - \mathbf{m}) \quad (4.28)$$

$$= \bar{\mathbf{y}} \wedge \bar{\mathbf{z}} \quad (4.29)$$

$$= (\mathbf{aS} \wedge \bar{\mathbf{z}})\tau_0^2 + (\mathbf{bS} \wedge \bar{\mathbf{z}})\tau_0 + (\mathbf{cS} \wedge \bar{\mathbf{z}}) \quad (4.30)$$

$$=: u_{\mathbf{a}}^{\mathbf{S}}\tau_0^2 + u_{\mathbf{b}}^{\mathbf{S}}\tau_0 + u_{\mathbf{c}}^{\mathbf{S}} , \quad (4.31)$$

with

$$u_{\mathbf{q}}^{\mathbf{S}} = \mathbf{qS} \wedge \bar{\mathbf{z}}, \quad \mathbf{q} \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}\} . \quad (4.32)$$

Note that in the second line we exploited that the cross product is invariant under the rotation of both vectors. (This can easily be seen from $\mathbf{a} \wedge \mathbf{b} = \|\mathbf{a}\|\|\mathbf{b}\| \sin(\psi)$.)

The statement in Equation (4.31) provides us with a quadratic polynomial with respect to τ , which provides different solutions based on the parameters of the polynomial. Here, we specifically need to treat the linear and quadratic cases.

The solution for the linear case, i.e. $u_{\mathbf{a}}^{\mathbf{S}} = 0$, is

$$\tau_0 = -\frac{u_{\mathbf{c}}^{\mathbf{S}}}{u_{\mathbf{b}}^{\mathbf{S}}} . \quad (4.33)$$

For the quadratic case ($u_{\mathbf{a}}^{\mathbf{S}} \neq 0$) we find

$$\tau_0 = \frac{-u_{\mathbf{b}}^{\mathbf{S}} \pm \sqrt{(u_{\mathbf{b}}^{\mathbf{S}})^2 - 4u_{\mathbf{a}}^{\mathbf{S}}u_{\mathbf{c}}^{\mathbf{S}}}}{2u_{\mathbf{a}}^{\mathbf{S}}} . \quad (4.34)$$

In order to determine the sign in (4.34) we use $\bar{\mathbf{y}} \cdot \bar{\mathbf{z}} > 0$.

This definition fully describes the association method and, therefore, the relation of state and measurement. With this formulation, we can state the parameter τ and therefore the measurement source on the contour for a given measurement based on the radial association.

4.3.6 Spline Model: Jacobian

For tracking an extended target with the B-spline model, the implementation of a Bayesian filter is necessary. Due to the non-linearity of the spline formulation, either an UKF or an EKF can be applied. For this purpose we will implement an EKF as described in Section 2.3.1.

The EKF in turn requires the calculation of the predicted measurement $h(\hat{\mathbf{x}}_k)$ based on the prior estimates mean $\hat{\mathbf{x}}$ which has been defined in the previous section and the measurement model's Jacobian matrix \mathbf{H}_k . In the following, the calculation of the Jacobian will be derived. For the simultaneous integration of the measurements $\{\mathbf{z}_{k,i}\}_{i=1}^{n_k}$ received at time k and the corresponding associated nuisance parameter τ_i for each $\mathbf{z}_{k,i}$ we can state the stacked measurements, measurement noise covariances and stacked measurement sources

$$\mathbf{z}_k = [\mathbf{z}_{k,1}^T, \dots, \mathbf{z}_{k,n_k}^T] \quad (4.35)$$

$$\mathbf{R}_k = \text{diag}(\mathbf{R}_{k,1}, \dots, \mathbf{R}_{k,n_k}) \quad (4.36)$$

$$h(\hat{\mathbf{x}}_k) = [h(\tau_1, \hat{\mathbf{x}}_k)^T, \dots, h(\tau_{n_k}, \hat{\mathbf{x}}_k)^T] . \quad (4.37)$$

In this work, we are going to apply the spline model for constant but scalable star-convex vehicle shapes with line segments for sides and quadratic segments for its corners. We enforce the star-convex shape by placing the *basis points* as a rectangular box. As stated previously, the complete freedom of shape parameters is not desired for tracking vehicles. Instead, as in the case of a box tracker, an anisotropically scalable version of a given contour should be provided such that it is possible to model trucks and small cars alike. Thus, with the corresponding nuisance parameter τ_i for measurement \mathbf{z}_k , the Jacobian of $h(\hat{x}_k)$ in equation (4.37) evaluates to

$$\mathbf{H}_k = \left[\frac{\partial h(\tau_1, \hat{\mathbf{x}}_k)}{\partial \hat{\mathbf{x}}_k}, \dots, \frac{\partial h(\tau_{n_k}, \hat{\mathbf{x}}_k)}{\partial \hat{\mathbf{x}}_k} \right] , \quad (4.38)$$

$$\frac{\partial h(\tau, \hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}_k} = \left[\frac{\partial h(\tau, \hat{\mathbf{x}})}{\partial \mathbf{m}}, \frac{\partial h(\tau, \hat{\mathbf{x}})}{\partial \phi}, \frac{\partial h(\tau, \hat{\mathbf{x}})}{\partial \mathbf{x}_{\text{shape}}} \right] , \quad (4.39)$$

$$\frac{\partial h(\tau, \hat{\mathbf{x}})}{\partial \mathbf{m}} = \mathbf{I}_2 + \mathbf{R}_\phi \mathbf{S} \frac{\partial C(\tau)}{\partial \tau} \otimes \frac{\partial \tau}{\partial m} , \quad (4.40)$$

$$\frac{\partial h(\tau, \hat{\mathbf{x}})}{\partial \phi} = \mathbf{R}_\phi \mathbf{S} \frac{\partial C(\tau)}{\partial \tau} \frac{\partial \tau}{\partial \phi} + \frac{\partial \mathbf{R}_\phi}{\partial \phi} \mathbf{S} C(\tau) , \quad (4.41)$$

$$\frac{\partial h(\tau, \hat{\mathbf{x}})}{\partial x_{\text{shape}}} = \mathbf{R}_\phi \mathbf{S} \frac{\partial C(\tau)}{\partial \tau} \otimes \frac{\partial \tau}{\partial x_{\text{shape}}} + \mathbf{R}_\phi \text{diag} C(\tau) , \quad (4.42)$$

for which a set of partial derivatives are required to be calculated, specifically the derivative of the contour \mathcal{C} with respect to τ and the derivatives of τ by \mathbf{m} , ϕ and $\mathbf{x}_{\text{shape}}$. In the following paragraphs the derivation of the partial derivatives will be stated. We will use $\hat{\mathbf{S}} = \text{diag}(s_y, s_x)$, for the sake of simplicity, such that the scaling factor is applied to the measurement with respect to

the cross product, based on the following reorganisation of terms

$$\mathbf{S}\mathbf{q} \wedge \bar{\mathbf{z}} = \begin{pmatrix} q_x s_x \\ q_y s_y \end{pmatrix} \wedge \begin{pmatrix} \bar{z}_x \\ \bar{z}_y \end{pmatrix} = q_x s_x \bar{z}_y - q_y s_y \bar{z}_x \quad (4.43)$$

$$= \begin{pmatrix} q_x \\ q_y \end{pmatrix} \wedge \begin{pmatrix} \bar{z}_x s_y \\ \bar{z}_y s_x \end{pmatrix} = \mathbf{q} \wedge \hat{\mathbf{S}}\bar{\mathbf{z}} . \quad (4.44)$$

Contour Derivative

The derivative of the contour $C(\tau)$ with respect to τ is based on Equation (4.24), it is straight forward with the application of the reformulation of Equation (4.27) and only requires the derivation of a polynomial, it yields

$$\frac{\partial C(\tau)}{\partial \tau} = \mathbf{P}_k \mathbf{M} [2\tau, 1, 0]^T = 2\mathbf{a}\tau + \mathbf{b}\tau . \quad (4.45)$$

τ Derivatives

The derivatives of the running parameter τ with respect to the state parameters are split into the linear and the quadratic case. In both cases we can state a general derivative with respect to an arbitrary variable g , which can take the form of any shape parameter.

Linear Case The solution for the linear case (4.33) may be rewritten, by moving the scaling factor to the right

$$\tau = -\frac{u_{\mathbf{c}}^{\mathbf{S}}}{u_{\mathbf{b}}^{\mathbf{S}}} = -\frac{\mathbf{c} \wedge \hat{\mathbf{S}}\bar{\mathbf{z}}}{\mathbf{b} \wedge \hat{\mathbf{S}}\bar{\mathbf{z}}} . \quad (4.46)$$

Based on Equation (4.46), we can calculate the derivative of τ with respect to an arbitrary factor g as

$$\frac{\partial \tau}{\partial g} = -\frac{1}{u_{\mathbf{b}}^{\mathbf{S}}} \mathbf{c} \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} + \frac{u_{\mathbf{c}}^{\mathbf{S}}}{(u_{\mathbf{b}}^{\mathbf{S}})^2} \mathbf{b} \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} = \frac{1}{(u_{\mathbf{b}}^{\mathbf{S}})^2} [u_{\mathbf{c}}^{\mathbf{S}} \mathbf{b} - u_{\mathbf{b}}^{\mathbf{S}} \mathbf{c}] \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} , \quad (4.47)$$

which only depends on g in the right-hand side of the cross product requiring the solution for the partial derivative

$$\frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} . \quad (4.48)$$

Quadratic Case For the quadratic case we use the statement of the solution in Equation (4.34) with the reformulation of

$$\theta := \sqrt{(u_{\mathbf{b}}^{\mathbf{S}})^2 - 4u_{\mathbf{a}}^{\mathbf{S}}u_{\mathbf{c}}^{\mathbf{S}}} . \quad (4.49)$$

This allows us to state the partial derivative of τ with regards to the arbitrary factor g such that

$$2 \frac{\partial \tau}{\partial g} = -\frac{1}{(u_{\mathbf{a}}^{\mathbf{S}})^2} \mathbf{a} \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} [\pm \theta - u_{\mathbf{b}}^{\mathbf{S}}] \quad (4.50)$$

$$- \frac{1}{u_{\mathbf{a}}^{\mathbf{S}}} \mathbf{b} \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} \quad (4.51)$$

$$+ \frac{1}{\pm 2\theta u_{\mathbf{a}}^{\mathbf{S}}} \left[2u_{\mathbf{b}}^{\mathbf{S}} \mathbf{b} \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} - 4 \left(\mathbf{a} \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} \right) u_{\mathbf{c}}^{\mathbf{S}} - 4u_{\mathbf{a}}^{\mathbf{S}} \left(\mathbf{c} \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} \right) \right] , \quad (4.52)$$

where we shifted the factor 2 over to the left side for sake of readability.

Hence, we are able to state

$$\pm 2(u_{\mathbf{a}}^{\mathbf{S}})^2 \theta \frac{\partial \tau}{\partial g} = \left[\mp \theta [\pm \theta - u_{\mathbf{b}}^{\mathbf{S}}] \mathbf{a} \mp \theta u_{\mathbf{a}}^{\mathbf{S}} \mathbf{b} + u_{\mathbf{a}}^{\mathbf{S}} [u_{\mathbf{b}}^{\mathbf{S}} \mathbf{b} - 2u_{\mathbf{c}}^{\mathbf{S}} \mathbf{a} - 2u_{\mathbf{a}}^{\mathbf{S}} \mathbf{c}] \right] \wedge \frac{\partial(\bar{\mathbf{z}}\hat{\mathbf{S}})}{\partial g} \quad (4.53)$$

$$= \left[(\mp \theta [\pm \theta - u_{\mathbf{b}}^{\mathbf{S}}] - 2u_{\mathbf{a}}^{\mathbf{S}} u_{\mathbf{c}}^{\mathbf{S}}) \mathbf{a} + u_{\mathbf{a}}^{\mathbf{S}} (u_{\mathbf{b}}^{\mathbf{S}} \mp \theta) \mathbf{b} - 2(u_{\mathbf{a}}^{\mathbf{S}})^2 \mathbf{c} \right] \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} \quad (4.54)$$

$$= \left[([-\theta^2 \pm \theta u_{\mathbf{b}}^{\mathbf{S}}] - 2u_{\mathbf{a}}^{\mathbf{S}} u_{\mathbf{c}}^{\mathbf{S}}) \mathbf{a} + u_{\mathbf{a}}^{\mathbf{S}} (u_{\mathbf{b}}^{\mathbf{S}} \mp \theta) \mathbf{b} - 2(u_{\mathbf{a}}^{\mathbf{S}})^2 \mathbf{c} \right] \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} \quad (4.55)$$

$$= \left[([-(u_{\mathbf{b}}^{\mathbf{S}})^2 + 4u_{\mathbf{a}}^{\mathbf{S}} u_{\mathbf{c}}^{\mathbf{S}} \pm \theta u_{\mathbf{b}}^{\mathbf{S}}] - 2u_{\mathbf{a}}^{\mathbf{S}} u_{\mathbf{c}}^{\mathbf{S}}) \mathbf{a} + u_{\mathbf{a}}^{\mathbf{S}} (u_{\mathbf{b}}^{\mathbf{S}} \mp \theta) \mathbf{b} - 2(u_{\mathbf{a}}^{\mathbf{S}})^2 \mathbf{c} \right] \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} \quad (4.56)$$

$$= \left[([-(u_{\mathbf{b}}^{\mathbf{S}})^2 + 2u_{\mathbf{a}}^{\mathbf{S}} u_{\mathbf{c}}^{\mathbf{S}} \pm \theta u_{\mathbf{b}}^{\mathbf{S}}]) \mathbf{a} + u_{\mathbf{a}}^{\mathbf{S}} (u_{\mathbf{b}}^{\mathbf{S}} \mp \theta) \mathbf{b} - 2(u_{\mathbf{a}}^{\mathbf{S}})^2 \mathbf{c} \right] \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} \quad (4.57)$$

and ultimately

$$\pm 2(u_{\mathbf{a}}^{\mathbf{S}})^2 \theta \frac{\partial \tau}{\partial g} = \left[\left(-(u_{\mathbf{b}}^{\mathbf{S}})^2 + 2u_{\mathbf{a}}^{\mathbf{S}} u_{\mathbf{c}}^{\mathbf{S}} \pm \theta u_{\mathbf{b}}^{\mathbf{S}} \right) \mathbf{a} + u_{\mathbf{a}}^{\mathbf{S}} \left((u_{\mathbf{b}}^{\mathbf{S}} \mp \theta) \mathbf{b} - 2u_{\mathbf{a}}^{\mathbf{S}} \mathbf{c} \right) \right] \wedge \frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} . \quad (4.58)$$

Here, we use the cross product with matrices on the right-hand side. We define this as column-wise cross-products similar to dot products, namely

$$\mathbf{a} \wedge \mathbf{B} = \mathbf{a} \wedge \begin{pmatrix} b_{00} & b_{01} \\ b_{10} & b_{11} \end{pmatrix} , \quad (4.59)$$

such that

$$\mathbf{a} \wedge \mathbf{B} = \mathbf{a} \wedge [\mathbf{b}_0, \mathbf{b}_1] = [\mathbf{a} \wedge \mathbf{b}_0, \mathbf{a} \wedge \mathbf{b}_1] , \quad (4.60)$$

with column vectors $\mathbf{a}, \mathbf{b}_0, \mathbf{b}_1 \in \mathbb{R}^n$. Note that in the case of \mathbb{R}^2 , $\mathbf{a} \wedge \mathbf{b}_i \in \mathbb{R}$, and the above equation yields a row vector.

Analogous to the linear case this only depends on g in the partial derivative in Equation (4.48).

$\partial(\hat{\mathbf{z}}\hat{\mathbf{S}})/\partial g$ -Factors To complete the derivation for both the linear and the quadratic case, we need to calculate the partial derivatives of the form

$$\frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial g} = \hat{\mathbf{S}} \frac{\partial \mathbf{R}_\phi^{-1}(\mathbf{z} - \mathbf{m})}{\partial g} . \quad (4.61)$$

These need to be calculated with respect to our state parameters which we will be grouped as $\mathbf{m}, \phi, \mathbf{x}_{shape}$, for the sake of convenience.

For the center $g = \mathbf{m}$ we find

$$\frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial \mathbf{m}} = \hat{\mathbf{S}} \mathbf{R}_\phi^{-1} (-\mathbf{I}_2) = -\hat{\mathbf{S}} \mathbf{R}_\phi^{-1} . \quad (4.62)$$

The derivative with regards to the yaw angle, with $g = \phi$, yields

$$\frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial \phi} = \hat{\mathbf{S}} \frac{\partial \mathbf{R}_\phi^{-1}}{\partial \phi} (\mathbf{z} - \mathbf{m}) = \hat{\mathbf{S}} \frac{\partial \mathbf{R}_\phi^{-1}}{\partial \phi} \mathbf{R}_\phi \bar{\mathbf{z}} , \quad (4.63)$$

in which the partial derivative of the inverse rotation matrix with regards to the angle is straight forward, with

$$\frac{\partial \mathbf{R}_\phi^{-1}}{\partial \phi} = \begin{pmatrix} -\sin(\phi) & -\cos(\phi) \\ \cos(\phi) & -\sin(\phi) \end{pmatrix} . \quad (4.64)$$

The partial derivative for the shape parameters, with $g = \mathbf{x}_{shape}$, yields

$$\frac{\partial(\hat{\mathbf{S}}\bar{\mathbf{z}})}{\partial \mathbf{x}_{shape}} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \bar{\mathbf{z}} . \quad (4.65)$$

With the association and the derivatives stated we are now able to use the spline model with an EKF to perform a Bayesian updated. This model specifically combines the convenient definition of a Cartesian defined contour with the advantage of a radial association.

4.4 Modelling the Measurement Distribution With Asymmetric Noise

Based on the analysis of the measurement distribution from a roof-mounted LIDAR in Chapter 3 it can be observed that the distribution of the measurements on the target differs significantly for an elevated mounting position on the roof in contrast to a low mounting position on the bumper such as the visibility of the structures changes. While we can assume that measurements stem from the 2d boundary for low mounted sensors, we need to account for the distribution of measurements when concerning roof-mounted sensors.

4.4.1 Distribution Analysis For Radial Random Hypersurface Models

We will evaluate the complex measurement distribution over the targets 2d volume with respect to the use of a radially defined RHM described in Section 2.4.1 in comparison to the typical assumption of measurement origin. The measurement distribution in the RHM is modelled by using a scalar scaling parameter s , which is assumed to be 1 in case of measurement origins on the target's boundary, while the distribution of measurements around this boundary is assumed to be the measurement noise covariance. Alternatively, the squared scaling factor s^2 is assumed to be uniformly distributed in $[0,1]$ [BH11] as described in Section 2.4.

To show the discrepancy between the typical assumptions in literature and the actually present scaling factors, we use the data from the nuScenes accumulated dataset in Chapter 3 in the unit coordinate system. For this data we calculate the scaling factor of each measurement \mathbf{z} with respect to an approximated boundary based on the B-spline model, which is already designed to fit a cars boundary in scaled coordinates, as

$$s_c^{\mathbf{z}} = \frac{\|\mathbf{z} - \mathbf{m}\|_2}{\|\hat{\mathbf{z}}_{radial} - \mathbf{m}\|_2}, \quad (4.66)$$

such that the scaling factor $s_c^{\mathbf{z}}$ calculates as the radial distance from the targets center \mathbf{m} to the measurement divided by the distance from the center to the surface point $\hat{\mathbf{z}}_{radial}$ under the same angle that is posed by the center to measurement. Thus a scaling factor $s_c^{\mathbf{z}} < 1$ indicates that a measurements stems from the interior, while a scaling factor $s_c^{\mathbf{z}} \geq 1$ indicates a measurement on or outside the boundary. We show the approximated boundary over the accumulated data of a measurement to the boundary in Figure 4.5.

The resulting distributions of the scaling factor as well as the squared scaling factor is illustrated in Figure 4.6a and Figure 4.6b respectively.

The histogram of the scaling factor shows a relatively sharp drop-off towards the outside of the vehicle, while a slower drop-off is visible from the peak to 0. This translates to the realization that a large amount of measurements is generated on the target's boundary, which is not exclusive

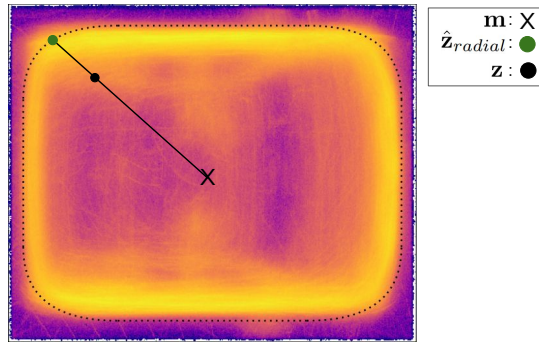


Figure 4.5: Logarithmically scaled heat map of all LIDAR points from cars in the nuScenes dataset in unit coordinates. The black dotted line shows an approximation of the average shape of cars in scaled coordinates and measurement surface point and association to the center are indicated.

as a relevant amount of measurements is generated with a lower scaling factor. Consequently, the squared scaling factor also shows a shift of the mass of the distribution towards the peak, illustrating that it's not feasible to assume a normal distribution on the squared scaling factor. Thus, the typical literature assumption of measurement origin will yield a bias that either will over- or underestimate the target's size with respect to the chosen assumptions as the true distribution of measurements is more complex than a simple uniform distribution over the volume or exclusive measurement origins on the targets boundary.

4.4.2 Proposed Asymmetric Noise Model

Based on the comparison in the previous analysis, we propose to use an asymmetric noise model which incorporates the distribution of the measurements. By interpreting all measurements to have an origin on the target's boundary but different noise terms, we break the symmetry that is typically given with measurements that only originate from the target's boundary. We specifically interpret measurements that are generated towards the target's outside to be generated with standard deviation r_{out} , while measurements that are generated towards the target's volume are interpreted to be generated with measurement noise r_{in} . Under the constraint that $r_{in} \geq r_{out}$ we can enforce a model which incorporates that the measurements on the boundary have a more substantial influence on the target's shape estimate than measurements on the inside. This condition reflects that the measurements are mostly but not exclusively generated on the target's boundary while a tighter distribution of measurements is available on the contour.

We, however, understand that the noise term with which a measurement was generated is inherently a hidden variable in this asymmetric model and needs to be determined. While this problem could be solved by introducing multi modality to treat the hidden assignment, we use a simple heuristic to determine the noise term with respect to the target's predicted boundary. This

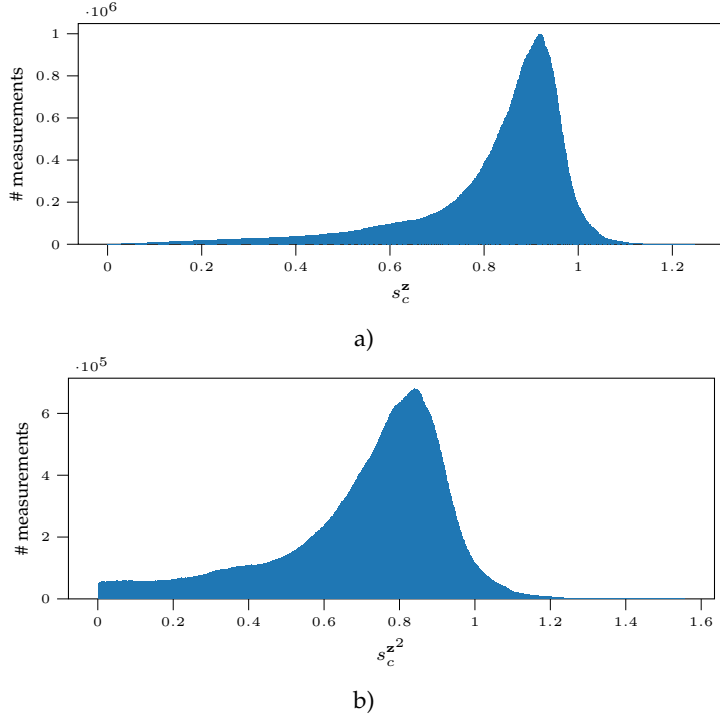


Figure 4.6: Histograms of the scaling factor s_c^z in 4.6a and the squared scaling factor $s_c^{z^2}$ in 4.6b for all measurements from moving cars in the nuScenes dataset.

avoids the arising association problem with multi hypothesis and the explicit generation of these hypotheses.

We assign the noise term based on the relative position of the measurement to the assumed measurement source on the target's boundary, such that measurements on the inside of the predicted boundary are assumed to be generated with a noise term based on the standard deviation r_{in} while measurement on the outside of the boundary are assumed to be generated with r_{out} . The calculation of this criterion can be performed using the normal vector on the assumed measurement source $\hat{\mathbf{z}}$, which can be calculated as the 90° rotated derivative of the contour and the vector from the measurement source to the actual measurement. This is possible as the sign of the dot product of the normal vector and the vector from the assumed noiseless measurement origin to the measurement indicates whether the measurement \mathbf{z}_i lies inside or outside the target's contour.

With

$$g_i = \left(\mathbf{R}_{\pi/2} \frac{\partial C(\alpha, \mathbf{x})}{\partial \alpha} \right) \cdot (\mathbf{z}_i - \hat{\mathbf{z}}_i) \quad (4.67)$$

we find that $g_i > 0$ is equivalent to the measurement lying inside the target's surface and that $g_i \leq 0$ is equivalent to the measurement lying outside. This criterion would also be usable in

three dimensions and only calculates terms that are not computationally heavy. Furthermore, the application of our asymmetric noise model to the Kalman Filter is simple, as only the noise terms need to be adapted depending on the measurement's origin.

The effect of this assignment to surface and volume leads to a stronger reaction to measurements assigned to r_{out} as the noise is smaller; thus, a correction toward these measurements is stronger, which, as the boundary is supposed to be the origin, is the desired behaviour. Thus, this method implicitly acts as a boundary extraction that is not necessarily bound to radially defined measurement models and association functions.

We need to discuss the implications the hard decision of measurement assignment generates as misassignments of measurements on the volume to r_{in} and measurements on the boundary to r_{out} are prone to happen and need to be understood.

First, we discuss the case in which measurements on the volume are assigned to the boundary of the target. Based on the distribution of data and the behaviour of the scanner, this case can only occur if the contour of the prior shape is estimated on the target's inside. In turn, the lower measurement noise and the assumed larger number of measurements on the boundary implicates a shape change towards the measurement and as they reside in the direction of the actual boundary with respect to the prior estimate, thus, towards the boundary of the true target. Moreover, it should be noted that each update forces the estimated boundary further towards the outside. Thus, less of the volume from the inside allows measurements to be assigned to the wrong noise term until only measurements on or close to the boundary are assigned to r_{out} .

Second, we have measurements on the boundary which are assigned to the volume. Again, based on the measurement distribution this can only happen if the measurements are inside the current estimate's contour, implying that the current prior estimate must overestimate the target's boundary. This means that all measurements for this part of the boundary are inside the current estimate, implying a shape reduction as no other residuum towards the outside can be generated. Thus, the shape will have to correct towards the boundary or inside the boundary, where the case of underestimating the boundary will result in the first case of misassignment. We note that this process will be slower due to the larger measurement noise term r_{in} .

In both cases, the estimated boundary will only be able to correct towards the point clouds actual boundary over time, thus avoiding the estimation errors typical assumptions with surface noise or a uniform measurement distribution over the volume would introduce.

4.4.3 Application Of The Asymmetric Noise To Spline Target Model

For this work, the asymmetric noise shall be integrated into the B-spline model. Thus, the classification as inside/outside of target and, therefore, of the assignment to a noise term has to be calculated. As we already calculated the contour derivative with respect to the running parameter

τ , the generic calculation of g_i in (4.67) can be combined with the definition of the B-splines contour derivative in (4.45):

$$\frac{\partial C(\tau_i, \mathbf{x})}{\partial \tau_i} = \mathbf{R}_\phi \cdot \left(\mathbf{S} \cdot \mathbf{P}_k \cdot \mathbf{M} \cdot [2\tau_i, 1, 0]^T \right). \quad (4.68)$$

The criterion in Equation (4.67) is then

$$g_i = \left(\mathbf{R}_{\pi/2+\phi} \left(\mathbf{S} \cdot \mathbf{P}_k \cdot \mathbf{M} \cdot [2\tau_i, 1, 0]^T \right) \right) \cdot (\mathbf{z}_i - \hat{\mathbf{z}}_i). \quad (4.69)$$

4.5 Evaluation

To evaluate the behaviour of the proposed method, we will perform experiments for LIDAR sensors in bumper height and roof-mounted LIDAR sensors.

In Section 4.5.1 basic experiments regarding the spline model's performance are conducted. These are concerning the scaling and tracking performance in comparison to a state of the art Gaussian process model constrained to vehicle shapes.

The next Section 4.5.2 concerns the evaluation of the asymmetric noise heuristic. It covers experiments regarding the inclusion of the asymmetric noise heuristic and special cases of misassignment that may occur. We specifically conduct experiments regarding false initialization and the tracking performance in comparison to the typically deployed surface noise assumption. The experiments are performed on a large selection of instances from the nuScenes dataset [CBL⁺20].

4.5.1 Bumper Mounted LIDAR Experiments

For the evaluation of the proposed spline target model with the assumption of measurements on the contour, we will perform basic experiments with a LIDAR mounted in bumper height. The data stems from a Valeo ScaLa Generation 1. In these experiments we compare the performance of a state of the art handcrafted Gaussian process with a predefined, constrained shape, modelled to represent a typical car's shape based on prior knowledge. Please note that the Lyft Level 5 dataset was not used in this experiment as only key frames are available, which span large time gaps and make the estimation very unreliable, as the prior's mean becomes unreliable, and the state uncertainty gets very large.

Fine-tuning a contour to match a given shape means the tuning of all shape relevant parameters. In the case of a Gaussian process, the angles defining points on the contour have to be chosen, and the distance function needs to be tuned. This is a cumbersome process as the relation between Cartesian and radial coordinates can be very counterintuitive. Another problem is that changing the scaling parameters anisotropically will lead to strong shape distortions such as strong oscillations. Scaling

the complete contour, on the other hand, introduces strong nonlinearities in the measurement equation.

A Cartesian defined model, on the other hand, has an intuitive meaning. Our splines, in particular, may be scaled anisotropically by scaling the basis points in one dimension. The effects of anisotropically scaling the basis angles for a Gaussian process with 20 spanning angles and the eight 2d basis points of our spline model are shown in Figure 4.7. Here, we compare a hand-tuned Gaussian process to our spline model and find that a naïve scaling of the model’s spanning parameters can lead to strong shape distortions.

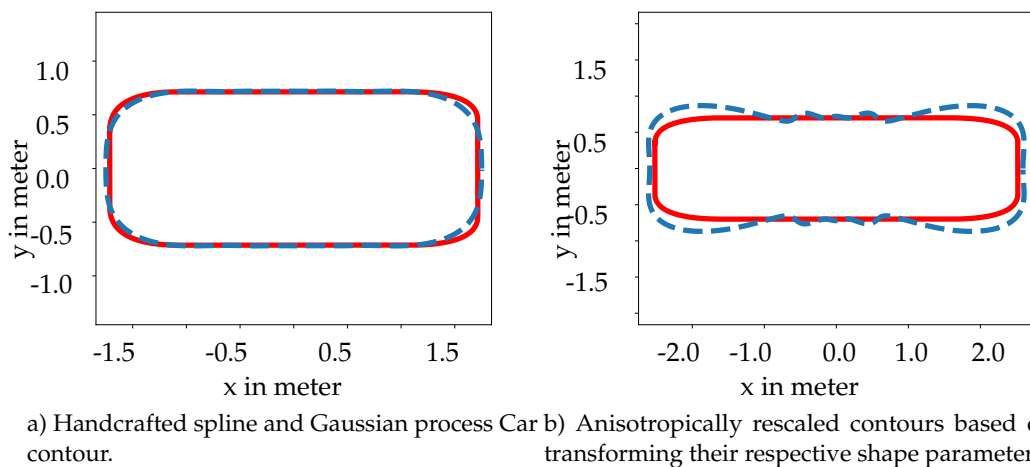


Figure 4.7: Comparison of modelling prior knowledge with spline and Gaussian process models. The spline model is shown as a solid red boundary and the Gaussian process model is shown as a solid blue dashed boundary. The left figure shows the prior model with 20 shape parameters for a vehicle boundary and the spline model’s shape guess with 16 parameters. The right figure shows the prior guess of the same models on an elongated version of the vehicle’s shape. Based on [2]©2018IEEE.

This makes the direct comparison of the shapes very complicated without allowing a full update for the radial function. The closest approximation of scaling behaviour in literature is the use of isotropic scaling, as it is trivial to implement with a scalar factor in radially defined shapes. Thus, we aim for a simple comparison of this scaling behavior.

For the experiments the priors are set to the shape models from Figure 4.7. We, however, constrain the Gaussian process to a uniformly scaling parameter for the scaling experiments. For the tracking experiment, we will keep the shapes constant to avoid unfair comparisons in the performance. The implementation of an extension of a radial function to anisotropic scaling requires a mapping from radial parameters to the Cartesian as well as the scaled shape’s angle to the non-scaled angle in every calculation, which is based on the two different spaces of the contour and the state

parameters, which again would make the shape model more unintuitive and hard to handle. As of now, we are not aware of a work that handles this problem.

Shape Scaling

We first illustrate the scaling behaviour of the spline based model based on the basis functions compared to the scaling of a hand-tuned Gaussian process with isotropic scaling, illustrating the advantage of this behaviour based on a simulation of measurements.

The true boundary in this simulation is represented as a polyline of three segments that cover two sides of a car's approximated contour as well as a corner approximated as a line segment. The measurement noise covariance matrix for this experiment is set to $R = \text{diag}(0.1^2, 0.1^2)$ and we provide 20 measurements per update.

We compared the proposed spline model with eight basis points with the handcrafted state of the art Gaussian process 20 shape defining angles and according fix distances to the surface.

The initial state for the spline is set to

$$\mathbf{x}_{0,spline} = [3, 1, 0, 1.3, 1.3]^T \quad (4.70)$$

$$\mathbf{P}_{0,spline} = \text{diag}(10^2, 10^2, 0.1^2, 1, 1), \quad (4.71)$$

and

$$\mathbf{x}_{0,gp} = [3, 1, 0, 1.3]^T \quad (4.72)$$

$$\mathbf{P}_{0,gp} = \text{diag}(10^2, 10^2, 0.1^2, 1), \quad (4.73)$$

for the Gaussian process model, while no process model is assumed to be present in this static scenario.

The results from 10 as well as 100 updates in this experiment are presented in Figure 4.8.

While both models fit the shape with respect to their limitations, the advantage of the anisotropic scaling is clearly visible as the shape can only be represented by altering length and width independently. Therefore, the Cartesian defined spline model can alter length and width to fit the shape after 10 updates and sticks to this boundary after 190 further updates. On the other hand, the isotropic Gaussian process finds an estimate with well-estimated length but slightly wrong rotation and underestimated width, which grows towards the correct estimate of the width after 100 iterations while overestimating the length as a trade-off for a correctly estimated orientation and width.

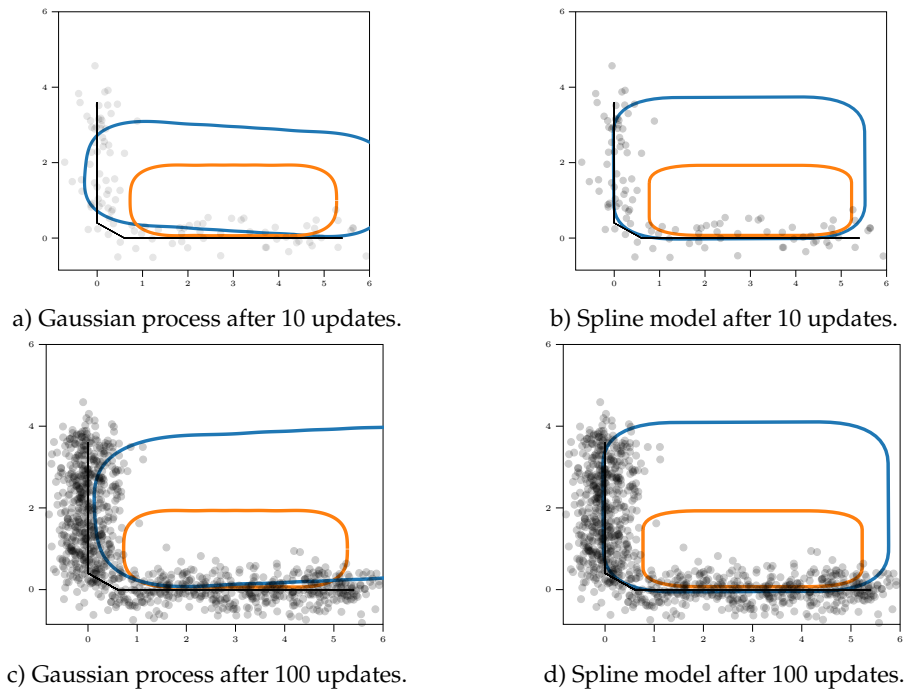


Figure 4.8: Scaling differences in hand tuned Gaussian process with constrained shape but isotropic scaling and the spline model with anisotropic scaling.

Tracking

We evaluate the tracking performance of this extended target model with respect to the state of the art Gaussian process model. For a fair comparison, we restrict the models to constant shapes and focus on the similarity of their performance. This is done because the free shape updates make the comparison of the likelihoods as well as the difference in state parameters incompatible. Here a similarity in the likelihood is a strong indicator that these models are interchangeable in typical multi extended target tracking approaches. The data used for this experiment is provided by a ScaLa 1 mounted in bumper height as described in 4.5.1.

For both target models we use the coordinated turn motion model described in Chapter 2 Section 2.5. With constant shapes the state in both models can be fully described by the kinematic parameters of the coordinated turn motion model with the center in meters, orientation in radiant, radial velocity in meter per second and finally the yaw rate in radiant per second, such that

$$\mathbf{x}_{kin} = [\mathbf{m}, \phi, v, \dot{\phi}]^T . \quad (4.74)$$

We set the process noise terms to

$$\sigma_a = 2.25, \quad (4.75)$$

$$\sigma_w = 1.5, \quad (4.76)$$

while the measurement noise covariance matrix is set to

$$R = \text{diag}(0.02, 0.02) . \quad (4.77)$$

We extract two single targets for the evaluation, which cover the two most common scenarios that occur in vehicle tracking. First, a car driving in a straight line and second, a car in a turning manoeuvre is extracted. For both scenarios the estimated tracks of every fourth time step are shown for both algorithms. We additionally provide the comparison of every state parameter and likelihood. The results of both tracking algorithms on a straight trajectory can be found in Figure 4.9 while the result on a curved track can be found in Figure 4.12. The state prior is set to

$$\mathbf{x}_{0,l} = [41, 10, \pi, 0, 0]^T , \quad (4.78)$$

in the straight trajectory and to

$$\mathbf{x}_{0,s} = [92.0, -1.5, 0.0, 0]^T , \quad (4.79)$$

in the curved one. The prior covariance matrix in both scenarios is set to

$$P_0 = \text{diag}(10, 10, 0.01, 5, 5) . \quad (4.80)$$

In addition to the tracked shapes over time, the state parameters of both algorithms, as well as the innovation and association log-likelihood of the measurements to their source, are provided in Fig. 4.14 for the curved trajectory with the corresponding differences. The likelihood measures are chosen to evaluate the usability of the model in multi-target-tracking algorithms as it is often used, e.g. in [HSRD16].

All state parameters show significant similarity to each other, which supports the equivalence of both methods in their performance for a given shape. Please note that we do not expect the exact same behaviour of both models due to the fact that the shapes are not equivalent. Nevertheless, the state parameters are close to each other over all time steps, where the highest deviation is available for the yaw angle, which on average differs by less than 0.01 and a maximum of 0.025, in the curved trajectory, between the target models. The similarity in the likelihoods is essential. It allows using the provided spline model for every multi-target tracker that currently implements the Gaussian process as an extended target model for the tracking of vehicles [HSRD16]. The experiments showed that both likelihoods follow the same trend. The association log-likelihood

differs in a range of 10^{-4} while the innovation log-likelihood varies to a more notable degree with an average difference of 0.23 which happened in the curved trajectory, which is still exceptionally small.

4.5.2 Roof Mounted LIDAR Experiments

To evaluate the asymmetric noise heuristic, we will perform a set of experiments and a direct comparison to naïve assumptions of the measurement origin.

We start by illustrating the behaviour of falsely initialized sizes to cover the two cases of misassignment discussed in Section 4.4.

Next, we will compare the behaviour of the asymmetric noise to the naïve assumption that measurements are generated from the 2d contour, as it is the most comparable assumption available in literature [HSRD16,GRMS14]. This evaluation will be performed for a large set of instances from the nuScenes dataset using the Wasserstein distance from Section 2.8. For which we first illustrate and explain the measures and performance on a single representative instance.

Dataset The sensor data used for all experiments is a set of single target instances from the nuScenes dataset, labelled as a car. We recognize here that these instances cover a maximal period of 20 seconds. The extraction of the data belonging to a single instance is performed analogously to the extraction for the data analysis in Chapter 3. Thus, it is based on the annotation bounding box, linearly interpolated to the non-annotated frames.

The instances are further filtered as the experiments aim to provide a measure of the tracking performance of the approach. Thus, the recovery from long times of misdetection or a sparse coverage of the target are to be filtered out. Another goal of this evaluation is to examine the applicability to dynamic targets. Thus, we are primarily interested in moving targets in our evaluation. Our last goal is to avoid data starved scenarios, which can stem from various environmental factors, such as occlusions. Based on these reasons, we perform a track selection based on a set of criteria.

First, we select only instances in which the maximum time without measurements is at most 1 second, and we filter these such that the maximum distance between 2 frames is below 5 meters to ensure that no large gaps exist in the instance. From these instances we select only those which moved a minimum of 5 meters to enforce the selection of moving targets. And last, we reduce these to only include traces in which the average number of measurements per frame exceeds 5 to filter out strongly occluded targets. This selection provides us with 2719 relevant instances based on which we can evaluate the behaviour of extended target models for roof-mounted LIDAR sensors on a large scale.

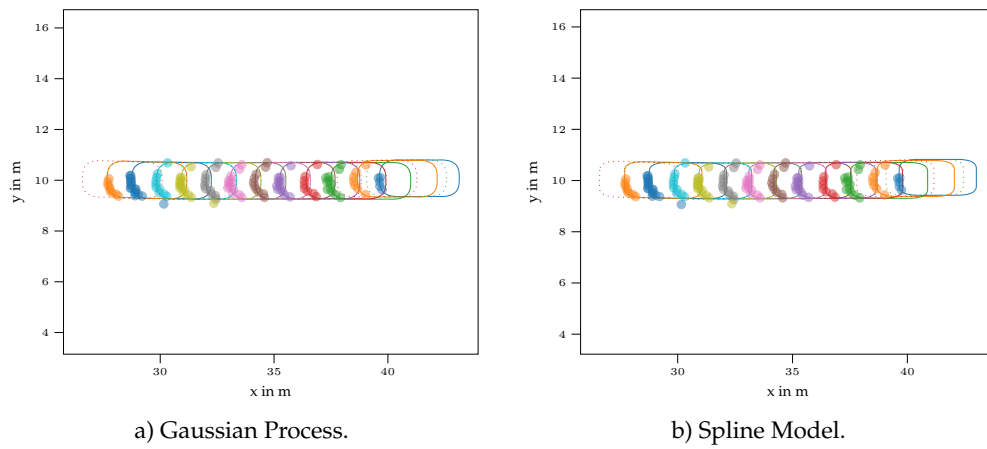


Figure 4.9: Plot of every 4th update for spline and the Gaussian process model on the straight trajectory. Predicted estimate (dashed shape), corrected estimate (full shape), and measurements (dots) of each time step are shown in the same color.

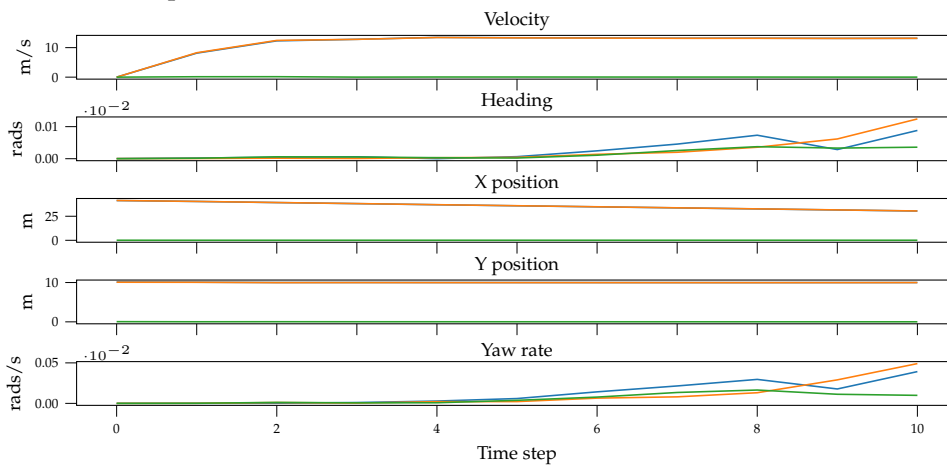


Figure 4.10: State parameters and their difference. Parameters of the spline model in blue, parameters of the Gaussian process in orange and the difference between them in green.

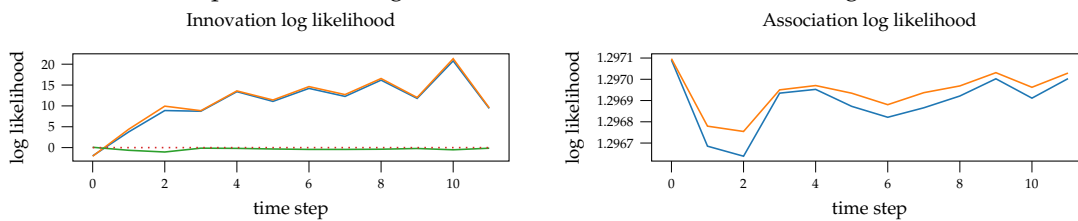


Figure 4.11: Measurements association and innovation log likelihood of the Gaussian process in orange and the spline model in blue over time. Innovation log likelihood in the left and association log likelihoods in the right Figure. For the innovation likelihood the difference is shown in green, for the association it is omitted as it is too close.

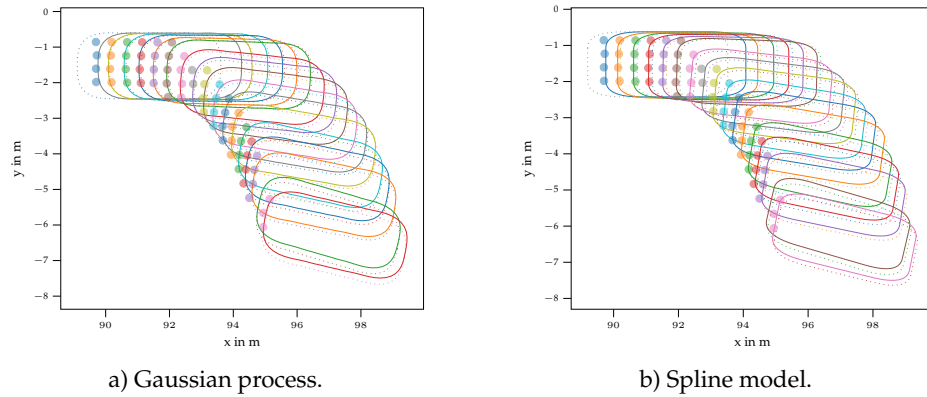


Figure 4.12: Plot of every 4th update for spline and the Gaussian process model on the curved trajectory. Predicted estimate (dashed shape), corrected estimate (full shape), and measurements (dots) of each time step are shown in the same color. Based on [2]©2018IEEE.

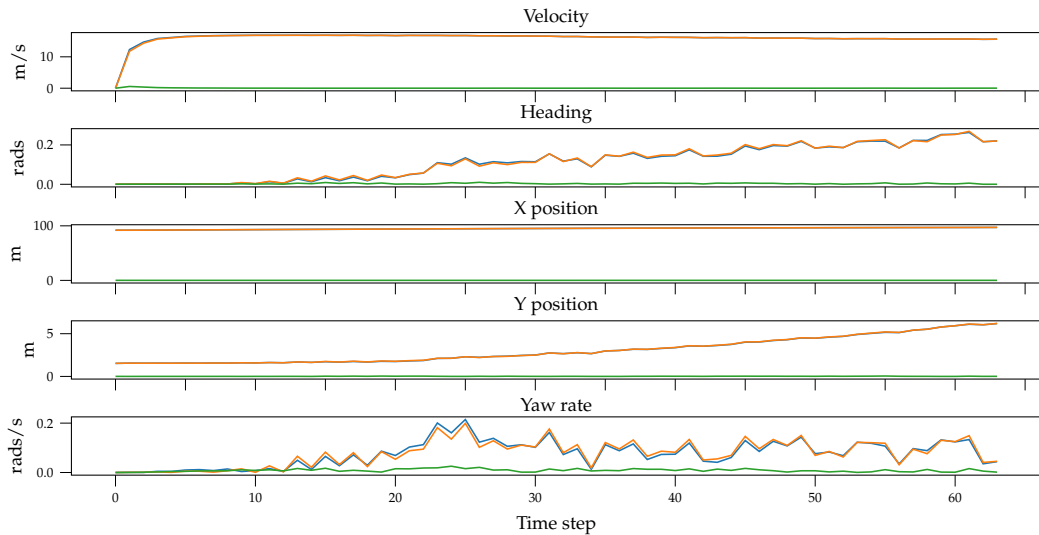


Figure 4.13: State parameters and their difference. Parameters of the spline model in blue, parameters of the Gaussian process in orange and the difference between them in green. Based on [2]©2018IEEE.

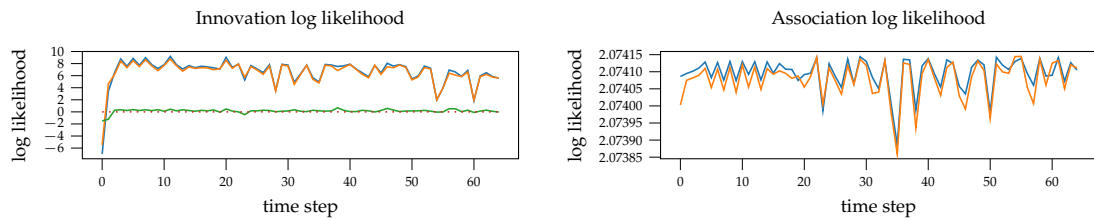


Figure 4.14: Measurements association and innovation log likelihood of the Gaussian process in orange and the spline model in blue over time. Innovation log likelihood in the left and association log likelihoods in the right Figure. For the innovation likelihood the difference is shown in green, for the association it is omitted as it is too close. Based on [2]©2018IEEE.

Model Parameters The extent model we choose for the experiments regarding the roof-mounted LIDAR is the proposed B-spline model with 8 basis points. The chosen motion model is the coordinated turn motion model described in Section 2.5.

The measurement noise parameters are chosen as

$$Q_i = \text{diag}(0.01^2, 0.01^2), \quad (4.81)$$

while the noise for the asymmetric noise heuristic is selected as

$$r_{out} = 0.01^2 \text{ and} \quad (4.82)$$

$$r_{in} = \max\left(0.3^2 \frac{\|\hat{z}_i - \mathbf{m}\|}{2}, r_{out}\right). \quad (4.83)$$

The parameters for the motion model are $\sigma_a = 1$ and $\sigma_\omega = 0.1$; an additional uncertainty of the target's slip is $\sigma_{slip} = 0.01$. To allow for changes in the shape parameters we assume the extent in length and width to be noise corrupted as well with $\sigma_{ext} = [0.01, 0.01]$.

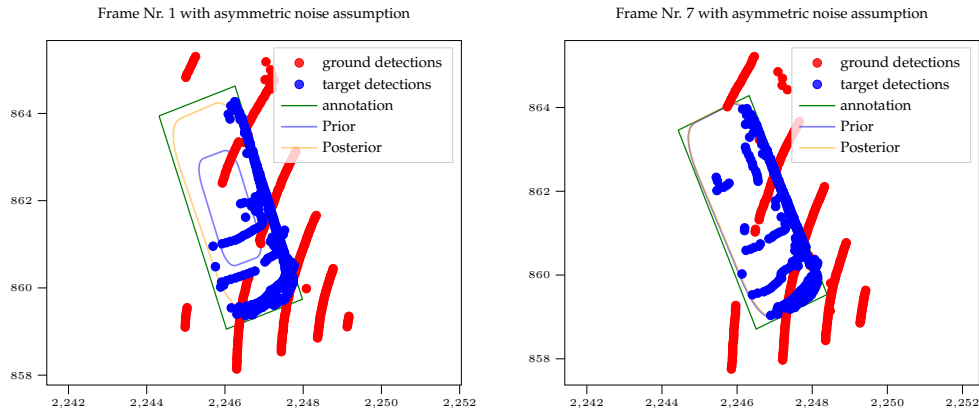
For the initialization we choose the shape estimate with respect to the annotation and the experiment. The state covariance for all experiments is chosen as $\text{diag}([2.5, 2.5, 0.5, 0.0, 0.0]) + Q_{init}$, where Q_{init} is the prediction uncertainty for a prediction driving straight for 0.01 seconds.

Misassignment of measurements

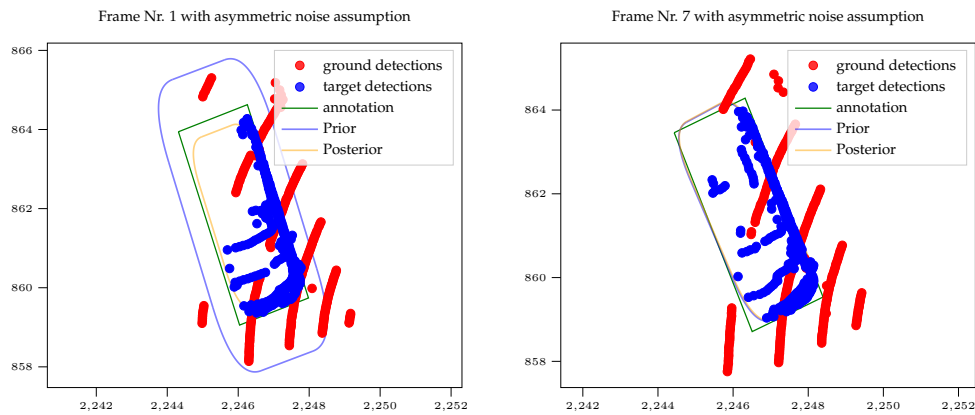
We start the experiments with an assessment of the cases in which the measurements are assigned to the wrong noise term. We create such scenarios by initializing the target with a boundary that is too large and too small, respectively, thus enforcing the cases of assigning all measurements to the interior as well as part of the interior to the surface. For this experiment, we use the instance with hash 0c06998e67cd4fbbbd89c9618806353f as an example. We initialize the shape with the annotated length and width scaled by respective factors 0.5 and 1.5. The first and seventh update for the scenarios can be found in Figure 4.15a and Figure 4.15b, respectively. In the update with the half-sized shape, an update to the boundary of the proper exterior can be observed. Both directly correct towards the target's actual boundary. The seventh update further illustrates that the boundary is kept over time.

Model Comparison

We use the extracted data from the nuScenes dataset to evaluate the asymmetric noise assumption on a large scale. The measurement noise assumption we compare it to is the assumption that measurements are generated on the targets surface which only are corrupted by Gaussian white noise. This generation model is the typical assumption concerning the generation of measurements in literature [HSRD16, GLO11]. As mentioned before, the tracking performance of the asymmetric



a) Asymmetric noise behavior on an initialization with half the shapes size. Illustrated are frame 1 and 7.



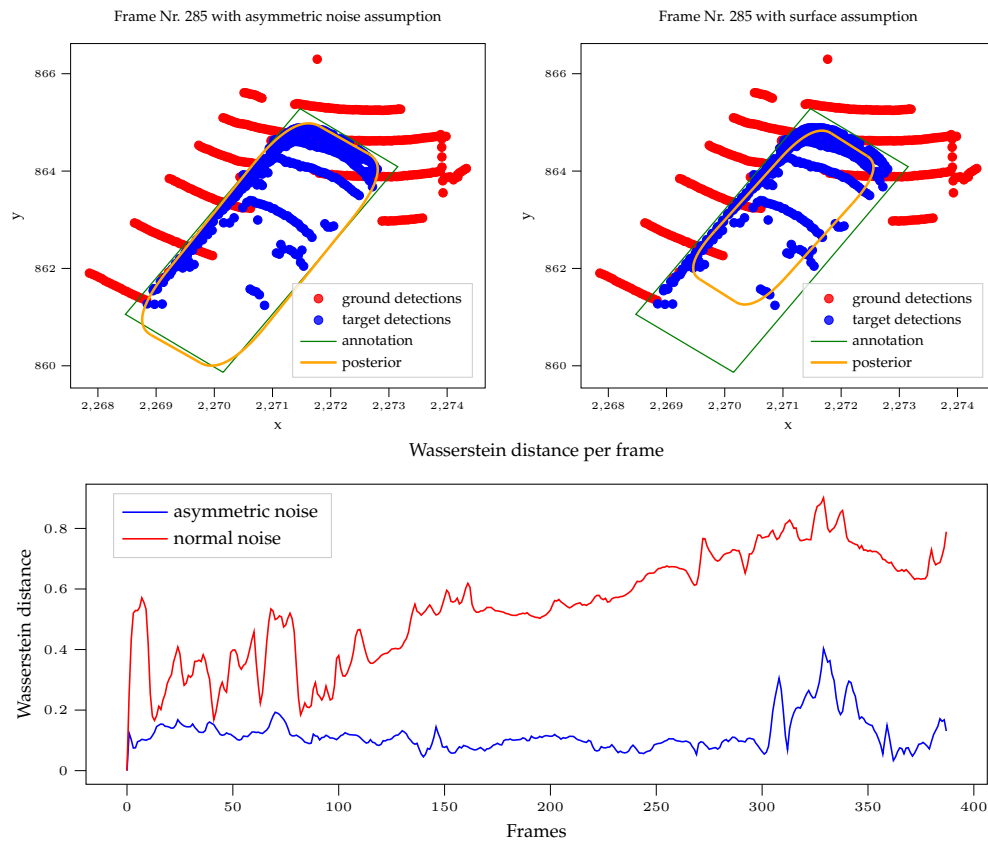
b) Asymmetric noise behavior on an initialization with 1.5 times the shapes size. Illustrated are frame 1 and 7.

Figure 4.15: Initialization effects for asymmetric noise.

noise assumption shall be evaluated and the initial state will be set to the first available annotation bounding box.

We start this evaluation with a single instance shown in Figure 4.16. In this example, the assumption of a measurement origin on the target's boundary is evaluated against the asymmetric noise heuristic. We illustrate a single frame from the trajectory as well as the Wasserstein distance over the whole trajectory.

Figure 4.16 illustrates some base observations. First, we can observe the shrinking effects implicated by the assumptions of having only measurements on the target's boundary as this assumption is only partially valid. While the boundary is extracted correctly in the front and the visible side of the vehicle an obvious discrepancy generated by the interior can be seen, which leads to an underestimation of the size of the target especially on the non visible sides. This observation is in line with the expected behaviour as suggested by the accumulated distribution in scaled



a) Example Wasserstein distance over all frames for both noise assumptions.

Figure 4.16: Example frame from the nuScenes dataset illustrating the typical estimation bias from instance `0c06998e67cd4fbbbd89c9618806353f`. The first figure provides the estimate on frame 285 with the assumption that measurements are generated from the boundary of the target. The second figure shows the same frame with the assumption that measurements are generated asymmetrically. The third figure illustrates the Wasserstein distance over the whole instance. Based on [3]©2020IEEE.

coordinates in Section 3.5.2 as well as the scaling factor distribution in Section 4.4. The asymmetric noise assumption, on the other hand, fits the measurements on the boundary of the target as they are implicitly extracted in the update step, whereas measurements on the interior are weighted less, due to the larger variance in the noise term.

The Wasserstein distance over all frames indicates that this behaviour is consistent over the entire trajectory. Both algorithms start with the same estimate and therefore provide the same Wasserstein distance in the first frame. Thereafter, the asymmetric noise outperforms the surface noise assumption on every frame as the shrinking of the shape leads to a larger distance. We note that both algorithms have the highest Wasserstein distance in the segment starting from frame 300, which in the trajectory is a section with a curve segment in which the target is moving very

slowly. From frame 340 onward, the target accelerates again and moves straight. In this section, the Wasserstein distance is reduced again in both algorithms.

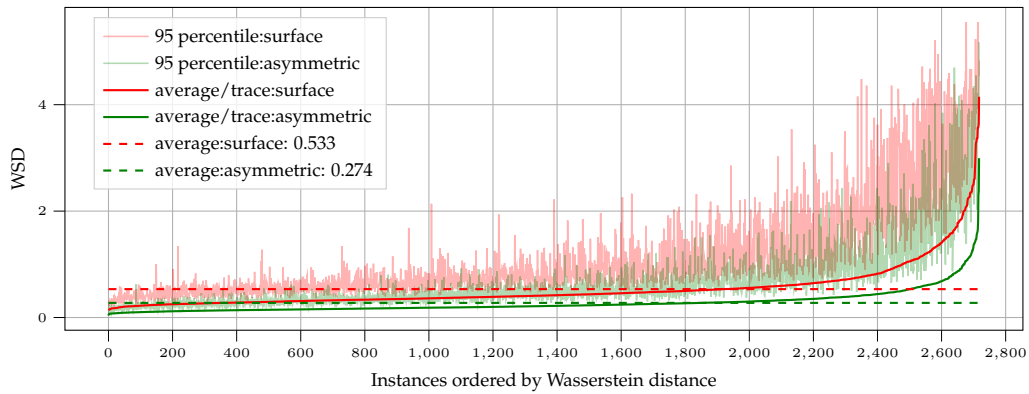
After the exemplary analysis on a single instance, we perform the same evaluation with the selection of nuScenes traces and use statistics on the Wasserstein metrics to evaluate the performance on each instance. For each instance, the average, as well as the 95th percentile of the Wasserstein distance over each time step of the trajectory, is calculated over all time steps with data. These statistics, as well as the mean of the averaged Wasserstein distance per trace, is provided for all traces. The results of this evaluation are shown in Figure 4.17 in which three plots over the data are presented that provide insights into the relative performance of the algorithms. For the reader's intuition of the metric we note that the score the score essentially describes the average distance from the eight discrete points to their pair on the ground truth, a score of 0.2 therefore implies 20 cm offset.

In Figure 4.17a the Wasserstein distances average and 95th percentile are shown. Here they are ordered by the average per instance, which is done for both algorithms individually. We further added the total average in the Wasserstein distance over all traces. In this Figure, the trend of a lower estimation error is clearly visible in the average and 95th percentile. We can further see that the overall average reduces from 0.533 for the surface noise assumption to 0.274 for the asymmetric noise assumption.

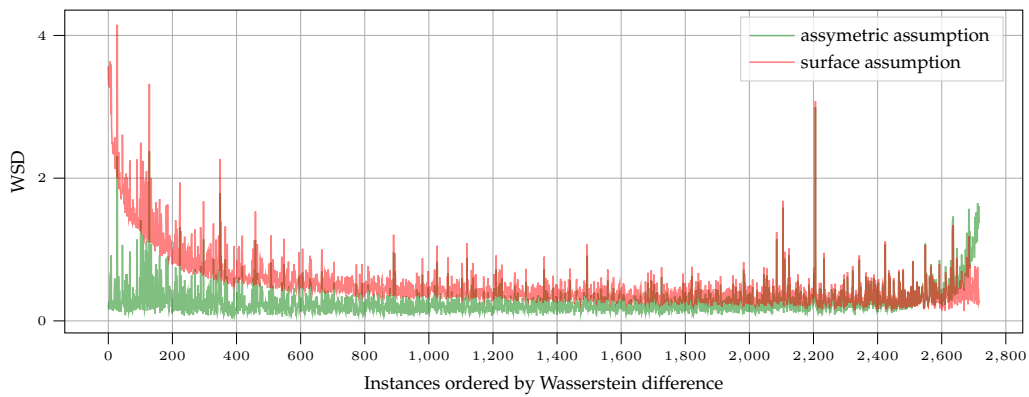
In Figure 4.17b the average Wasserstein distance is shown ordered by the total difference between the algorithms. Hence, the average Wasserstein distance for the i th instances is WSD_s^i for the surface noise assumption and WSD_a^i asymmetric noise assumption. Thus, the order is determined by score

$$WSD_d^i = WSD_s^i - WSD_a^i . \quad (4.84)$$

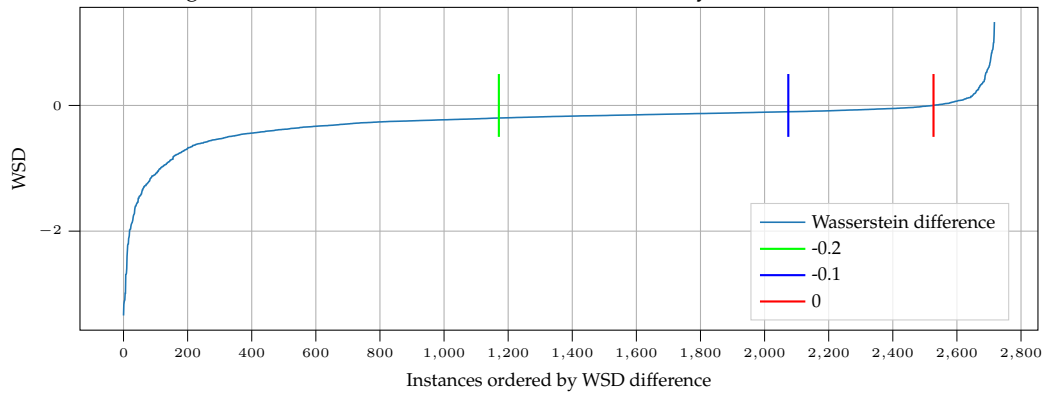
The third Figure 4.17c shows the total difference WSD_d . In it the direct comparison of the gain or loss in average Wasserstein per trace can be made. We further marked specific sections of the gain with vertical lines. A gain greater than 0.2 can be found in 0.43 % of the instances, while a gain greater than 0.1 can be found in 0.76 % of the instances and a total gain greater than 0 can be found in 92.9 % of the instances. The surface noise assumption exceeds the performance of the asymmetric noise assumption in the rest of the instances, these are instances in which the surface noise can actually lead to a stabilization. They were inspected and are instances with high dynamics and only a few stabilizing measurements on a single side which under the asymmetric noise assumption only lead to slow convergence to the surface, due to the interpretation as an interior, which in combination with strong curves can lead to misassociations of the measurements to a wrong side which in turn leads to wrong state as the rotation cannot be estimated correctly. However, this behaviour only occurs if the rotation indicating measurements are sparse, and the misestimation can also occur in the surface noise assumption in similar scenarios.



a) Wasserstein distance results ordered by the Wasserstein distance. Averages and 95th percentiles shown are calculated over all frames of a single instance and the total statistics in dotted lines are calculated over all frames.



b) Average Wasserstein distance (WSD) results ordered by the the total difference.



c) Difference in the WSD between both algorithms.

Figure 4.17: Results over the entire batch of traces, the first plot provides the average and the 95 percentile of the WSD per trace sorted by the average. The second plot shows the average WSD for each trace sorted by the difference between the asymmetric noise assumption to the surface assumption in the average WSD allowing for a trace wise comparison. The third figure provides the difference per trace in the average WSD. Based on [3]©2020IEEE.

4.6 Discussion Of The Spline Model And The Asymmetric Noise Assumption For Cars In LIDAR Data

In this section, we will discuss the advantages and disadvantages of the spline model for cars and the asymmetric noise assumption for the complex measurement distribution, both with respect to the stated model requirements.

First, we note that modelling the target's shape with splines is done with a straightforward approach, and the desired shape features in scaled coordinates are directly modelled, and the scaling of this shape is simply modelled by a scaling of the splines basis points. We showed a similar, tediously handcrafted Gaussian process with the same shape features but pointed to the problems regarding targeted scaling behaviour. We compared the isotropic scaling of a Gaussian as the closest scaling behaviour that may be introduced. Here we note that a fairer comparison is currently unavailable for radially defined shapes that still model the desired contour.

Regarding the tracking with fixed shapes, we illustrated that the spline model performs similar to the Gaussian process in the estimation of the targets parameters and the likelihood, which directly allows for the use of the spline model instead of the Gaussian process in multi-target tracking frameworks. We note the multi-target tracking aspect was subsequently evaluated in [BBWR19], this work compared the performance of the spline model in a PHD filter for extended targets to a rectangular shape model from [BRAD16], based on a simulated environment. It showed the better performance of the B-spline based model and its stability against a different number of measurements, even though the simulation produced measurements based on rectangular shapes.

We further analysed the distribution of measurements from a roof-mounted LIDAR for radial RHMs and introduced a lightweight heuristic to model the observed behaviour with the asymmetric noise assumption and compared it to the assumption of surface noise as assumed in other state of the art car models such as [HSRD16, GRMS14, BRAD16].

Thus, the advantages of the spline model with the asymmetric noise are summarized as

- modelling shape features is a simple process and is produced with a minimal amount of basis points,
- the inclusion of anisotropic scaling can be performed directly on the splines basis points,
- the model may directly be used in a multi-target-tracking framework for cars,
- the surface distribution may be included via the asymmetric noise assumption.

Thus, we provided a model that fulfilled all of our car's extent model requirements stated based on the data analysis in Chapter 3.

However, we are aware that the model, as it was developed data-driven, may not simply be altered to serve other targets or scenarios. A relaxation to other shapes as removing the prior knowledge

of the shape of the target requires the basis points to be free. This would result in a free shape model, for which the radial association is no longer valid. We noted that an approach for this is available with the extended target PMHT but a significant drawback of this approach may be summarized as,

- there is no easy relaxation of the model to general shapes, especially for volume scattering.

This drawback, however, is not important for the specific problem of modelling the extent of cars.

Radar Extent Modelling and Tracking

Contents

| | |
|--|----|
| 5.1 Problem Setting | 77 |
| 5.2 State Of The Art: Radar Extended Target Models | 79 |
| 5.3 Gaussian Mixture As An Extended Target Model | 81 |
| 5.4 Learned Gaussian Mixture Model | 82 |
| 5.5 Extent Model And PMHT Update | 83 |
| 5.6 Evaluation | 86 |
| 5.7 Discussion Of The GMM_PMHT | 95 |

Remark. *This chapter is based on the conference publications [4], [1], which use multi-target algorithms to approach the problem of single extended target tracking in the automotive area based on learned GMMs.*

RADAR sensor data provide two features that differentiate it entirely from data a LIDAR sensor generates. These features result in a different measurement generation process and, therefore, the different requirements an extent model has to fulfil, as shown in Section 3.5. First, the data provide a significantly lower number of planar measurements per time step than the large amount of 3d LIDAR measurements. Second, the measurements originate from specific measurement sources as was further discussed in Section 3.5 and are not simply distributed on the target's surface or volume. Thus, to provide a full shape estimate a suitable extent model needs to be created that accounts for specific measurement sources that emit a low amount of measurements.

5.1 Problem Setting

With respect to the complex density that is present in the RADAR data analysed in Section 3.5, we realised that specific measurement sources are present in the conditionals of the accumulated

RADAR data, thus, based on the aspect angles and the relative view on the target. These origins are, broadly speaking, the wheelhouses and the number plates, depending on the aspect angle under which the target is seen. This poses problems to be solved for RADAR data, which we will elaborate on.

The first problem is modelling these sources based on the available data, which contain the discrete origin points as well as the spatial distribution on the target that each source provides. Thus, we need a description that captures the distribution of measurement origins and their individual measurement generating process on the target. The next problem is the statement of an appropriate update incorporating the sources. Assuming that we can model these origins as discrete measurement sources on a target, we state the measurement association as well as the update as a measurement generating process with multiple related hidden variables. We can start by specifying the relation of sources and measurements with this statement, as it is independent of the realization of the sources.

Given an extended target with measurement sources $\hat{\mathbf{z}}_j(\mathbf{x})$ and measurement index $j \in [1, \dots, N]$, which are defined with respect to a target's state parameters \mathbf{x} , we can state the general measurement equation of the i th measurement \mathbf{z}_i as

$$\mathbf{z}_i = \hat{\mathbf{z}}_{(z_i)}(\mathbf{x}) + \mathbf{w} \quad (5.1)$$

$$= h(\mathbf{x}, q_i) + \mathbf{w} \quad (5.2)$$

where q_i denotes the index of the source which generated the measurement. This relation is our hidden variable as the true origin cannot be observed by the sensor and \mathbf{w} is, again, the zero mean Gaussian white noise. Here, we state the source $\hat{\mathbf{z}}_{q_i}(\mathbf{x})$ based on the state \mathbf{x} via the potentially non linear measurement function $h(\cdot, \cdot)$. We note that a major problem is the hidden nature of the assignment q_i , for which a resolution strategy needs to be determined.

Based on this statement and the work of Swain [Swa13] the problem can be interpreted via point process theory. In this manner, the problem can be described as a three-layer process model in which the target state \mathbf{x} poses as a parent process. It is related to a set of child processes, namely the discrete measurement sources on the target $\hat{\mathbf{z}}$. Here, each of the sources pose as their own measurement generation process that relates the source $\hat{\mathbf{z}}$ to the measurements \mathbf{z} , based on the respective noise term \mathbf{w} . This relation of measurements \mathbf{z} , measurement origins $\hat{\mathbf{z}}$ and the target state \mathbf{x} in an extended target tracking problem is shown exemplarily in Figure 5.1.

A related problem in data fusion is the multi target tracking problem where multiple point targets are to be tracked where each target acts as an independent measurement generating source. Based on the work [Swa13] we realize that the significant difference between multi and extended target tracking can be stated as the dependency of the state to the measurement sources, which for multiple point targets are the single states themselves. In extended target tracking, the sources are a fixed set of origins on the target, tied together by their relation to the parent state. Thus,

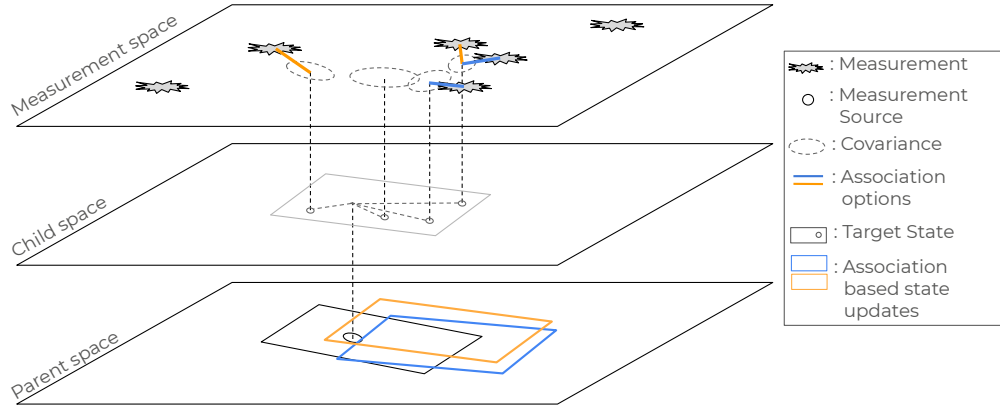


Figure 5.1: Illustration of the representative spaces and measurement generation relations for extended target tracking with discrete measurement sources.

the resolution of the association of a measurement to the single measurement origin is also fixed in relation to the parent state, allowing for the use of association and update procedures from multi-target tracking, where the update only changes the parent states parameters based on the parent child relation. For a closer comparison, we can formulate the multi-target tracking problem, where we reuse the equivalent notation for comparable entities from extended target tracking, as follows.

Given a set of point targets \mathbf{x}_j with $j \in [0, \dots, N]$ we can state the measurement equation of the i th measurement \mathbf{z}_i as

$$\mathbf{z}_i = h(\mathbf{x}_{q_i}) + \mathbf{w} \quad (5.3)$$

where q_i denotes the index of the target which generated the measurement and is our hidden variable as the true origin cannot be observed with the sensor. The relation of measurements \mathbf{z} and targets \mathbf{x}_j is shown in Figure 5.2.

Thus, we aim to generate an appropriate extent model based on discrete measurement sources while exploiting the relation between multi-target and extended target tracking, resulting in the use of multi-target procedures for the update of extended target models. We specifically aim to integrate it into the PMHT update explained in Section 2.4.3.

5.2 State Of The Art: Radar Extended Target Models

Tracking extended targets based on automotive RADAR data has been the focus of different research groups with a different focus on the properties RADAR scan points provide. We are

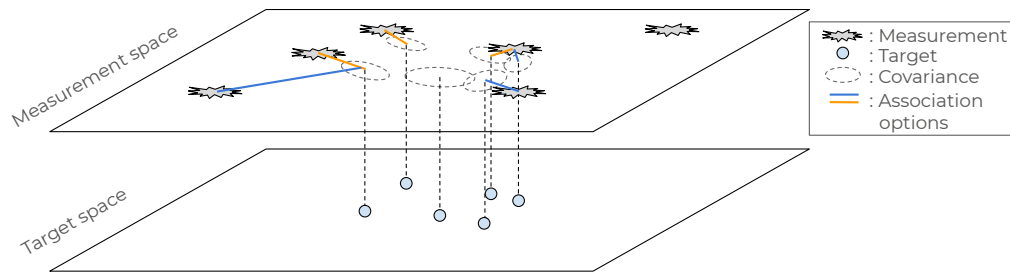


Figure 5.2: Illustration of the representative spaces and the measurement generation relations for multi target tracking.

looking for the incorporation of measurements from high-resolution RADAR systems, as the higher resolution gives rise to the extended target tracking problem.

In [GNB⁺15] the GGIW model is used to track automotive targets based on RADAR measurements, thus representing targets based on elliptical shapes. In [BRAD16] a rectangular extended target model is proposed concerning the distribution of measurements on the target. As an extension of it, the volcanormal model was developed [BDD17] which rightfully assumes that the measurements are not generated based on a commonly assumed Gaussian distribution. Instead a model is provided that alters the Mahalanobis distance in the RM approach, such that a density is generated that mostly but not uniquely generates measurements on the target's boundary instead of the target's center. In [TBH18] the Doppler measurement is included in the Gaussian process RHM for rigid vehicle shapes, such that the relation of the measured radial velocity is exploited. In [KBK⁺15] extended target tracking with high-resolution RADAR measurements has been investigated based on a detailed kinematic state. In [SD18] a particle filter solution using a learned Variational Gaussian Mixture model (VGM) as a spatial density in a unit coordinate system has been proposed. A set of publications regarding the use of truncated Gaussians for tracking automotive targets were provided with the initial approach in [XWB⁺20b] and extensions to partial visibility in [XWB⁺20c]. In [XWB⁺20a] and [XWB⁺21] learning the extent as a density based on truncated Gaussians was proposed¹.

Among the approaches using learned distributions, we especially note the work of Scheel et al. [SD18], as it provides a likelihood function that closely models the distribution on selected targets via a mixture of Gaussians. The major drawback of this method is that it is based on a particle filter and therefore suffers from the so-called curse of dimensionality [BBL⁺08], which means that the number of samples the particle filter requires scales exponentially with the dimension of the state. It would therefore be desirable to combine the learned representation of the measurement distribution with a closed-form update.

¹Please note that [XWB⁺21] compares their approach against the closed form PMHT update on learned GMMs presented in [1] which will be elaborated on in this chapter.

5.3 Gaussian Mixture As An Extended Target Model

We aim to approach the problem by using prior knowledge embedded in a learned spatial distribution on the target's volume, similar to [SD18]. We want to use a similarly trained model with stronger discretisation for the aspect angle, which will be explained in Section 5.4 while we formulate an EKF update based on the PMHT in Section 5.5. We aim to interpret a learned GMM such that it acts as a scatter source model based on the distribution of measurements the actual data offers. Our interest for this mixture is mostly to represent the spatial distribution of measurements with respect to the relative sensor position and the targets orientation, as we saw distinct modes based on simple conditioning on the aspect angle in the data analysis in Section 3.3, which is stated as the angle between the target's orientation vector and the vector from the sensor to the target's center.

We are able to describe a single Gaussian mixture model, representing the measurement distribution with respect to the aspect angle ψ from the sensor to the target's center as

$$p(\mathbf{z}|\mathbf{x}, \psi) = \sum_{i=1}^M \pi_i(\psi) \mathcal{N}(\mathbf{z}; \mu_i(\mathbf{x}, \psi), \Sigma_i(\mathbf{x}, \psi)), \quad (5.4)$$

where M is the number of components in the mixture, $\mu_i(\mathbf{x}, \psi)$ describes a single components mean, $\Sigma_i(\mathbf{x}, \psi)$ is the mixture component's covariance matrix and $\pi_i(\psi)$ is the mixture component's prior weight. In contrast to using the GMM as a likelihood function, we aim for an interpretation of the GMM as a mixed set of the most prevalent measurement origins on the target. In this interpretation an origin is described by a mixture component's mean, while the measurement noise for this component is assumed to be the spatial distribution this component provides. Thus, the measurement equation for such a GMM can be stated as

$$\mathbf{z}_i = \mu_{q_i}(\mathbf{x}, \psi) + \mathbf{w}_{q_i}, \quad (5.5)$$

assuming that measurements originate from a single Gaussian of the GMM. Here q_i denotes the index from which the measurement is emitted, $\mu_{q_i}(\mathbf{x}, \psi)$ is the components mean, and \mathbf{w} is zero-mean Gaussian white noise with $\mathbf{w}_i \sim \mathcal{N}(0, \Sigma_i(\mathbf{x}, \psi))$. Please note that the probability that this measurement was emitted by the source with index i is defined by the weight π_{q_i} as it is the relative weight of the component in the mixture. This representation is close to the definition of the general spatial distribution model in [GS05], and it essentially provides the generation process in the child space by assuming that a measurement was generated from the mixture itself.

Using the notation from the PMHT in Section 2.4.3, we can state the required quantities in the child and measurement space for the application of the PMHT for extended target tracking. In the formulation, $\mathcal{A} = \{q_i\}$ is the set of associations to the measurement sources described by the component means $\{\mu_i(\mathbf{x}, \psi)\}_i^M$, which have individual noise terms described by the respective

covariances $\{\Sigma_i(\mathbf{x}, \psi)\}_i^M$. The state source relation is the final missing part and depends entirely on the relation between the parent process and the child process. For this relation, we will use a modified version of the approach from [SD18].

5.4 Learned Gaussian Mixture Model

The RADAR model we aim to learn is based on the nuScenes data set [CBL⁺20] accumulation from Section 3.5, which specifically provides us with 2d, bounding box filtered measurements in scaled coordinates, which is a common, scale-independent coordinate space. We specifically only use a subset of the data from the nuScenes teaser data set [CBL⁺19] for the training to avoid overfitting to the whole data set.

For the individual GMM, we use discretised aspect angle ranges to cover the visible sources with respect to the relative position of the sensor and target which were shown in the data analysis in Section 3.5. These ranges are chosen to reduce the dependency on the prior pose as we only use a single state estimate, which makes the estimation more robust against errors in the orientation. This differentiates our GMM from the model trained in [SD18], which uses a continuous representation over the aspect angle. This leads to more precise likelihoods with respect to the angle, which is desirable if multiple particles are to be differentiated and when the target itself does not vary significantly, which for a single learned target is the case.

Thus, we learn a single mixture for each of eight equally large bins in scaled coordinates. We choose the data for the training of each mixture depending on the respective aspect angle range, which allows us to implicitly include a more considerable variance in aspect angles in a single GMM. We specifically choose eight ranges to cover the front, side and back views as well as views on the corners of the vehicle. These are precisely the ranges used in the data analysis from Section 3.5. For the training, we use the variational mixture training from sklearn² and the theory of the variational training can be found in [Att99]. We further use a fixed number of 20 mixture components per conditional. A set of eight GMMs is learned in the unit coordinate system, in which the data resides. They are exemplarily shown in Figure 5.3, in it the heatmap is shown with a resolution of 30×30 bins. We show a trained model on the accumulated points as well as the points conditioned on viewpoints behind the target, on the left side as well as the front right of the target.

The learned GMM now represents the measurement origins in scaled coordinates and the components act as the measurement sources in the generating child process. We note that the model in scaled coordinates is now constant and can be describe as

$$p(\mathbf{z}|\psi) = \sum_{i=1}^M \pi_i(\psi) \mathcal{N}(\mathbf{z}; \mu_i(\psi), \Sigma_i(\psi)) , \quad (5.6)$$

²(sklearn.mixture.BayesianGaussianMixture) [BLB⁺13]

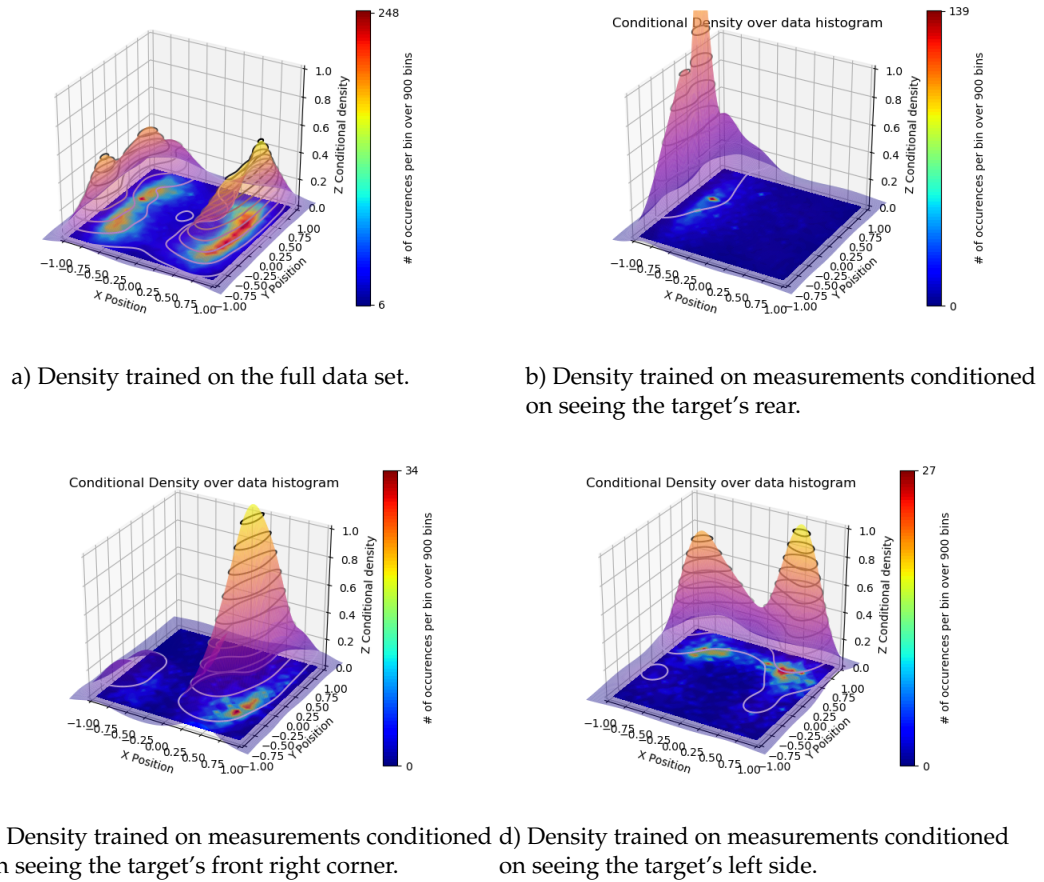


Figure 5.3: Example GMM densities over the histogram of the used training data consisting of 30x30 bins.

which removes the state dependency from the learned mixture. These sources now need to be related to the measurements in global coordinates and target's state as the parent process. For this relation, a measurement equation can be stated to allow for the formulation of an update with an appropriate filter.

5.5 Extent Model And PMHT Update

The learned density provides us with a measurement generating model in scaled coordinates. This relation is relatively uncommon for typical measurement scenarios as we do not use the measurement model in global coordinates, where measurements are generated from sources in the space the sensor can cover. We instead replace the measurement generation process with a learned GMM that acts as an alternative generating model. With this knowledge, we may now

state the parent-child relationship, as the learned mixture model representing the child process is now available.

To state the measurement equation for the learned distribution from the previous section, we denote the state parameters as

$$\begin{aligned}\mathbf{x}_{\text{kin}} &= [\mathbf{m}, \phi]^T, \\ \mathbf{x}_{\text{shape}} &= [l, w]^T,\end{aligned}\tag{5.7}$$

with \mathbf{m} being the position of the bounding box center in x - and y -coordinates, ϕ the target's yaw, l the targets half length and w the targets half width. Similar to the definition in the LIDAR model, higher-order kinematic terms can be introduced to the motion model. However, the stated parameters fully describe the parent state in global coordinates. They essentially provide an extent model that acts as an oriented, scaled bounding box relating new measurements to the sources in the learned mixture density in scaled coordinates. Now the relation to the measurement sources needs to be stated based on these state parameters.

In the first step for this relation, we need to determine the appropriate aspect angle ψ or use multiple hypotheses. For this thesis, we choose the fixed aspect angle ψ based on the prior state estimate \mathbf{x} and the sensor coordinates, which, in turn, determines the appropriate conditional. Under this assumption we will omit the dependency of $\pi_i(\psi)$, $\mu_i(\psi)$ and $\Sigma_i(\psi)$ on ψ for the rest of the section and refer to them as π_i , μ_i and Σ_i to avoid cluttered notation.

It is convenient to state the relationship between the learned mixture and new measurements in local target coordinates, i.e., the coordinate system that resides in \mathbf{m} with the orientation ϕ . The relation of the measurement in scaled coordinates $\bar{\mathbf{z}}_i(\mathbf{x})$ and a new measurement \mathbf{z}_i in global coordinates can be stated as

$$\bar{\mathbf{z}}_i(\mathbf{x}) = \mathbf{S}^{-1} \cdot \left(\mathbf{R}_\phi^{-1} \cdot (\mathbf{z}_i - \mathbf{m}) \right),\tag{5.8}$$

with the rotation matrix \mathbf{R}_ϕ and scaling matrix \mathbf{S} , defined via

$$\mathbf{R}_\phi = \begin{pmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{pmatrix}, \quad \mathbf{S} = \begin{pmatrix} l & 0 \\ 0 & w \end{pmatrix}.\tag{5.9}$$

We can then state the measurement equation for a single measurement \mathbf{z}_i originating from a specific mixture component of the GMM, denoted by q_i , namely

$$\mathbf{z}_i = \mathbf{R}_\phi \mathbf{S} (\mu_{q_i} + \mathbf{w}_{q_i}) + \mathbf{m},\tag{5.10}$$

where μ_{q_i} is the mixture component's mean while \mathbf{w}_{q_i} is a random sample generated from a zero-mean Gaussian with the mixture component's covariance matrix Σ_{q_i} .

The transformation in Equation 5.10 generally introduces a state dependency to the measurement noise through rotation and scaling, which we avoid by applying an implicit measurement model with

$$\begin{aligned}\mathbf{0}_2 &= \mu_{q_i} - \bar{\mathbf{z}}_i(\mathbf{x}) + \mathbf{w}_{q_i} \\ &= \tilde{h}_i(\mathbf{x}, \mathbf{z}_i, \mu_{q_i}) + \mathbf{w}_{q_i} .\end{aligned}\quad (5.11)$$

This new measurement model states the difference in local coordinates. Thus, with the statement of the measurement model, we now replace q_i by the index $j \in [1, \dots, M]$, as it is inherently unknown and we need to incorporate all potential associations of a measurement to all sources. Now, it is possible to associate each measurement with each GM component in the mixture, such that we can state the measurement equation for the i th measurement and the j th component of the mixture as

$$\mathbf{0}_2 = \tilde{h}_i(\mathbf{x}, \mathbf{z}_i, \mu_j) + \mathbf{w}_j. \quad (5.12)$$

We now combine the reformulation of the measurement equation in Equation (5.11) with EM formalism of the PMHT, with which we already made good experience in the prior works [5, 6], to calculate the posterior weights $\omega_{i,j}^k$ from Equation (2.52), for a fixed time

$$\omega_{i,j} = \frac{\pi_j \mathcal{N}(\mathbf{0}_2; \tilde{h}_i(\mathbf{x}, \mathbf{z}_i, \mu_j), \Sigma_j)}{\sum_{t=1}^M \pi_t \mathcal{N}(\mathbf{0}_2; \tilde{h}_i(\mathbf{x}, \mathbf{z}_i, \mu_j), \Sigma_t)}. \quad (5.13)$$

The statement of the weights allows for the calculation of the pseudo measurements based on Equation (2.56), such that every source with index j has only one exact pseudo measurement which we also denote with the same index. We note that we are technically weighting $\tilde{h}_i(\mathbf{x}, \mathbf{z}_i, \mu_j)$ and not the measurements $\bar{\mathbf{z}}_i$. However, an individual pseudo measurement can directly be applied to $\bar{\mathbf{z}}_i$ instead, due to the linear nature of the measurement function in Equation (5.12) and the fact that μ_j is fixed for the calculation of the individual weights.

We, therefore, denote that the pseudo measurement in local coordinates is $\tilde{\mathbf{z}}_j$ which, analogously, can be expressed as the weighted sum of the measurements in global coordinates $\hat{\mathbf{z}}_j$ according to (5.8). In turn, we find the relation between them as

$$\hat{\mathbf{z}}_j = \mathbf{R}_\phi \mathbf{S} \tilde{\mathbf{z}}_j + \mathbf{m}. \quad (5.14)$$

The resulting measurement equation stating the relation of source and pseudo measurement is then given by

$$\mathbf{0}_2 = \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j) + \mathbf{w}_j. \quad (5.15)$$

We further denote the pseudo covariance matrix in scaled coordinates as $\tilde{\Sigma}_j$.

This statement can now be used in an appropriate filter for nonlinear problems to find the MAP estimate, since the measurement equation (5.15) for each measurement source and pseudo measurement is still nonlinear. Thus, an EKF or UKF may be used. The application of the UKF is again straightforward, but the EKF requires the derivation of the Jacobian, which will be provided in the following.

For the application of the EKF the Jacobian of the measurement equation $\tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)$ in the unscaled coordinate system is calculated as

$$\frac{\partial \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)}{\partial \mathbf{x}} = \left[\frac{\partial \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)}{\partial \mathbf{m}}, \frac{\partial \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)}{\partial \phi}, \frac{\partial \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)}{\partial l}, \frac{\partial \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)}{\partial w} \right], \quad (5.16)$$

where

$$\frac{\partial \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)}{\partial \mathbf{m}} = -\mathbf{S}^{-1} R_\phi^{-1} \mathbf{I}_2 \quad (5.17)$$

$$\frac{\partial \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)}{\partial \phi} = \mathbf{S}^{-1} \frac{\partial R_\phi^{-1}}{\partial \phi} (\tilde{\mathbf{z}}_j - \mathbf{m}), \quad (5.18)$$

$$\frac{\partial \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)}{\partial l} = - \begin{pmatrix} \frac{2}{l^2} & 0 \\ 0 & 0 \end{pmatrix} \mathbf{R}_\phi^{-1} (\tilde{\mathbf{z}}_j - \mathbf{m}), \quad (5.19)$$

$$\frac{\partial \tilde{h}_j(\mathbf{x}, \hat{\mathbf{z}}_j, \mu_j)}{\partial w} = - \begin{pmatrix} 0 & 0 \\ 0 & \frac{2}{w^2} \end{pmatrix} \mathbf{R}_\phi^{-1} (\tilde{\mathbf{z}}_j - \mathbf{m}). \quad (5.20)$$

We note that the PMHT performs a batch update over multiple time steps. This kind of update, in general, is optimized using a Kalman smoother. For all intents and purposes we would need to use a nonlinear Kalman smoother as the measurement equation is nonlinear due to the relation of parent and child space. We instead choose to constrain the batch size to one, which makes the backwards propagation in the retrodiction of the PMHT obsolete and results in a simple nonlinear Kalman update. We will refer to this model as the GMM_PMHT for the remainder of this thesis.

We illustrate the general relation in Figure 5.4, please note that we omit the transformation to local coordinates to avoid an overcomplicated structure in the figure, since the pseudo measurements can be expressed in global coordinates via a simple transformation.

5.6 Evaluation

For this evaluation, we start with a small set of example instances from the nuScenes teaser data set based on which the performance of the derived GMM_PMHT algorithm is tested and compared against the spline model for LIDAR data as well as the GGIW. After this small scale evaluation, we move on to the set of selected traces from the full nuScenes data set. On these traces we use the same evaluation against the spline model as a direct comparison of a learned model against typical

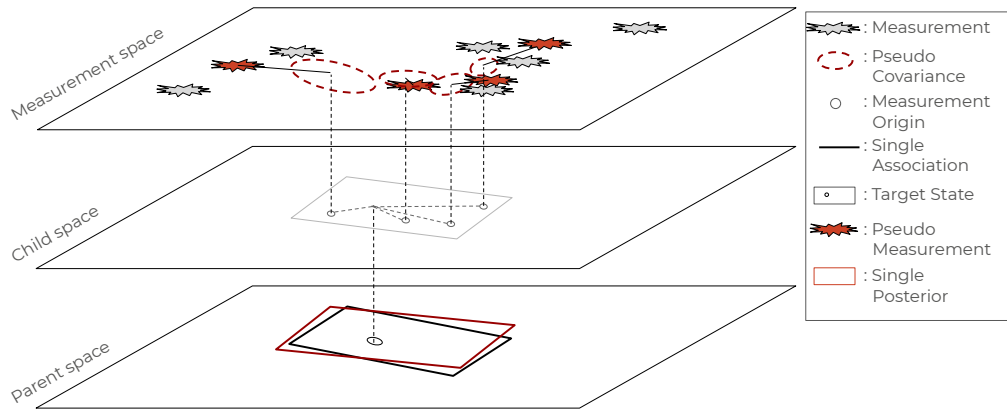


Figure 5.4: Schematic illustration of the update for the PMHT model.

extended target assumptions, of measurements generated on the targets surface, with closed-form solutions in RADAR data.

5.6.1 Single Target Evaluation

Our initial aim is the comparison of the GMM_PMHT update on real data from typical automotive use cases against commonly used extended target tracking models. We specifically compare against representatives for classical contour models with our spline model and elliptical extended target models.

Data Set We use a set of 4 scenarios to show the base performance of these algorithms in comparison. For these scenarios, the trajectory of the ego vehicle and the key-frame annotations of the target are shown in Figure 5.5.

The target in the first scenario is a slow-moving car in a parking area that follows the ego vehicle. This scenario shows substantial changes in the orientation. It provides a view on multiple sides of the target as the front, left, and right sides are visible at different times of the trajectory. The targets in the following two scenarios are similar. They both start in a curve and quickly go into a straight motion. The significant difference between the scenarios is the curvature of the targets trajectory and the relative position of sensor and target. The second scenario provides a view on the front of the target, while the third scenario provides a view of the target's rear. In the last scenario the target and ego vehicle separate, the ego vehicle takes a left turn while the target takes a right turn. This provides a view on the rear of the target, given the diverging paths. It should be noted that this target is not visible over the entire scene, which results in a lower number of annotated frames.

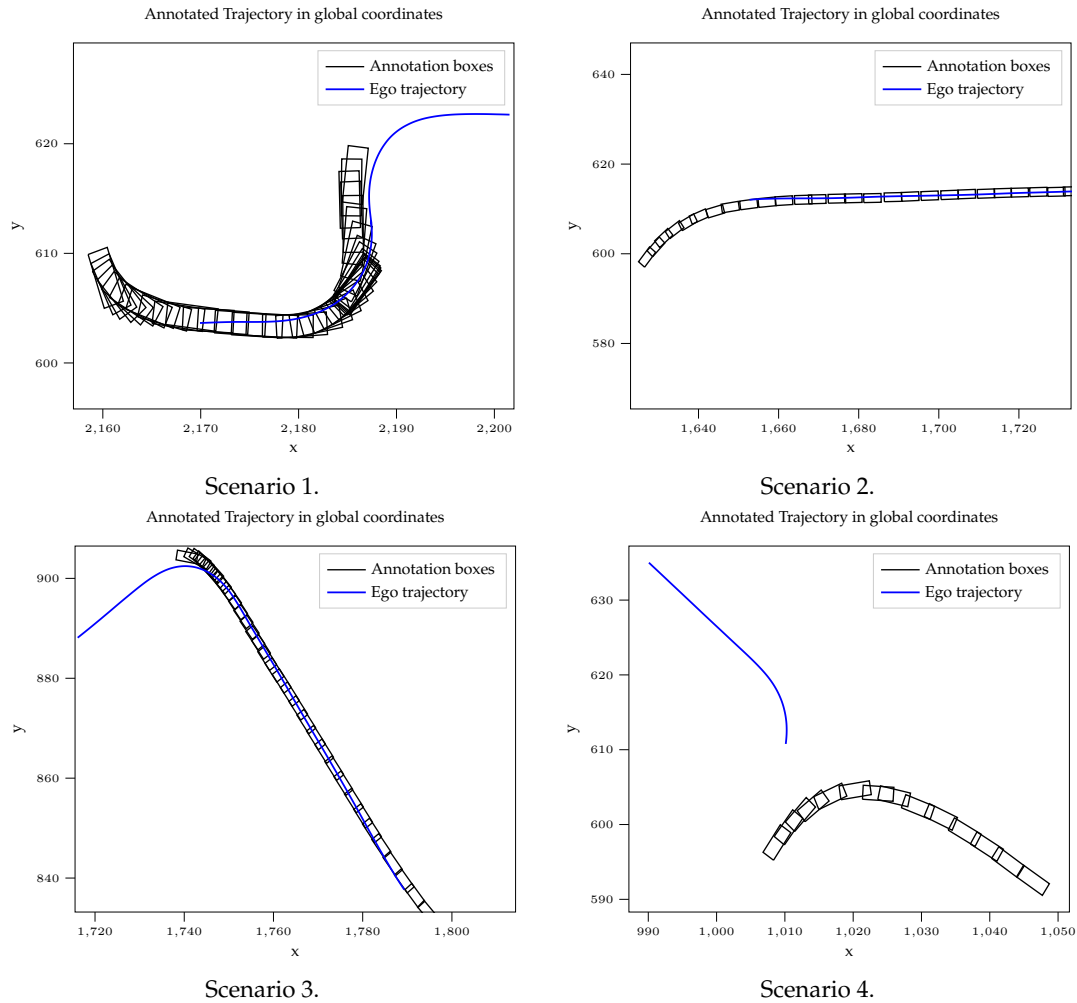


Figure 5.5: Ground truth annotation data for the four picked scenarios in global coordinates. Depicted are the annotation boxes in black, as well as the trajectory of the ego vehicle in blue. Based on [1]©2019IEEE

Models And Parameters For the evaluation of the developed approach, we use the spline model for LIDAR data under the assumption of surface noise as it provides us with a model designed for automotive targets. We emphasise that the model was specifically designed to provide a better approximation of contour features which, while visible in the accumulated RADAR data in Section 3.5 is not necessarily present in a single scan due to the sparse nature of the RADAR data in comparison to LIDAR data. As a representative of elliptical models, we use the GGIW from [GO12], which describes the extent as a distribution.

We choose the initial prior kinematic and extent parameters as the position, rotation, length and width as the corresponding parameters of the first annotation, for all algorithms in the experiments.

This annotation is chosen on the first available time step that provides measurements. The initial velocity and yaw rate were set to 0. For the last scenario, we add an additional offset of 2 meters in y to the target's position to illustrate the influence of an erroneous initialisation on the algorithms.

The motion model for the B-spline model and the PMHT model is chosen as the coordinated turn motion model from Section 2.5, such that their state may be described as

$$\mathbf{x}_{kin} = [\mathbf{m}^T, \phi, v, \dot{\phi}]^T . \quad (5.21)$$

We choose the noise parameters for these models as

$$\sigma_a^2 = 1.5^2 , \sigma_\omega^2 = 0.05^2 , \quad (5.22)$$

$$(5.23)$$

while additional noise on the extent parameters is added with

$$\sigma_\omega^2 = 0.05^2, \sigma_l^2 = 10^{-5} . \quad (5.24)$$

The difference in the noise for the shape parameters is motivated by the general size distribution shown in the cars dimensions in Section 3.5. The accumulated shape over all instances suggests that the width varies significantly less than the length. We further note that we introduce an alteration to the GMM_PMHT, shifting the center to the rear, which is modelled in scaled coordinate in this case by the vector $[0, 0.5]$.

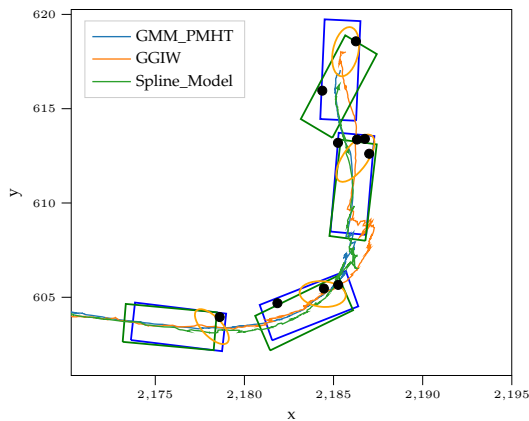
For the GGIW, which typically tracks the velocity independently of the orientation of the state, the motion model is assumed to be a vectorial defined constant velocity model in which the noise is defined as

$$\sigma_v = \text{diag}(1.5, 1.5) . \quad (5.25)$$

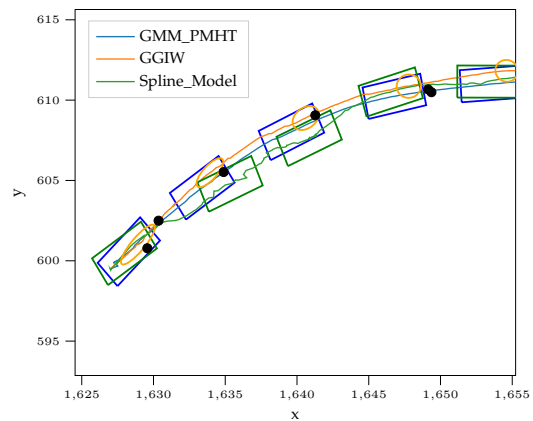
Additionally, the temporal decay of the model is set to $\tau = 3s$ while the window size is set to $w = 20$ according to the definitions in [BRG⁺15].

Model Comparison For the evaluation we use the eight point WSD distance for bounding boxes defined in Section 2.8 on the annotated frames from the nuScenes data set. The bounding box for spline and learned model can simply be stated by interpreting the model's shape parameters as a rectangular, oriented bounding box. The parameters of the GGIW are also interpreted as a bounding box, by using the orientation, length and width of the represented ellipse as parameters.

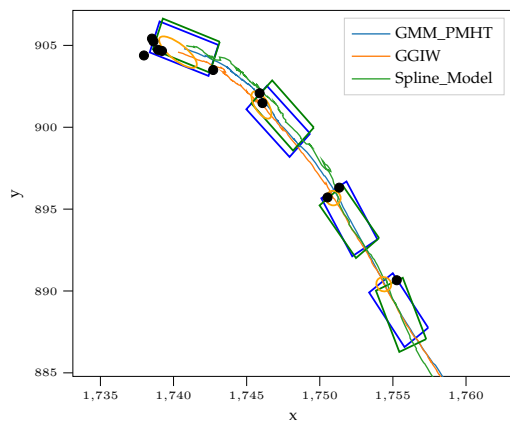
We show a set of frames from each scenario in Figure 5.6, while the resulting Wasserstein distances on each keyframe are shown in Figure 5.7.



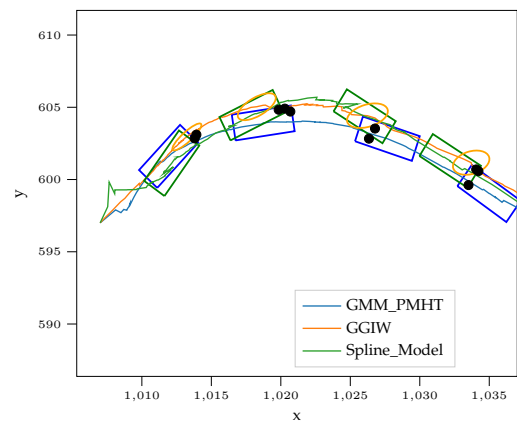
a) Subset of resulting estimates for scenario 1.



b) Subset of resulting estimates for scenario 2.



c) Subset of resulting estimates for scenario 3.



d) Subset of resulting estimates for scenario 4.

Figure 5.6: Updated estimates from all algorithms are shown on sample excerpts from the different scenarios. For each algorithm, the center of the estimate is provided as a coloured line where blue indicates the GMM_PMHT, orange indicates the GGIW and green indicates the spline's bounding box. For the selected updates, the corresponding measurements are shown as black dots, while the bounding boxes of each model are shown in their corresponding colour, and the GGIW estimate is indicated as an ellipse. Based on [1]©2019IEEE

Scenario 1 shows that all trackers follow the target's trajectory to a certain degree. The GGIW fits the measurements very well, which due to their sparse nature, only cover a smaller section of the full extent of the target. This behaviour, however, results in a bad estimate with regards to the full shape which is visible in the Wasserstein distance. The spline model follows the measurements well but attributes measurements that are slightly on the inside of the target directly to the surface, which results in slight misplacements of the shape in general. A substantial deviation in the estimation of the orientation is visible in the final curve as the greedy radial association cannot stabilise the orientation with the sparse measurements available. The learned model, however, follows the

measurements closely, even in solid manoeuvres. Every model's behaviour results in a respective Wasserstein distance. The GGIW estimates the shape smaller than expected, which increases the Wasserstein distance significantly. The spline model does not change the shape significantly. However, the precise attribution of measurements to the surface increases the Wasserstein distance, especially in the last curve. The GMM_PMHT provides the lowest Wasserstein distance as the attribution of the measurements to the learned sources, which represent the typical car's spatial distribution and seems to be more stable.

Scenario 2 provides a similar behaviour for the GGIW as it fits the visible front of the target. The spline model, on the other hand, shows a divergent behaviour in the beginning as measurements are attributed to the wrong side of the target. An appropriate association is found after a set of iterations, after which the target is tracked accurately again. The learned model, again, seems to find a reasonable estimate of the target. The model-specific behaviour is shown in the Wasserstein distance. The specific misassignment of the measurements is visible in the first 7 keyframes. The Wasserstein distance in the GMM_PMHT is the lowest as the model fits the measurements well.

Scenario 3 shows similar results as scenario 2, as the trajectories are alike, except for the visibility of the respective sides. The GGIW fits the rear instead of the front. The major difference to scenario 2 can be seen in the GGIW, where a strong reduction of the Wasserstein distance is found after keyframe 30, as more measurements indicating the true extent are visible. The spline model recovers quicker from the wrong associations and is closer to the learned model's performance in contrast to scenario 2, while the learned model again performs with the lowest overall Wasserstein distance.

Scenario 4 powerfully illustrates the effects of the erroneous initialisation. All models suffer significantly from the displacement. We note that the estimated position of the GGIW fits firmly on the visible measurements but is slightly skewed due to the initial shift. On the other hand, the spline model is strongly affected as the greedy association assigns the measurements to the wrong side which in turn is compensated by changes in the kinematic parameters, such that the model eventually fits the estimates front to measurements on the annotations rear. The spline model recovers in the later phase due to long term predictions but is off significantly in between estimates. The learned distribution, on the other hand, recovers quickly from the shift as the measurement sources are still limited to the target's visible side. We note that other shifts would potentially lead to less recovery potential, if the aspect angle changes such that the assumed measurement origins are also on a completely different side.

The overall lowest Wasserstein distance can be found with the learned model. The spline model suffers from the assumption of a surface origin, which in most cases results in a slight offset from the target's visible boundary but becomes significant as soon as the association goes to the wrong side of the target. The GGIW suffers from the partial coverage of the actual spatial occupancy of the target. We, however, note that the available spatial distribution is covered well over time. The

learned model accounts for this with the aspect angle conditionals, by learning the measurement sources on target's visible parts and, thus, as part of the measurement distribution on cars.

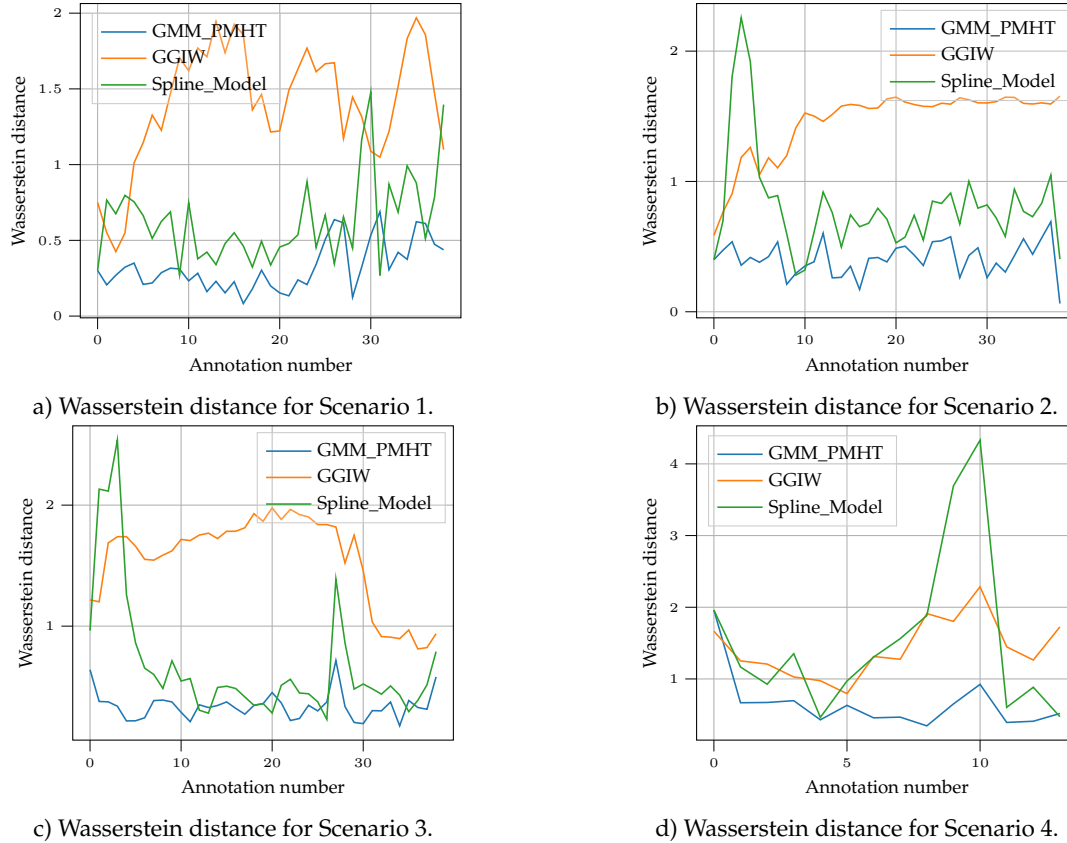


Figure 5.7: Wasserstein distances for the key-frames of all scenarios. [1]©2019IEEE

5.6.2 Large Scale Evaluation

After the experiments on single instances we move to an evaluation of the model on a large scale similar to the LIDAR evaluation. We drop the GGIW as it is rather unsuitable for the representation of the target's full extent based on the available measurements patterns. We, however, note that it performs very well in representing the available density of measurements.

Data Set For this evaluation, we consider a subset of RADAR data from the complete nuScenes data set as extracted in Section 3.5. We specifically used 650 selected instances from the full nuScenes data set. This selection is the same as in [4]. The tracks in this selection were filtered to remove pathological cases, which are tracks with at most three frames on which RADAR measurements were generated. This resulted in a total of 170.887 frames for the evaluation. We

also move away from only comparing the performance on keyframes and use all interpolated frames with RADAR measurements instead.

Models And Parameters This comparison will be made between the spline model and the GMM_PMHT. We, however, changed the initialisation to include the velocity by setting it via the spatial difference of the annotation centers over the time between the initial frames. We further altered the respective model parameters based on the large scale data set.

First, we increased the uncertainty of the extent parameters to $\sigma_{l,w} = 10^{-2}$ m/s allowing for stronger shape changes. Next, we changed the motion models noise parameters based on a grid search on the data. This change was motivated by the realisation that each extent model performed differently concerning the noise parameter setting of the motion model. As we were interested in the optimal performance of both models, we performed a grid search for the noise parameters σ_a and σ_α of the coordinated turn motion model, based on the mean Wasserstein distance over all frames of the data set, to find the optimal process noise parameters for both target models. The optimisation was conducted in the intervals $\sigma_a \in [0.1, 2]$ m s^{-2} and $\sigma_\alpha \in [0.1, 1]$ rad s^{-2} . The step size was set to 0.1 for both parameters. The result of this grid search is provided in Table 5.1.

| Algorithm | spline model | GMM_PMHT |
|---------------------------------|--------------|----------|
| σ_a [m s^{-2}] | 0.5 | 1.9 |
| σ_α [rad s^{-2}] | 0.1 | 1.0 |

Table 5.1: Optimised parameters for the noise parameters σ_α and σ_a for both target models.

Last, we investigated which number of iterations suffices for the GMM_PMHT to converge. We show the total mean, 95th percentile and median over the number of iterations for this target model in Figure 5.8. The result seems to be relatively stable after 10 iterations and the optimal number of iterations seems to be at 13 iterations. Therefore, we used 13 iterations for the large scale evaluation.

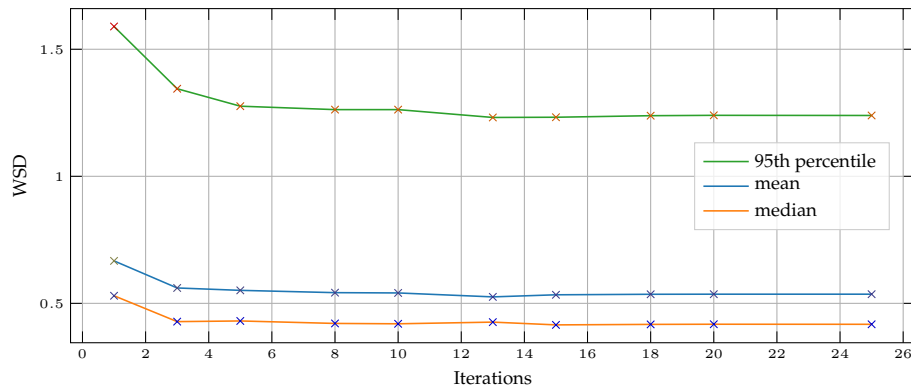
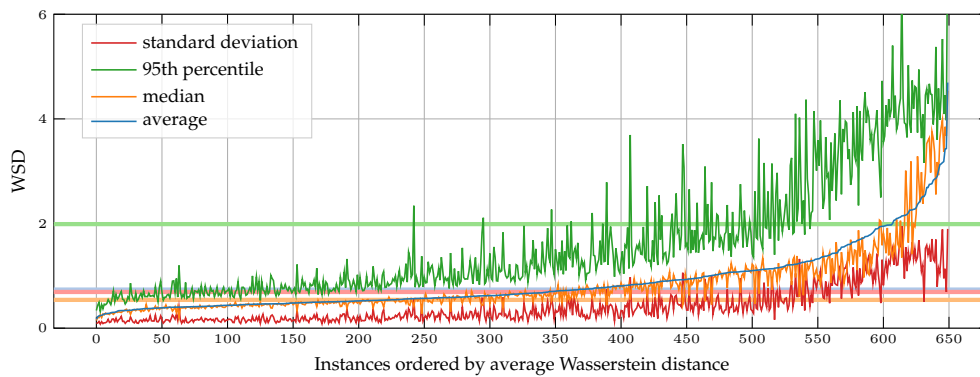


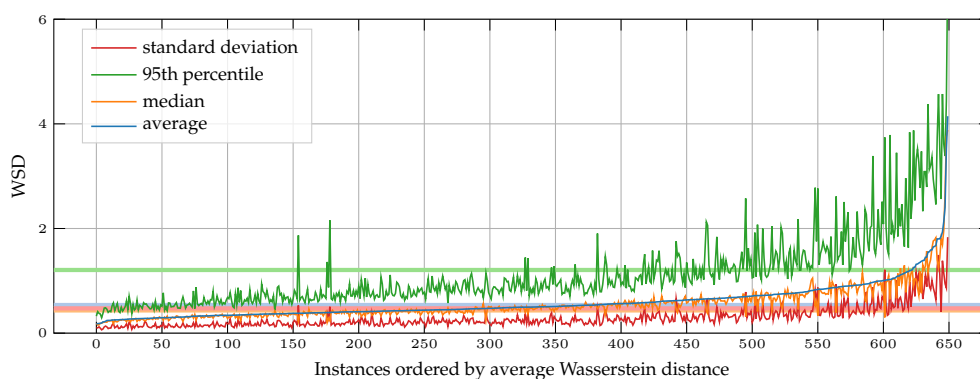
Figure 5.8: Performance evaluation for different numbers of PMHT iterations.

Model Comparison We performed a complete evaluation of the used instances for both algorithms with the tuned parameters. The resulting Wasserstein distance per trace for both algorithms is shown in Figure 5. We illustrate mean, median and 95th-percentile per instance as well as the statistics over all frames of these instances. The statistics over all instances are further provided in Table 5.2.

The statistics show an overall lower Wasserstein distance in average, median and 95th-percentile with the GMM_PMHT in contrast to the spline model, which results from the same issues illustrated in the selected single target scenarios. The learned GMM model allows for an association to sources that represent the target well. In contrast, the greedy association to the surface of the spline contour introduces erroneous updates for a small number of measurements.



a) Statistics over our spline algorithm [2].



b) Statistics over our GMM_PMHT [1].

Figure 5.9: Statistics over the compared algorithms. Shown are mean, median and 95th percentile

| Algorithm | GMM_PMHT | B-spline |
|-------------------|----------|----------|
| Average WSD | 0.538 | 0.738 |
| Median WSD | 0.434 | 0.541 |
| 95 percentile WSD | 1.207 | 1.989 |

Table 5.2: Summary of the WSD statistics over all 650 traces.

5.7 Discussion Of The GMM_PMHT

In this chapter, a closed-form MAP estimator for a learned Gaussian mixture model for RADAR data was derived by exploiting the interpretation of individual components as discrete measurement sources with defined noise covariances. We have shown that a better estimation performance can be gained against closed-form sequential updates of classical extent models.

The advantages of the model can be stated as

- the Gaussian mixture incorporates the measurement origins on the target based on real data,
- the model exploits known relations from multi-target tracking,
- the PMHT only provides a single estimate and the multimodality of the associations is avoided using the pseudo measurements which state a single measurement per source,
- the model uses a closed-form update instead of particle filtering methods.

The model does not account for the number of measurements a source emits as the GMM does not account for it and only provides relative emission probabilities by the mixture weights. The number of measurements is however often used as a Poisson point process in multi target tracking algorithms and might be of interest for further development. We further want to emphasise that we did not include the Doppler measurements in the target model as we focused on the spatial distribution. This stands in contrast to the work of Scheel et. al. [SD18], which learned the deviation in the measured Doppler velocity from the model's assumed Doppler velocity, based on detailed knowledge of targets motion model. As we accumulated the data over a large set of different targets, no knowledge of a standard motion model was available. This makes the calculation of the true Doppler measurements extremely complex, which, in turn, impedes the calculation. Assuming a standard calculation of the rotation center for a large set of different targets becomes available, the Doppler measurements could be included. Other approaches for modelling the Doppler velocity can be considered, such as [TBH18].

Thus, we can list the disadvantages as

- the measurement noise of the sensor cannot be separated from the spatial distribution on the target as they are treated as a singular noise term,
- the PMHT provides a MAP estimate and not a typically used posterior,

- the Doppler measurements are not included.

We note that the work in [XWB⁺21] compared the GMM_PMHT to their novel approach based on learned truncated Gaussian model and showed comparable performance with better and worse performance in different state parameters. We further note that our model was already extended in our work [4]. It uses cluster processes based on the work of [Swa13] to resolve the association problem. This cluster processes based approach uses multiple hypotheses that are merged, pruned and lead to a posterior update instead of the MAP update. Thus, it uses an alternative multi-target tracking approach to resolve the posed association problem³.

³Please note that this work is not included in this chapter, as the main contribution, namely the use of cluster processes, was developed by Dr. Jens Honer.

Conclusion And Future Work

We considered the problem of creating accurate extended target models with appropriate closed form inference strategies for automotive use cases in this thesis.

First, we analysed the properties of LIDAR and RADAR data, as a minor contribution in Chapter 3. Based on this analysis we determined the extended target model requirements for the respective sensor types. We found that state of the art elliptical models are already good representations for pedestrians and bicycles. We, however, were able to state requirements for the extended target models representing cars, laying the foundation for the major contributions. We first used the bounding boxes to find that an anisotropic scaling behaviour is suitable for representing a car's general extent. In LIDAR data, high-resolution measurements are generated from the target, allowing for the perception of shape details like the target's rounded corners, which are to be incorporated. We further showed that the scattering behaviour on the target changes with respect to the sensor's mounting position and should be considered in the target's model. For RADAR data, a scatter source behaviour based on the relative orientation of the target in the sensor's field of view was found on a large scale of recorded vehicles, confirming the results of works like [SD18] and [BML⁺17].

Our main contributions are summarized as follows.

To approach the problems presented in LIDAR, we developed a Cartesian defined 2d B-spline model, Chapter 4, which incorporates the desired anisotropic scaling and contour features into the extent representation. We started by stating a RHM like formulation for a free B-spline contour model with a MAP update but constrained it to suitable contour models and provided an inference strategy based on a radial associations and measurement equation. We evaluated this spline model against a well-tuned state of the art Gaussian process model showing the advantage of the scaling behaviour and a comparable performance in tracking.

We further developed a heuristic to accomplish the incorporation of measurements from LIDAR sensors mounted on the roof of the recording vessel, based on the relative position of measurements to the prior estimate's contour. This heuristic was applied to the B-spline model and has been evaluated against the naïve assumption of measurements stemming from the target's contour, showing strong improvements in the estimation of the target's extent.

We stated the general problem for RADAR based models in relation to multi-target tracking in Chapter 5 and exploited the similarities in the problem formulations to allow for the use of multi-target tracking algorithms in extended target tracking. We specifically deployed the PMHT for extended target tracking based on a learned GMM in which each component of the mixture is interpreted as a potential measurement source. This allowed for the use of the learned distribution with closed-form updates which previously only were used with particle filtering methods. We evaluated this model against other closed form solutions with contour and ellipse representations and showed a performance increase.

A set of extensions based on this thesis are possible as future work, with focus on the sensor types.

The Cartesian B-spline vehicle model was explicitly designed for tracking of cars based on LIDAR data, but there are numerous other road users. While pedestrians and bikes are well represented by elliptical target models, trucks and busses pose more complex structures that splines could model. An exemplary measurement accumulation of trucks is shown in Figure 6.1. We note that the truck is in general less rigid with the separation of the drivers cab and trailer. Integrating this dynamic as a degree of freedom in the extended target model would be desirable as future work.

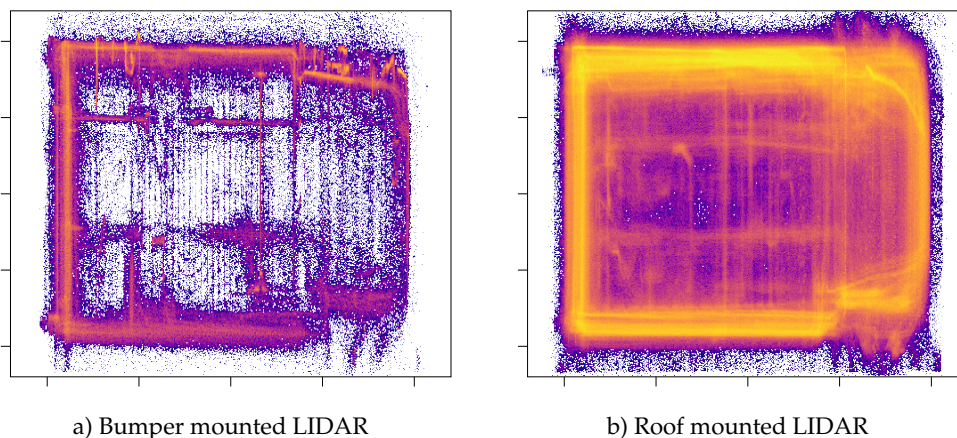


Figure 6.1: Example accumulation for trucks in logarithmic scale.

For RADAR data, we explored the application of the PMHT for a learned GMM, with varying aspect angle, for tracking cars in radar data. Here, improvements developed for the PMHT could be introduced. This, for example, includes the exchange of the association likelihood with the predicted likelihood in the expectation step as proposed in [WK07]. Another direction that can

be pursued is the application of other multi-target algorithms for the statement of an update. We have already deployed another approach in [4], in which the learned GMM was used with a cluster process interpretation in the sense of random finite sets which allowed for an update with multi-modal posterior. However, further interpretations and inference strategies can be developed. Last, we note that learning other models than a GMM can be considered.

Own Publications

- [1] H. Kaulbersch, J. Honer, and M. Baum, "EM-based Extended Target Tracking with Automotive Radar using Learned Spatial Distribution Models," in *Information Fusion (FUSION), 2019 22th International Conference on*, pp. 1–8, 2019.
- [2] H. Kaulbersch, J. Honer, and M. Baum, "A Cartesian B-Spline Vehicle Model for Extended Object Tracking," in *IEEE 21st International Conference on Information Fusion (FUSION)*, pp. 1–5, 2018.
- [3] H. Kaulbersch, J. Honer, and M. Baum, "Asymmetric noise tailoring for vehicle lidar data in extended object tracking," in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 191–196, IEEE, 2020.
- [4] J. Honer and H. Kaulbersch, "Bayesian extended target tracking with automotive radar using learned spatial distribution models," in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 316–322, IEEE, 2020.
- [5] H. Kaulbersch, M. Baum, and P. Willett, "An EM Approach for Contour Tracking based on Point Clouds," in *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 529–533, IEEE, 2016.
- [6] H. Kaulbersch, M. Baum, and P. Willett, "EM Approach for Tracking Star-Convex Extended Objects," in *IEEE 20th International Conference on Information Fusion (Fusion)*, pp. 1–7, 2017.
- [7] M. Simon, K. Amende, A. Kraus, J. Honer, T. Samann, H. Kaulbersch, S. Milz, and H. Michael Gross, "Complexer-Yolo: Real-Time 3d Object Detection and Tracking on Semantic Point Clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0–0, 2019.

Bibliography

- [Att99] Hagai Attias. Inferring Parameters and Structure of Latent Variable Models by Variational Bayes. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 21–30. Morgan Kaufmann Publishers Inc., 1999.
- [BBL⁺08] Thomas Bengtsson, Peter Bickel, Bo Li, et al. Curse-of-Dimensionality Revisited: Collapse of the Particle Filter in Very Large Scale Systems. In *Probability and statistics: Essays in honor of David A. Freedman*, pages 316–334. Institute of Mathematical Statistics, 2008.
- [BBWR19] Tim Baur, Julian Böhler, Stefan Wirtensohn, and Johannes Reuter. Tracking of Spline Modeled Extended Targets Using a Gaussian Mixture Phd Filter. In *2019 22th International Conference on Information Fusion (FUSION)*, pages 1–8. IEEE, 2019.
- [BDD17] Peter Broßeit, Bharanidhar Duraisamy, and Jürgen Dickmann. The Volcanormal Density for Radar-Based Extended Target Tracking. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE, 2017.
- [BFH12] Marcus Baum, Florian Faion, and Uwe D Hanebeck. Modeling the Target Extent with Multiplicative Noise. In *2012 15th International Conference on Information Fusion*, pages 2406–2412. IEEE, 2012.
- [BGM⁺19] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A Dataset for Semantic Scene Understanding of Lidar Sequences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9297–9307, 2019.
- [BGM⁺20] Dan Barnes, Matthew Gadd, Paul Murcutt, Paul Newman, and Ingmar Posner. The Oxford Radar Robotcar Dataset: A Radar Extension to the Oxford RobotCar Dataset. In

- 2020 *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6433–6438. IEEE, 2020.
- [BH09] Marcus Baum and Uwe D Hanebeck. Random Hypersurface Models for Extended Object Tracking. In *Signal Processing and Information Technology (ISSPIT), 2009 IEEE International Symposium on*, pages 178–183. IEEE, 2009.
- [BH11] Marcus Baum and Uwe D Hanebeck. Shape Tracking of Extended Objects and Group Targets with Star-Convex RHMs. In *IEEE Proceedings of the 14th International Conference on Information Fusion (FUSION)*, pages 1–8, 2011.
- [BH14] Marcus Baum and Uwe D Hanebeck. Extended Object Tracking with Random Hypersurface Models. *IEEE Transactions on Aerospace and Electronic Systems*, 50(1):149–159, 2014.
- [BI12] Andrew Blake and Michael Isard. *Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion*. Springer Science & Business Media, 2012.
- [BLB⁺13] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.
- [BML⁺17] Philipp Berthold, Martin Michaelis, Thorsten Luettel, Daniel Meissner, and Hans-Joachim Wuensche. Radar Reflection Characteristics of Vehicles for Contour and Feature Estimation. In *2017 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–6. IEEE, 2017.
- [BRAD16] Peter Broßeit, Matthias Rapp, Nils Appenrodt, and Jürgen Dickmann. Probabilistic Rectangular-shape Estimation for Extended Object Tracking. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 279–285. IEEE, 2016.
- [BRG⁺15] Michael Beard, Stephan Reuter, Karl Granström, Ba-Tuong Vo, Ba-Ngu Vo, and Alexander Scheel. Multiple Extended Target Tracking with Labeled Random Finite Sets. *IEEE Transactions on Signal Processing*, 64(7):1638–1653, 2015.
- [BWS⁺17] Steven Bordonaro, Peter Willett, YB Shalom, T Luginbuhl, and M Baum. Extended Object Tracking with Exploitation of Range Rate Measurements. *Journal of Advances in Information Fusion*, 2017.

- [CBL⁺19] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal Dataset for Autonomous Driving. *arXiv preprint arXiv:1903.11027*, 2019.
- [CBL⁺20] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A Multimodal Dataset for Autonomous Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11621–11631, 2020.
- [CFG⁺15] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An Information-Rich 3d Model Repository. *arXiv preprint arXiv:1512.03012*, 2015.
- [CLL17] Xiaomeng Cao, Jian Lan, and X Rong Li. Extended Object Tracking Using Control-Points-Based Extension Deformation. In *2017 20th International Conference on Information Fusion (Fusion)*, pages 1–8. IEEE, 2017.
- [CLS⁺19] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, et al. Argoverse: 3d Tracking and Forecasting with Rich Maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8748–8757, 2019.
- [DB72] Carl De Boor. On Calculating with B-splines. *Journal of Approximation theory*, 6(1):50–62, 1972.
- [DMT⁺21] Jean-Luc Déziel, Pierre Merriaux, Francis Tremblay, Dave Lessard, Dominique Plourde, Julien Stanguennec, Pierre Goulet, and Pierre Olivier. PixSet: An Opportunity for 3d Computer Vision to Go Beyond Point Clouds with a Full-Waveform Lidar Dataset. *arXiv preprint arXiv:2102.12010*, 2021.
- [EW19] Felix Ebert and Hans-Joachim Wuensche. Dynamic Object Tracking and 3d Surface Estimation using Gaussian Processes and Extended Kalman Filter. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 1122–1127. IEEE, 2019.
- [FFK10] Michael Feldmann, Dietrich Franken, and Wolfgang Koch. Tracking of Extended Objects and Group Targets using Random Matrices. *2010 IEEE Transactions on Signal Processing*, 59(4):1409–1420, 2010.
- [GBR17a] Karl Granström, Marcus Baum, and Stephan Reuter. Extended Object Tracking: Introduction, Overview and Applications. *ISIF Journal of Advances in Information Fusion*, 12(2), December 2017.
- [GBR17b] Karl Granström, Marcus Baum, and Stephan Reuter. Extended Object Tracking: Introduction, Overview and Applications. *ISIF Journal of Advances in Information Fusion*, 12(2), Dec. 2017.

- [GKM⁺20] Jakob Geyer, Yohannes Kassahun, Mentar Mahmudi, Xavier Ricou, Rupesh Durgesh, Andrew S Chung, Lorenz Hauswald, Viet Hoang Pham, Maximilian Mühlegg, Sebastian Dorn, et al. A2D2: Audi Autonomous Driving Dataset. *arXiv preprint arXiv:2004.06320*, 2020.
- [GLO11] Karl Granström, Christian Lundquist, and Umut Orguner. Tracking Rectangular and Elliptical Extended Targets using Laser Measurements. In *14th International Conference on Information Fusion*, pages 1–8. IEEE, 2011.
- [GLSU13] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision Meets Robotics: The Kitti Dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.
- [GNB⁺15] Karl Granström, Antonio Natale, Paolo Braca, Giovanni Ludeno, and Francesco Serafino. Gamma Gaussian Inverse Wishart Probability Hypothesis Density for Extended Target Tracking using X-Band Marine Radar Data. *IEEE Transactions on Geoscience and Remote Sensing*, 53(12):6617–6631, 2015.
- [GO12] Karl Granström and Umut Orguner. Estimation and Maintenance of Measurement Rates for Multiple Extended Target Tracking. In *IEEE 15th International Conference on Information Fusion*, pages 2170–2176, 2012.
- [GRMS14] Karl Granström, Stephan Reuter, Daniel Meissner, and Alexander Scheel. A multiple model PHD Approach to Tracking of Cars under an assumed Rectangular Shape. In *17th International Conference on Information Fusion (FUSION)*, pages 1–8. IEEE, 2014.
- [GS84] Clark R. Givens and Rae Michael Shortt. A Class of Wasserstein Metrics for Probability Distributions. *The Michigan Mathematical Journal*, 31(2):231–240, 1984.
- [GS05] Kevin Gilholm and David Salmond. Spatial Distribution Model for Tracking Extended Objects. *IEE Proceedings-Radar, Sonar and Navigation*, 152(5):364–371, 2005.
- [HDKD18] Stefan Haag, Bharanidhar Duraisamy, Wolfgang Koch, and Jürgen Dickmann. Radar and Lidar Target Signatures of Various Object Types and Evaluation of Extended Object Tracking Methods for Autonomous Driving Applications. In *2018 21st International Conference on Information Fusion (FUSION)*, pages 1746–1755. IEEE, 2018.
- [Hei20] Mehrnoosh Heidarpour. *EXTENDED TARGET TRACKING METHODS IN MODERN SENSOR APPLICATIONS*. PhD thesis, McMaster University, Hamilton, Ontario, Canada, 2020.
- [HSRD16] Tobias Hirscher, Alexander Scheel, Stephan Reuter, and Klaus Dietmayer. Multiple Extended Object Tracking using Gaussian Processes. In *IEEE 19th International Conference on Information Fusion (FUSION)*, pages 868–875, 2016.

- [HTS⁺12] Jürgen Hasch, Eray Topak, Raik Schnabel, Thomas Zwick, Robert Weigel, and Christian Waldschmidt. Millimeter-Wave Technology for Automotive Radar Sensors in the 77 GHz Frequency Band. *IEEE Transactions on Microwave Theory and Techniques*, 60(3):845–860, 2012.
- [HVV15] Hung Gia Hoang, Ba Tuong Vo, and Ba-Ngu Vo. A Fast Implementation of the Generalized Labeled Multi-Bernoulli Filter with Joint Prediction and Update. In *2015 18th International Conference on Information Fusion (Fusion)*, pages 999–1006. IEEE, 2015.
- [Kal60] Rudolph Emil Kalman. A new Approach to Linear Filtering and Prediction Problems. *Trans. ASME, Journal of Basic Engineering*, 82:35–45, 1960.
- [KBK⁺15] Dominik Kellner, Michael Barjenbruch, Jens Klappstein, Jürgen Dickmann, and Klaus Dietmayer. Tracking of Extended Objects with High-Resolution Doppler Radar. *IEEE Transactions on Intelligent Transportation Systems*, 17(5):1341–1353, 2015.
- [KÖ19] Murat Kumru and Emre Özkan. Three-Dimensional Extended Object Tracking and Shape Learning using Gaussian Processes. *arXiv preprint arXiv:1909.11358*, 2019.
- [Koc08] Johann Wolfgang Koch. Bayesian Approach to Extended Object and Cluster Tracking using Random Matrices. *IEEE Transactions on Aerospace and Electronic Systems*, 44(3):1042–1059, 2008.
- [Koc16] Wolfgang Koch. *Tracking and Sensor Data Fusion*. Springer, 2016.
- [KUH⁺19] R. Kesten, M. Usman, J. Houston, T. Pandya, K. Nadhamuni, A. Ferreira, M. Yuan, B. Low, A. Jain, P. Ondruska, S. Omari, S. Shah, A. Kulkarni, A. Kazakova, C. Tao, L. Platinsky, W. Jiang, and V. Shet. Lyft Level 5 Perception Dataset 2020. <https://level5.lyft.com/dataset/>, 2019.
- [LJ05] X Rong Li and Vesselin P Jilkov. Survey of Maneuvering Target Tracking. Part V. Multiple-Model Methods. *IEEE Transactions on Aerospace and Electronic Systems*, 41(4):1255–1321, 2005.
- [LOG10] Christian Lundquist, Umut Orguner, and Fredrik Gustafsson. Extended Target Tracking using Polynomials with Applications to Road-map Estimation. *IEEE Transactions on Signal Processing*, 59(1):15–26, 2010.
- [MCS⁺14] Lyudmila Mihaylova, Avishy Y Carmi, François Septier, Amadou Gning, Sze Kim Pang, and Simon Godsill. Overview of Bayesian Sequential Monte Carlo Methods for Group and Extended Object Tracking. *Digital Signal Processing*, 25:1–16, 2014.
- [MK19] Michael Meyer and Georg Kuschik. Automotive Radar Dataset for Deep Learning based 3d Object Detection. In *2019 16th European Radar Conference (EuRAD)*, pages 129–132. IEEE, 2019.

- [MPLN17] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017.
- [MWW⁺20] Sivabalan Manivasagam, Shenlong Wang, Kelvin Wong, Wenyuan Zeng, Mikita Sazanovich, Shuhan Tan, Bin Yang, Wei-Chiu Ma, and Raquel Urtasun. Lidarsim: Realistic Lidar Simulation by Leveraging the Real World. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11167–11176, 2020.
- [NBW19] Benjamin Naujoks, Patrick Burger, and Hans-Joachim Wuensche. Fast 3d Extended Target Tracking using NURBS Surfaces. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 1104–1109. IEEE, 2019.
- [NKM12] Shu Kay Ng, Thriyambakam Krishnan, and Geoffrey J McLachlan. The Em Algorithm. In *Handbook of computational statistics*, pages 139–172. Springer, 2012.
- [PMGA11] Nikolay Petrov, Lyudmila Mihaylova, Amadou Gning, and Donka Angelova. A Novel Sequential Monte Carlo Approach for Extended Object Tracking Based on Border Parameterisation. In *14th International Conference on Information Fusion*, pages 1–8. IEEE, 2011.
- [PMGC19] Abhishek Patil, Srikanth Malla, Haiming Gang, and Yi-Ting Chen. The h3d Dataset for Full-Surround 3d Multi-Object Detection and Tracking in Crowded Urban Scenes. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9552–9557. IEEE, 2019.
- [PSP⁺20] Quang-Hieu Pham, Pierre Sevestre, Ramanpreet Singh Pahwa, Huijing Zhan, Chun Ho Pang, Yuda Chen, Armin Mustafa, Vijay Chandrasekhar, and Jie Lin. A*3D Dataset: Towards Autonomous Driving in Challenging Environments. In *Proc. of The International Conference in Robotics and Automation (ICRA)*, 2020.
- [RHG14] Michael Roth, Gustaf Hendeby, and Fredrik Gustafsson. EKF/UKF Maneuvering Target Tracking using Coordinated Turn Models with Polar /Cartesian Velocity. In *Information Fusion (FUSION), 2014 17th International Conference on*, pages 1–8. IEEE, 2014.
- [SD18] Alexander Scheel and Klaus Dietmayer. Tracking Multiple Vehicles using a Variational Radar Model. *IEEE Transactions on Intelligent Transportation Systems*, 20(10):3721–3736, 2018.
- [SDPM⁺21] Marcel Sheeny, Emanuele De Pellegrin, Saptarshi Mukherjee, Alireza Ahrabian, Sen Wang, and Andrew Wallace. RADIATE: A Radar Dataset for Automotive Perception. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–7, 2021.

- [SKD⁺20] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454, 2020.
- [SL95] Roy L Streit and Tod E Luginbuhl. Probabilistic Multi-Hypothesis Tracking. Technical report, Naval Underwater Systems Center Newport RI, 1995.
- [SVV08] Dominic Schuhmacher, Ba-Tuong Vo, and Ba-Ngu Vo. A Consistent Metric for Performance Evaluation of Multi-Object Filters. *IEEE transactions on signal processing*, 56(8):3447–3457, 2008.
- [Swa13] Anthony J Swain. *Group and Extended Target Tracking with the Probability Hypothesis Density Filter*. PhD thesis, Heriot-Watt University, 2013.
- [TBH18] Kolja Thormann, Marcus Baum, and Jens Honer. Extended Target Tracking using Gaussian Processes with High-Resolution Automotive Radar. In *2018 21st International Conference on Information Fusion (FUSION)*, pages 1764–1770. IEEE, 2018.
- [TLTK19] Xu Tang, Mingyan Li, Ratnasingham Tharmarasa, and Thiagalingam Kirubarajan. Seamless Tracking of Apparent Point and Extended Targets using Gaussian process PMHT. *IEEE Transactions on Signal Processing*, 67(18):4825–4838, 2019.
- [TQM⁺20] Weikai Tan, Nannan Qin, Lingfei Ma, Ying Li, Jing Du, Guorong Cai, Ke Yang, and Jonathan Li. Toronto-3d: A Large-Scale Mobile Lidar Dataset for Semantic Segmentation of Urban Roadways. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 202–203, 2020.
- [War19] Mial E Warren. Automotive LIDAR Technology. In *2019 Symposium on VLSI Circuits*, pages C254–C255. IEEE, 2019.
- [WD14] Monika Wieneke and Sam Davey. Histogram-PMHT for Extended Targets and Target Groups in Images. *IEEE Transactions on Aerospace and Electronic Systems*, 50(3):2199–2217, 2014.
- [WK07] Monika Wieneke and Wolfgang Koch. The PMHT: Solutions to some of its Problems. In *Signal and Data Processing of Small Targets 2007*, volume 6699, page 669917. International Society for Optics and Photonics, 2007.
- [WÖ15] Niklas Wahlström and Emre Özkan. Extended Target Tracking Using Gaussian Processes. *IEEE Transactions on Signal Processing*, 63(16):4165–4178, 2015.
- [WRS02] Peter Willett, Yanhua Ruan, and R Streit. PMHT: Problems and some Solutions. *IEEE Transactions on Aerospace and Electronic Systems*, 38(3):738–754, 2002.

- [XWB⁺20a] Yuxuan Xia, Pu Wang, Karl Berntorp, Petros Boufounos, Philip Orlik, Lennart Svensson, and Karl Granström. Extended Object Tracking with Automotive Radar using Learned Structural Measurement Model. In *2020 IEEE Radar Conference (RadarConf20)*, pages 1–6. IEEE, 2020.
- [XWB⁺20b] Yuxuan Xia, Pu Wang, Karl Berntorp, Toshiaki Koike-Akino, Hassan Mansour, Milutin Pajovic, Petros Boufounos, and Philip V Orlik. Extended Object Tracking Using Hierarchical Truncation Measurement Model with Automotive Radar. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4900–4904. IEEE, 2020.
- [XWB⁺20c] Yuxuan Xia, Pu Wang, Karl Berntorp, Hassan Mansour, Petros Boufounos, and Philip V Orlik. Extended Object Tracking using Hierarchical Truncation Model with Partial-View Measurements. In *2020 IEEE 11th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pages 1–5. IEEE, 2020.
- [XWB⁺21] Yuxuan Xia, Pu Wang, Karl Berntorp, Lennart Svensson, Karl Granström, Hassan Mansour, Petros Boufounos, and Philip V Orlik. Learning-Based Extended Object Tracking Using Hierarchical Truncation Measurement Model With Automotive Radar. *IEEE Journal of Selected Topics in Signal Processing*, 15(4):1013–1029, 2021.
- [YB19] Shishan Yang and Marcus Baum. Tracking the Orientation and Axes Lengths of an Elliptical Extended Object. *IEEE Transactions on Signal Processing*, 67(18):4720–4729, 2019.
- [YBG16] Shishan Yang, Marcus Baum, and Karl Granström. Metrics for Performance Evaluation of Elliptic Extended Object Tracking Methods. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 523–528. IEEE, 2016.
- [YHH⁺19] Senthil Yogamani, Ciarán Hughes, Jonathan Horgan, Ganesh Sistu, Pdraig Varley, Derek O’Dea, Michal Uricár, Stefan Milz, Martin Simon, Karl Amende, et al. Woodscape: A Multi-task, Multi-camera Fisheye Dataset for Autonomous Driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9308–9318, 2019.
- [ZFBH16] Antonio Zea, Florian Faion, Marcus Baum, and Uwe D Hanebeck. Level-Set Random Hypersurface Models for Tracking Non-Convex Extended Objects. *IEEE Transactions on Aerospace and Electronic Systems*, 52(6):2990–3007, 2016.
- [ZFH16] Antonio Zea, Florian Faion, and Uwe D Hanebeck. Tracking Elongated Extended Objects using Splines. In *Information Fusion (FUSION), 2016 19th International Conference on*, pages 612–619. IEEE, 2016.

