Technische Universität München
TUM School of Engineering and Design

Assessing the Environment Perception Reliability of Automated Vehicles

Marco Julian Kryda

Vollständiger Abdruck der von der TUM School of Engineering and Design der Technischen Universität München zur Erlangung eines

Doktors der Ingenieurwissenschaften (Dr.-Ing.)

genehmigten Dissertation.

Vorsitz: Prof. Dr.-Ing. Johannes Betz

Prüfer*innen der Dissertation:

1. Prof. Dr. sc. techn. Daniel Straub
2. Prof. Dr.-Ing. habil. Reinhard German

Die Dissertation wurde am 20.06.2023 bei der Technischen Universität München eingereicht und durch die TUM School of Engineering and Design am 16.11.2023 angenommen.

Zusammenfassung

Das Interesse am automatisierten Fahren ist in den vergangenen Jahren rasant gestiegen. Um die Akzeptanz in der Bevölkerung für automatisierte Fahrzeuge zu erlangen, spielt die Sicherheit der Fahrzeuge eine elementare Rolle. Folglich verlangt die Zulassung solcher Fahrzeuge das Ermitteln der Zuverlässigkeit dieser. Die Sicherheit beim menschlichen Fahren wird hierbei oft als Vergleich herangezogen. Statistisch betrachtet passiert ein Unfall mit Todesfolge alle paar hundert Millionen Kilometer. Das Testen eines automatisierten Fahrzeugs einschließlich eines Testfahrers über solch lange Distanzen erweist sich allerdings als unmöglich. Aus diesem Grund sind für die Zulassung automatisierter Fahrzeuge neue Validierungsmethoden notwendig. Viele neue Validierungsmethoden fokussieren sich auf Teilfunktionen des automatisierten Fahrens, welche die Umfeldwahrnehmung, die Trajektorienplanung und die Ansteuerung der Aktuatoren umfassen.
Diese Arbeit konzentriert sich auf Validierungsansätze für die Umgebungswahrnehmung von automatisierten Fahrzeugen. Eine zuverlässige Umgebungswahrnehmung ist unerlässlich, da sie die Eingangsdaten für alle nachfolgenden Funktionen liefert. Die Arbeit gibt zunächst einen kurzen Überblick über die Validierung von automatisierten Fahrzeugen, die Wahrnehmung von automatisierten Fahrzeugen, sowie die Validierung der Wahrnehmung. Essentiell in der Umfeldwahrnehmung ist die Erkennung der anderen Objekte im Straßenverkehr. Wird ein Objekt nicht erkannt, kann dies mittelbar oder unmittelbar zu einem Unfall führen. Daher ist die Objekterkennung ein elementarer Bestandteil der Umfeldwahrnehmung automatisierter Fahrzeuge. Basierend auf Sensoraufzeichnungen, unter anderem Kamerabildern, ermitteln Objekterkennungsalgorithmen Größe, Orientierung und Position umliegender Objekte für jeden Zeitpunkt. Die Bewertung von Objekterkennungsalgorithmen setzt sich in der Regel aus zwei aufeinander folgenden Schritten zusammen. Der erste Schritt besteht aus der Assoziation der erkannten Objekte mit Referenzobjekten. Die Assoziation basiert auf einem Vergleich verschiedener Zustandsgrößen wie z.B. der Position des Objektes relativ zum Ego-Fahrzeug. Die Bewertung des Objekterkennungsalgorithmus erfolgt durch einen zweiten Schritt, in dem die assoziierten Objekte über den gesamten Datensatz akkumuliert werden. Mit Hilfe aktueller Bewertungsverfahren können Objekterkennungsalgorithmen relativ zueinander eingestuft und bewertet werden. Bisherige Bewertungsverfahren erlauben jedoch keine Schlussfolgerung, ob eine Umfeldwahrnehmung bestehend aus einem Set an Sensoren und Algorithmik für das automatisierte Fahren ausreichend ist. Diese Arbeit diskutiert mögliche Spezifikationen für die Umgebungswahrnehmung und führt eine Untersuchung bestehender Assoziations- und Bewertungsverfahren durch. Alle diskutierten Bewertungsansätze beruhen auf dem Vergle-

ich mit einer Referenzwahrheit, was bei Aufzeichnungen in der Größenordnung von Hunderten von Millionen Kilometern unpraktikabel ist. Der letzte Teil der Arbeit beschäftigt sich daher mit Methoden zur Zuverlässigkeitsanalyse der Umfeldwahrnehmung, die nicht auf der Verwendung einer Referenzwahrheit beruhen. Wenn keine Referenzwahrheit für die Ermittlung der Zuverlässigkeit benötigt würde, könnte die Zuverlässigkeit aus den Daten einer Fahrzeugflotte ermittelt werden, ohne dabei auf das zeitintensive Labeling der Daten und die Referenzsensorik angewiesen zu sein.

Abstract

Automated driving gained increasing interest in the past years. Although automated vehicles are not available yet, accidents of vehicles with highly advanced driving assistance systems start debates about automated vehicle safety in the media. To find acceptance in society, the question about automated vehicle safety has to be answered before the release of such vehicles. A benchmark for the safety of an automated vehicle is the human driver. Statistically, a fatal accident happens every few hundred million kilometers in human driving. Driving an automated vehicle with a backup driver for these many kilometers to ensure its safety is, however, infeasible. Thus, other validation approaches are required. New validation approaches are often based on subfunctions of an automated vehicle which are the perception of the environment, the path planning and the actuation of motors and steering.

This work focuses on validation approaches for the environment perception of automated vehicles. A reliable environment perception is essential as it provides the input to all subsequent functions. The work first gives a brief overview of automated vehicle validation, the perception of automated vehicles and the validation of the perception. The detection of objects that participate in the traffic is essential for a safe navigation. Not detecting an object can lead to an accident immediately or intermediately. Hence, the validation of the environment perception often focuses on object detection. Object detection algorithms provide object detections for every sensor recording, or frame, which is taken in equidistant time intervals. The evaluation of object detection algorithms usually consists of two subsequent steps. The first step associates object detections with reference truth objects within a single frame. The association is based on a comparison of different sensor parameters like the position relative to the ego vehicle and its size. The second step accumulates the errors from all frames of the entire dataset. The evaluation measure provides a value for the performance of the object detection algorithm. Current evaluation measures mainly focus on the improvement of the environment perception using object detection. However, it is unclear what specifies an environment perception that is sufficient for automated driving. This work discusses sufficient specifications for the environment perception and performs an investigation of existing association and evaluation measures. All discussed evaluation approaches rely on the generation of a reference truth which is impractical for recordings in the order of hundreds of millions of kilometers. The last part of the thesis deals with methods that may not rely on a reference truth. When no reference truth is needed, one could circumvent the necessity for long tests with backup drivers.

List of Publications

Parts of this thesis are taken from our publications of which Marco Kryda was the first author. The publications include the following journal article:

Kryda, **M.**, Berk, M., Qiu, M., Buschardt, B. & Straub, D. Assessing Perception Sensor Reliability from Field Tests Without Reference Truth. *SAE Technical Paper 2023-01-5078.*

Furthermore, the publications include the following conference contributions:

Kryda, **M.**, Qiu, M., Berk, M., Buschardt, B., Antesberger, T., German, R. & Straub, D. *Associating sensor data and reference truth labels: A step towards SOTIF validation of perception sensors* in *Sixth IEEE International Workshop on Automotive Reliability, Test and Safety (ARTS)* (Oct. 2021).

Kryda, **M.**, Berk, M., Buschardt, B. & Straub, D. *Validating an approach to assess sensor perception reliabilities without ground truth* in *SAE WCX World Congress Experience Digital Summit* (SAE International, Apr. 2021).

Each section that is taken from either of the publications starts with a short note that includes the reference. Subsections within such a section are also taken from the reference. One additional conference contribution was generated in collaboration. No parts of this work are included in the thesis. However, for completion, this contribution is listed here:

Qiu, M., **Kryda**, **M.**, Bock, F., Antesberger, T., Straub, D. & German, R. *Parameter tuning for a Markov-based multi-sensor system* in *Software Engineering and Advanced Applications (SEAA)* (Sept. 2021).

Contents

1 Introduction **1**

2 Aspects of the perception of automated vehicles and their reliability **4**
2.1 Reliability requirements for the release of automated vehicles 4
2.2 Approval trap 4
2.3 Functional approach 4
2.4 Automated vehicle perception 5
2.4.1 Camera 5
2.4.2 LiDAR 6
2.4.3 RADAR 8
2.4.4 Object detection 8
2.4.5 Free space detection 10
2.4.6 Sensor fusion 10
2.5 Assessing the perception reliability 11
2.5.1 Association 11
2.5.2 Perception evaluation 12

3 Aspects of association measures and error definitions for the perception evaluation in individual time frames **18**
3.1 Existing association measures 20
3.1.1 Investigation outline 20
3.1.2 Investigation of existing association measures 21
3.2 Grid-based association measure 31
3.2.1 Association measure 32
3.2.2 Results and discussion 32
3.3 Trajectory-clustering-based association measure 34
3.3.1 Association measure 34
3.3.2 Results and discussion 36
3.4 Distance-weighted association using threshold-based measures 38
3.4.1 Concept behind a distance-weighted association 40
3.4.2 Threshold-based association measure in polar coordinates 42
3.4.3 Threshold-based association measure with interpolation at bounding box corners 44
3.4.4 Results and discussion 48

3.5 Drivable area-based error definition . . . 50
3.5.1 Drivable area, free space and occupancy grid . . . 51
3.5.2 Deriving the drivable area from LiDAR data . . . 51
3.5.3 Adjustments of LiDAR-based drivable area using morphological operations . . . 54
3.5.4 Error definition . . . 55
3.5.5 Results and discussion . . . 58
3.6 Parallels and comparison with the human perception . . . 60
3.6.1 Human resolution . . . 61
3.6.2 Human depth perception . . . 62
3.6.3 Discussion . . . 65
3.7 Conclusion . . . 67

4 Reference-truth-based perception evaluation and reliability assessment aggregated across frames 68
4.1 Situation-dependent safety considerations . . . 69
4.1.1 Impact of vehicle constellations on the perception performance . . . 69
4.1.2 Discussion . . . 71
4.2 Existing evaluation measures . . . 73
4.2.1 Investigation outline . . . 73
4.2.2 Investigation of existing evaluation measures . . . 74
4.3 Combining association measure and evaluation measure . . . 88
4.4 Conclusion . . . 90

5 Assessing the perception reliabilities without reference truth 92
5.1 Learning the sensor reliabilities using the grid-based association . . . 92
5.1.1 Validation . . . 93
5.1.2 Results . . . 93
5.1.3 Discussion . . . 101
5.2 Learning the sensor reliabilities using the trajectory-clustering-based association . . . 104
5.2.1 Validation . . . 104
5.2.2 Results . . . 105
5.2.3 Discussion . . . 111
5.3 Considering confidence values of object detection algorithms in the perception evaluation . . . 115
5.3.1 Method . . . 116
5.3.2 Validation . . . 117
5.3.3 Results . . . 119
5.3.4 Discussion . . . 123
5.4 Conclusion . . . 124

6 Discussion and conclusion **126**
6.1 Discussion 126
6.2 Outlook 129
6.3 Conclusion 130

A Acknowledgements **141**

B Symbols **143**

C Acronyms **146**

D Bounding box contour **148**

1 Introduction

Automated driving is gaining increasing attention in the automotive scene. Although no fully automated vehicles are available yet, accidents that involve vehicles with advanced driving assistance systems are thoroughly discussed in the media. Thus, the discussion about the release of automated vehicles is commonly associated with safety arguments which raise the question as to what specifies an automated vehicle to be safe enough to operate on public roads. These questions have to be answered first to establish authority requirements and to find acceptance in society for such vehicles.

A common statement is that automated vehicles are supposed to be safer than human drivers [1, 2]. Arguments for automated driving are that perception sensors can have a 360° view of the surrounding area, an automated vehicle cannot be distracted and also cannot be influenced by alcohol, fatigue, or drugs [3, 4].

Representative accident statistics about human driving are accessible. These statistics are based on billions of kilometers of driving due to the immense number of vehicles that are deployed worldwide. On average a fatal accident occurs every 6.6×10^8 km of human driving, resulting in an estimate for the rate of fatal accidents of $1.52 \times 10^{-9}\,\mathrm{km}^{-1}$ [5, 6]. For automated driving similar amounts of data are not yet accessible. Driving that many kilometers with a backup driver on the steering wheel to prove the safety of an automated vehicle is infeasible [7]. In addition, currently accepted validation approaches are not applicable to the environment perception of an automated vehicle.

New validation approaches for automated vehicles are often based on subfunctions which are: The perception of the environment, the planning of the ego trajectory and the actuation of the steering and the motors. This is summarized as sense, plan, act [8]. Sense, which provides the input to the automated vehicle on which all subsequent functions are based, is essential for its safety.

Consequently, an approach is required to assess the reliability of the environment perception of automated vehicles. A major task of the environment perception is the detection of surrounding objects that participate in the traffic. Not detecting objects that are present or detecting objects that are not present, also known as ghost objects, can result immediately or intermediately in accidents.

Object detection algorithms are often based on neural networks. The interpretation of individual components of neural networks is difficult and does not provide an interpretable estimate of their reliability. Neural-network-based approaches are, therefore, often described as black boxes [6, 9–11]. To make object detection algorithms comparable, an evaluation based on the algorithm's results is commonly performed [12–15]. Thus, object

detection is implicitly associated with the testing and validation of the algorithms. However, while allowing a relative comparison between different object detection algorithms, the utilized validation approaches do not answer the question of whether the reliability of object detection is sufficient for automated driving.

This question includes three major challenges, which have to be addressed: The first challenge deals with how to determine the reliability of the environment perception such that the derived value allows the classification into a sufficient or an insufficient perception for automated driving. While a comparison with human driving is intuitive on the vehicle level by using the rate of accidents, an evaluation on the perception level is not as intuitive. We address this challenge by investigating and categorizing existing validation measures. Moreover, we investigate principles of human depth perception to make it comparable to the environment perception of an automated vehicle.

The second challenge deals with the fact that object detection does not fully represent the environment. First, object detection is usually limited by a predefined set of object classes. Second, bounding boxes are commonly used to mark detected objects which only provide very rough approximations of objects. Third, some components of the environment such as guard railings and roads cannot be classified as objects. Nevertheless, it is important to detect guard railings and roads as well. Thus, instead of evaluating the perception based on object detection, we introduce an error definition for the frame-wise detection of the area that is accessible by the ego vehicle.

The third challenge deals with the fact that too little data with reference truth exist: Using humans as the benchmark, the rate of fatal accidents introduced by automated vehicles and their environment perception should not exceed the rate of fatal accidents caused by human driving. A statistical estimation of the perception-induced failure rate that is in the order of the human-induced failure rate requires data for distances larger than 6.6×10^8 km [5, 6]. Berk et al. [16] propose a method that exploits the sensor redundancy in automated vehicles to obtain an estimate of the sensor reliabilities which has only been tested with simulated data. We deploy the method with two real-world datasets and compare the estimated reliability values with reference values. Furthermore, we extend the method with a mathematical formalism to incorporate additional information from object detection algorithms into the evaluation.

The work is structured in the following way:

The second chapter provides an overview of the reliability analysis in automated driving, its environment perception and the validation of the perception. It starts with an overview of concepts used for electronic components in the automotive industry and describes why common validation approaches are not sufficient for automated driving before putting more focus on vehicle perception. Finally, the chapter presents validation concepts for the environment perception that are used throughout this work.

The third chapter deals with the evaluation of individual frames. This includes the association between detections and reference objects or the association between detections from multiple sensors. The association is the first part of the evaluation of the object-detection-based environment perception. The chapter starts with a comparison of association mea-

sures utilized in the object-detection-based environment perception for automated driving. The chapter proceeds with a description of additional association measures. Subsequent chapters partly rely on these association measures. Moreover, the chapter proposes an alternative way to evaluate individual frames. Finally, the chapter analyzes human perception, which is sufficient for the task of driving and quantifies the limitations of human perception.

The fourth chapter deals with the evaluation of object detection algorithms based on an entire dataset. The perception evaluation of the entire dataset is the second part of the perception validation after the evaluation of every individual frame. Chapter four first illustrates the difference between the scenario-based perception evaluation and the stochastic perception evaluation. The chapter continues with a comparison of existing evaluation measures, categorizing them as relative and absolute evaluation measures.

The fifth chapter focuses on the object-based environment perception evaluation without a reference truth by deploying the model from [16]. The evaluation is performed on two different datasets.

The sixth chapter provides a final discussion and a future outlook.

2 Aspects of the perception of automated vehicles and their reliability

2.1 Reliability requirements for the release of automated vehicles

Automated vehicles will intensely rely on electronic components to navigate on public roads. A safe automated vehicle requires these components to operate reliably.
ISO 26262 describes a validation standard for electric and electronic systems that run on road vehicles. However, even if all components conform to ISO 26262 and operate according to their definition without error, risks remain in automated driving [17].
Especially in perception, even though the sensor electronics operate reliably, some objects may not be detected or the behavior of traffic participants is not properly estimated [18]. Hence, ISO 21448:2022 introduces the safety of intended functionality (SOTIF) and provides a validation scheme for the use of components that are not fully described by their functional safety [18, 19].

2.2 Approval trap

Human driving is often taken as a benchmark for traffic safety. According to public road statistics an accident with fatal consequences occurs every 6.6×10^8 km on average [5, 6]. Driving an automated vehicle with a backup driver for such long distances in order to validate the safety of the vehicle is infeasible [2, 4]. Without the release of automated vehicles, the necessary amount of data will never be accessible. However, when driving that many kilometers is required for the release, these vehicles will never be released. This paradox is called the approval trap [20].

2.3 Functional approach

The aim of the validation is to ensure vehicle safety while being economically feasible. Regular validation approaches in the automobile industry are not suitable for automated driving. Proven validation approaches in particular for automated vehicles do not yet exist and "driving to safety" is impractical. Thus, new validation schemes together with a proof of concept are necessary before the release of automated vehicles [4, 20].

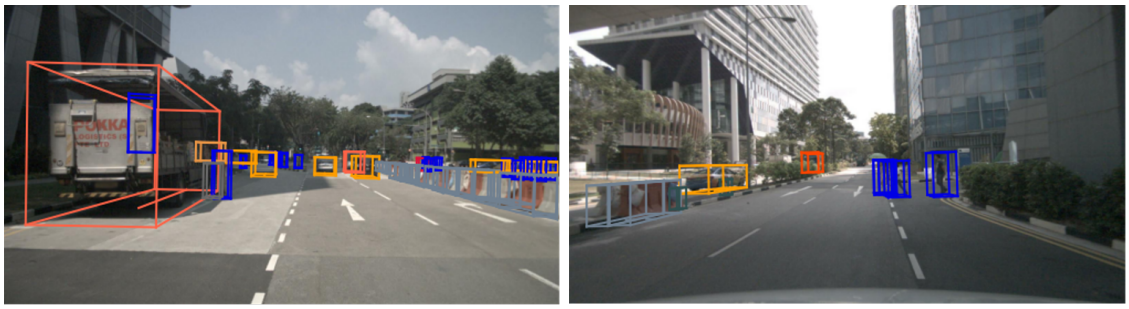
New validation approaches for automated vehicles often focus on individual or a combination of the three functions described by sense, plan and act [13, 18, 21, 22]. Under the assumption of independence between failure rates of the individual functions, the failure rate of the automated vehicle is evaluated as $\lambda_{sys} = \lambda_{sense} + \lambda_{plan} + \lambda_{act}$ [8].
With respect to the validation of automated vehicles, one has to be aware that every function of the automated vehicle has to be more reliable than the entire system as all function failure rates add to the failure rate of the system. Hence, the statistical approach of "driving to safety" for validating an individual function will require even more kilometers for the result to be significant.
The focus of this work is lying on the validation of sense which corresponds to the environment perception of the vehicle. As sense provides the input to the subsequent functions, its reliability is essential.

2.4 Automated vehicle perception

The intention of the environment perception is to provide a digital representation of the vehicle's surroundings. A digital representation is required to plan the trajectory of the automated vehicle.
This includes but is not limited to lane detections, traffic light detections, traffic sign detections and the detection of other traffic participants like pedestrians and other vehicles [18, 23]. Other environment models are based on a free space representation which describes the area that can be accessed by the ego-vehicle [24–26].
Perception sensors used for generating a digital representation include cameras, RAdio Detection And Ranging (RADAR) sensors, Light Detection And Ranging (LiDAR) scanners [23]. All three sensor types operate in different ranges of the electromagnetic spectrum.
The following section introduces the data obtained from the three different types of perception sensors. For illustrating the data obtained from any of the three different types of perception sensors in the following subsections, a frame of the nuScenes dataset is used [27].
A comparison of perception sensors can be found in [23].

2.4.1 Camera

Cameras are passive sensors that receive the light emitted and reflected by surrounding objects. Cameras operate in the human visible range of the electromagnetic spectrum and provide highly resolved RGB images of the environment. As a result, camera images are relatively easy to interpret. One main advantage of cameras is their cheap production costs. This allows the utilization of many cameras to obtain a 360° view around an automated vehicle.
In comparison to other sensors, cameras do not provide distance measurements which makes an estimation of the positions of surrounding objects challenging [12]. However, systems of multiple cameras and systems of non-stationary cameras allow distance estimations of objects in the surrounding environment [28–30].
The perception of cameras has a strong dependence on weather conditions and is crucially



(a) Front camera image (b) Rear camera image

Figure 2.1: Front and rear camera images from a frame of the nuScenes dataset with annotated bounding boxes [27]. The bounding boxes provide an approximation of the real-world objects.

limited in rainy, snowy and foggy situations and bad light conditions, e.g., at night and dawn. Cameras that operate in the infrared range of the electromagnetic spectrum can provide higher contrast in bad light conditions. Especially for the detection of warm bodies of humans and animals infrared cameras can be beneficial [23].
Figure 2.1 shows the camera images of a frame taken from the nuScenes dataset [27]. The camera images were taken during clear weather conditions.
In summary, cameras offer a low-cost sensor with a high resolution that highly correlates with human perception resulting in an easy interpretation of the recordings. However, cameras do not provide an explicit distance measurement and their perception strongly depends on the weather conditions.

2.4.2 LiDAR

LiDAR sensors are active sensors and are based on the time-of-flight principle: they send out laser pulses and measure the time difference between the emission of the laser pulse and the measured reflection of the laser pulse at a surrounding object. Using the speed of light, the distance to the surrounding object can be evaluated.
A LiDAR scanner repeats the process for different azimuth and horizontal angles to obtain a 2D or 3D representation of the environment. Recordings at different angles are obtained from multiple laser diodes that are aligned on a rotating axis which orients the laser diodes in different directions [23].
An example of a present-day LiDAR is the Puck LITE from Velodyne which has 16 lasers diodes or channels, scanning an azimuth range from $-15°$ to $15°$. Its axis rotates at a rate of 5 Hz to 20 Hz allowing it to take approximately 300000 distance measurements per second with a $360°$ field of view (FOV) [31].
Disadvantages of LiDAR scanners that are based on the physically rotating parts are that they are relatively expensive and that they are affected by vehicle vibrations [32]. Currently, solid-state LiDAR scanners are on the rise to overcome the problem of moving parts. So far, the issue with solid-state LiDARs was their limited FOV and the limited number of measurements per unit time as sensors with only a few hundred pixels were

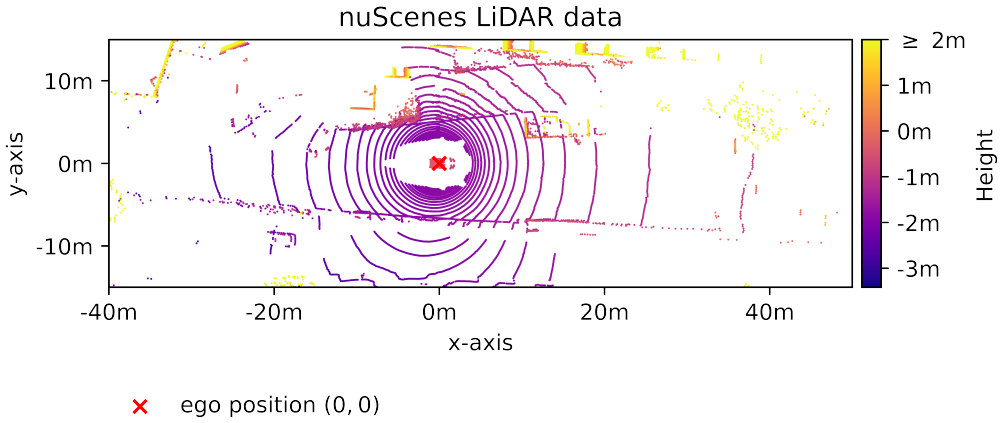


Figure 2.2: LiDAR data of the nuScenes dataset [27]. The height of the data is indicated by the colormap. Camera images of the scene are shown in Figure 2.1.

realized. Recent developments in solid-state LiDAR scanners immensely increased the number of measurements per unit time [33].

LiDARs used in automated driving commonly operate in the infrared range of the electromagnetic spectrum at wavelengths close to 905 nm or 1550 nm [23, 31]. 905 nm LiDARs require less energy compared to 1550 nm LiDARs as 1550 nm light is absorbed by water in the atmosphere. 1550 nm LiDARs, however, have a safety advantage over 905 nm LiDARs as their light is absorbed by the liquid of the human eye due to the higher absorption by water [23].

The high absorption by water also results in a bad performance in rain. In general, the performance of LiDAR sensors strongly decreases in harsh weather conditions [23, 34]. Besides absorption of the laser beam, rain and snow lead to false positives due to reflections at close distances. Moreover, water and dust dispersed by other vehicles can lead to false positives.

In addition, LiDAR detections depend on the reflectivity of objects. Objects that absorb most of the laser light rather than reflecting it cannot be detected by the sensor. Mirroring of the infrared laser light, also known as specular reflection or regular reflection in comparison to diffuse reflection, can lead to False Positive (FP) detections [35]. Furthermore, LiDAR sensors cannot detect transparent objects which can refract the laser, leading to the detection of the object beyond the transparent object [35].

Figure 2.2 illustrates the LiDAR data of a single frame taken from the nuScenes dataset [27]. The height of the LiDAR measurements is indicated by the underlying colormap. Camera images of the frame are shown in Figure 2.1.

In summary, due to the great resolution of LiDAR sensors, they play a key role in many automated driving applications and in research about automated driving perception. High resolving LiDAR sensors are also often used as reference sensors. However, despite their high resolution some optical challenges remain besides their high production costs.

2.4.3 RADAR

RADAR operates in the GHz frequency range of the electromagnetic spectrum. Radar systems for vehicle assistance systems and for automated driving work at frequencies of 24/77/79GHz [23].
Like LiDAR, RADAR sensors are active sensors that send out radio signals and receive the signal echoes of surrounding objects [23, 36]. The distance to the surrounding object can be evaluated using the time of flight of the signal emitted by the transmitting antenna. In order to achieve an estimation of the distance using the time of flight of the signal, the emitted signal is altered over time either by emitting radio frequent pulses or using a frequency modulated (FM) signal over time. RADAR with multiple-input, multiple-output (MIMO) waveforms are common in the field of automated driving [37]. MIMO sensors have multiple transmitting and receiving antennas which allows an angular resolution besides the measurement of the distance and the relative speed [37, 38].
RADAR operates at frequencies that allow to measure changes in the frequency due to the Doppler effect. Objects that move relative to the ego vehicle alter the frequency of the radio signal when being reflected. The shift in the frequency provides a relative speed estimate along the radial axis between RADAR sensor and the detected object.
The emitted radio signal of RADAR sensors gets reflected by the street. This allows RADAR sensors to perceive signals of vehicles beyond the vehicles that are right in front of the ego vehicle. Moreover, the emitted radio signal by RADAR sensors is insensitive to harsh weather conditions and allows the detection at far distances.
The angular resolution of RADAR sensors, however, is limited. For a single-input, multiple-output (SIMO) RADAR sensor with four receiver antennas the angular resolution is about 30° and one with eight receiver antennas about 15° [39]. Antennas are half the wavelength apart from each other [38]. In case of a 77 GHz RADAR half the wavelength is approximately $\lambda/2 \approx 2\,\text{mm}$.
Figure 2.3 demonstrates the preprocessed RADAR data obtained from the nuScenes dataset with the underlying LiDAR data [27]. The data of the five different RADAR sensors of the nuScenes vehicle is demonstrated in different colors. The RADAR detections are shown as a scatter plot. The velocity of the individual RADAR detections is indicated by the arrows.
In summary, RADAR sensors can to a certain extent perceive objects in occluded regions. Moreover, RADAR sensors provide a direct measurement of the radial velocity of other objects while being mostly insensitive to weather conditions. However, due to the physical properties of radio frequent signals, the angular resolution is sparse and the signal is prone to reflections that lead to ghost objects.

2.4.4 Object detection

Object detection is the part of the environment perception that focuses on identifying traffic participants like pedestrians, cyclists, animals and other vehicles. In the perception of automated vehicles, the focus usually lies on dynamic/movable objects [13, 21, 27, 40, 41].

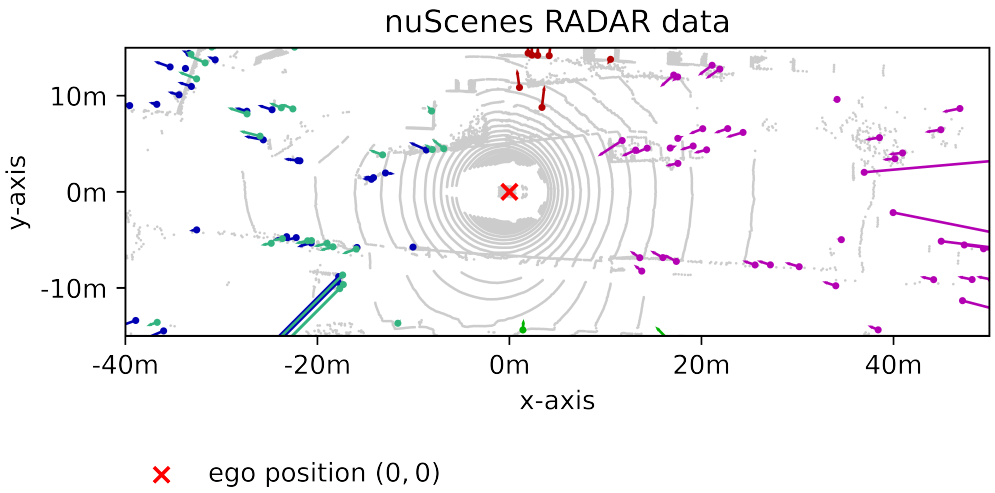


Figure 2.3: RADAR data of the nuScenes dataset [27]. For comparison RADAR data is shown on top of the LiDAR data which is visualized in grey. The five different colors indicate the data from the five different sensors located at the front, the front right, the front left and the back left and right. The corresponding camera image of the scene is shown in Figure 2.1.

Object detection algorithms return so called bounding boxes that encircle the detected object. Bounding boxes provide a rough estimation of surrounding objects. In 2D, a bounding box corresponds to a rectangle while, in 3D, a bounding box corresponds to a rectangular cuboid. Additional angles describe the relative orientation to the ego position in 3D. Object algorithms for automated driving usually provide only a single angle around an axis perpendicular to the driving plane. All these values for every object within a frame are summarized in a table. Object data refers to a list of the tables from all frames of a dataset.

Figure 2.1 illustrates the reference object data that is annotated by human labeling. The object data is projected onto the camera images. It gives an impression of possible deviations between the bounding boxes and the real-world object. In particular, many real-world objects do not have sharp edges and, therefore, do not agree well with a bounding box. An advantage of object detection is the straightforward estimation of the objects' future positions by translating the bounding boxes.

Object detection algorithms are often based on neural networks (NNs). The components of NNs are usually difficult to interpret. Therefore, NN algorithms are also described as black boxes [6, 9–11].

In order to compare neural networks and object detection algorithms an output-based validation is commonly performed. As object detection gained increasing research interest in the past decade, different validation procedures exist for object-detection-based environment perception [18]. This work investigates validation approaches for object detection with a focus on the reliability analysis of the object-based environment perception of automated vehicles.

2.4.5 Free space detection

Besides object detection, an additional representation of the surrounding environment is provided by free space detection. In comparison to the object-detection-based environment perception, free space detection allows a more detailed representation of the environment. The more detailed representation comes with an increase in the computational effort.
Multiple definitions of free space exist. The idea behind free space is to generate a grid map of the regions that are not accessible by a robot or an automated vehicle. Some free space approaches take the data of multiple frames into account to generate a more thorough map, a so-called occupancy grid [42, 43], while others explicitly define the free space within a single data frame [44]. Free space does not necessarily mean that it is accessible by the robot or the automated vehicle. Present definitions may not account for the fact that some free space regions might be too narrow to be accessible by the robot/vehicle or that they are only accessible on an indirect route.

2.4.6 Sensor fusion

Commonly, multiple types of perception sensors are utilized to obtain a more detailed and complete digital representation of the environment by fusing the sensor data.
Sensor data fusion can be performed on different data levels [26, 45].
The raw sensor data like the pixel color from the camera can be projected on the LiDAR data. An object detection algorithm can then be applied to the combined sensor raw data. A single object list is obtained from all sensor data [26].
Alternatively, sensor fusion can be performed on the object data level. An object detection algorithm is applied on the data of every sensor individually resulting in as many object lists as sensors. The obtained object lists from the different sensors are then fused together into a single coordinate frame [26].
The advantage of the raw data fusion is the utilization of all complementary information of the different sensors in order to obtain object lists from the object detection algorithm. This can result in more complete object lists in comparison to object lists obtained from a single sensor.
In comparison, sensor fusion on the object data level allows an easier interpretation. False detections can be related to sensor-specific properties. Fusion on the object data level provides redundancy on the object level which allows a comparison of the results obtained from neural-network-based object detection algorithms. However, while being beneficial in its interpretation, object-based sensor fusion comes with the disadvantage that complementary information of different sensors might be neglected due to limitations of object fusion algorithms [26].
This work utilizes an approach that relies on redundant sensor data in the form that either an object is present or no object is present. The perception evaluation in this work is, therefore, mainly based on redundant object data. As a result, fusion on the object level is utilized.

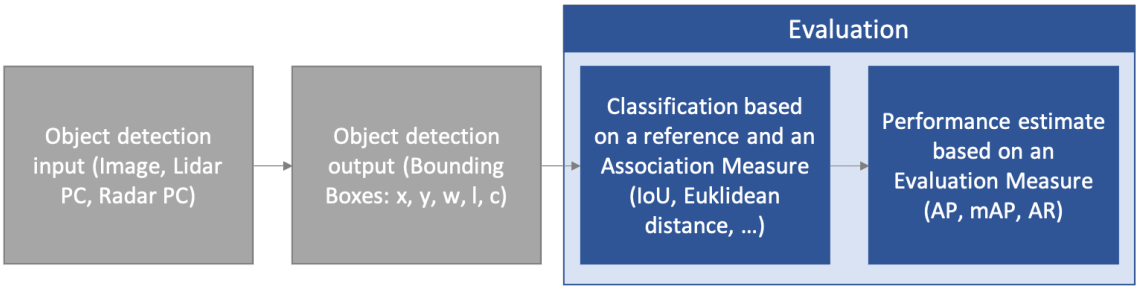


Figure 2.4: Flowchart of the object detection and evaluation pipeline. Starting with raw sensor data, object detection algorithms generate object lists. The evaluation of the object detection algorithm usually consists of two steps: the association, which allows a classification of each detection and reference object as correctly or wrongly identified, and the evaluation, which accumulated the correct and wrong detections.

2.5 Assessing the perception reliability

Safe automated driving requires a reliable environment perception. Thus, testing the perception and assessing the reliability of the perception is a crucial step for automated vehicle's safety assurance, which is required for the release of automated vehicles [8, 16, 18].

Detecting the objects in the surrounding environment of automated vehicles is safety critical. Not detecting an object can result in an accident, especially if the object is right in front of the ego vehicle.

Current approaches to validate the environment perception of automated vehicles are based on the object detection [18]. Image processing and pattern recognition introduced object detection prior to the use for automated vehicles [14, 15]. Therefore, evaluation procedures for object detection were initially utilized and optimized for 2D image processing.

The flowchart in Figure 2.4 visualizes the data processing and evaluation pipeline used for object detection. Starting at the sensor raw data, which corresponds to an image in image processing, object lists are generated. The object lists obtained by the object detection algorithm are then evaluated in two subsequent steps by comparing the object lists with reference object data. The reference data is usually obtained by human annotation of the data. Nowadays, state-of-the-art object detection algorithms already pre-label the data automatically (see for example [46]). The first step in the object detection evaluation compares the individual object detections with a list of reference objects in order to classify each detection as a correct or a wrong detection. This is referred to as association or as bounding box evaluation [15]. The second step in the object detection evaluation accumulates all correct and wrong detections [15, 47, 48]. The following two sections briefly introduce procedures that one can use for the association and the final evaluation.

2.5.1 Association

The association compares the detected objects of one frame with the reference objects that are present in the frame. Different measures for the association exist. Together with reference truth objects, the association allows the classification of every detection in either

of the three cases True Positive (TP), which means an object was detected and there was indeed an object, FP, which means an object was detected but no object was present and False Negative (FN) which means no object was detected even though an object was present. True Negative (TN) the case that no object is present and no object is detected is often not considered in object detection [14, 15].
One of the most commonly used association measures is the Jaccard index, also known as Intersection over Union (IoU), which compares the area or volume of detection and reference object [15, 49–53]. The requirement for object detection to be associated with a reference object is that the association measure has to exceed a predefined threshold. For the IoU, thresholds of 0.5 or 0.7 are used (see for example [49]). An object detection whose value of the association measure with a reference object is below (or above) the threshold is considered as different object instance. It might either be associated with another reference object or it corresponds to a FP. A reference object that has no detection assigned to it is a FN.
It can also occur that multiple object detections fulfill the association requirement with the same reference object within the same frame. Association algorithms, like the Hungarian algorithm [18], solve this by optimizing the sum of all values of the association measures from the same frame.
We want to point out that what we describe as association measure is also referred to as distance (for example in [18]) or as metric (for example in [53]). A mathematical definition of an association measure does not exist. The terms metric and distance are interchangeable [54]. The IoU itself is in mathematical terms not a distance as the result of the IoU of a set with itself is not equal to 0. Some studies suggest using $1 - IoU$ which is in agreement with the mathematical distance/metric definition (see for example [55]). However, [55] introduces another association measure, the generalized Intersection over Union (GIoU), that is not in agreement with the mathematical distance/metric definition as the GIoU can be negative.
Chapter 3 provides a comparison of existing association measures and introduces additional association measures that have been deployed for this work.

2.5.2 Perception evaluation

One aim of the perception evaluation is to provide a value that allows the comparison of different perception systems quantitatively with respect to their detection performance. A second aim of the perception evaluation is to provide an interpretable statement about the reliability of the perception system.
Here, the focus lies on the evaluation of object detection. Not detecting objects or detecting objects that are not present is safety critical in automated driving. Such a misdetection can result in an immediate or intermediate accident. Hence, evaluation of the object detection performance is relevant for the release of automated vehicles.
Multiple evaluation approaches exist for object detection [18]. Most of these approaches allow a relative comparison between different object detection algorithms. However, they do not provide any reliability estimate that answers the question if these object detection

algorithms are "safe enough" [2]. As discussed in section 2.1, human driving deals commonly as a safety benchmark. However, no measure allows a comparison of the performance of human perception with the performance of the perception based on object detection algorithms.
Common evaluation approaches that proceed according to the flowchart shown in Figure 2.4 require a reference truth [1, 14, 15]. Generating a reference truth for more than 6.6×10^8 km to prove the safety of the automated vehicle perception would require driving this distance [5]. Test driving such large distances is infeasible as previously discussed. Moreover, an additional effort is associated with labeling the data in order to obtain the reference truth.
[16] introduces an approach to validate object-based environment perception by exploiting sensor redundancy rather than using a reference truth. Validation without reference truth would allow to estimate the individual sensor reliabilities from a fleet of vehicles that have the perception sensors intended for automated driving installed without using automated driving capabilities.
This section introduces first the perception evaluation and reliability estimation of object-detection-based on a reference truth. In a subsequent step, the method from [16] is introduced.

With a reference truth

A detection may be classified by an association measure into TP, FP, FN and TN. The association itself is based on a defined threshold α. Besides the threshold α for the association, confidence values are usually provided for every detection by the object detection algorithms. Hence, an additional threshold τ only includes detections with confidence values higher or equal to that threshold. The following equations index α and τ in order to emphasize the dependence on these thresholds.
Multiple evaluation measures to quantitatively assess the perception performance with a reference exist. In this section, we introduce four evaluation measures that are repeatedly used throughout this work. These four evaluation measures are *precision* and *recall*, the *Probability Of Detection (POD)* and the *Probability of False Alarm (PFA)*.
Definitions of precision and recall can for example be found in [56]. Recall r, which corresponds to the conditional probability $Pr(D \mid O)$, where D corresponds to a detection and O corresponds to an object being present. Recall is also described as sensitivity or POD and corresponds to the TP probability given an object is present [16, 56].

$$Pr(D \mid O) = r_\tau = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{2.1}$$

In the ideal case, all present objects are observed, leading to a recall value of 1 which lets one conclude that all present objects are detected.
Precision p, which corresponds to the conditional probability $Pr(O \mid D)$, corresponds to the probability that the detected object corresponds to a correct detection.

	D	$\bar{D}$	
O	TP_τ	FN_τ	recall r_τ $\frac{TP_\tau}{TP_\tau+FN_\tau}$
$\bar{O}$	FP_τ	TN_τ	
	precision p_τ $\frac{TP_\tau}{TP_\tau+FP_\tau}$		

Table 2.1: Confusion matrix. Precision corresponds to the conditional probability $Pr\left(O \mid D\right)$ while recall corresponds to the conditional probability $Pr\left(D \mid O\right)$.

$$Pr\left(O \mid D\right) = p_\tau = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{2.2}$$

The ideal case leading to a precision value of 1 is the case when all detected objects are actually present.
PFA, also known as fallout, is defined as the conditional probability that an object is detected even though no object is present [16, 56].

$$PFA = Pr\left(D \mid \bar{O}\right) = \frac{FP_\tau}{TN_\tau + FP_\tau} \tag{2.3}$$

Table 2.1 shows the confusion matrix and indicates the evaluation of precision and recall. A clear interpretation with respect to the reliability is not possible for all four evaluation measures, actually three as recall and POD are the same. This is due to the fact, that the result of the evaluation measures strongly depends on the utilized association measure and the corresponding definition of an error. Common approaches weight false detections at far distances equivalent to false detections at close distances. In this case, the resulting value from the evaluation measure cannot be related to the reliability and the resulting safety of the algorithm. A further discussion about the interpretation of the evaluation measures can be found in section 4.2.

Without a reference truth

This section is taken from our publication [57]. Some parts have been modified. It provides a brief overview of the model from [16].

Generating a dataset with a reference truth is time intense. Besides driving the vehicle in order to record the data, obtaining a reference truth is also associated with labeling the data. Like for test driving, gaining a reference truth for more than 6.6 km is infeasible [5]. [16] introduces a model that exploits the M redundant sensors to estimate the sensor reliabilities without a reference truth. Learning the sensor reliabilities without the need for a reference truth would allow to estimate the sensor reliabilities from a fleet of vehicles that have the perception sensors installed.
The model in [16] is based on a binary representation of the sensor output. Either the sensor m detects an object in a certain area of its FOV at a given time indexed n (sensor

output $S_{mn} = 1$) or it does not detect an object within this area (sensor output $S_{mn} = 0$). Hereafter, the Boolean array, which represents the sensor outputs of all M sensors, is mapped to an integer number y_n between 0 and $2^M - 1$ to increase the readability. y_n is called sensor system output. The mapping is based on the transformation from a binary number to a decimal number $y_n = \sum_{m=0}^{M-1} S_{mn} \cdot 2^m$.
The distribution of recorded sensor system outputs y_n in a certain rectangle will be different for time-frames with an object and for time-frames without an object. Therefore, the model incorporates a hidden variable, which is subject to environmental influences and corresponds to the probability of an object being present. The overall distribution of the sensor system outputs y_n is evaluated based on the total probability theorem. The probability for sensor system output y_n with a given sensor parameter set $\boldsymbol{\theta}$ including each individual sensor's POD_m and PFA_m as well as potential correlation parameters and the probability that an object is present is described by the following equation:

$$p\left(y_n \mid \boldsymbol{\theta}, p_{obj}\right) = p_{obj} \cdot p\left(y_n \mid \boldsymbol{\theta}, O = 1\right) + \left(1 - p_{obj}\right) \cdot p\left(y_n \mid \boldsymbol{\theta}, O = 0\right) \tag{2.4}$$

Here, $O = 0$ corresponds to no object being present and $O = 1$ corresponds to an object being present. The individual sensor reliabilities are quantified by POD_m and PFA_m. The probability distribution for the sensor system outputs $p\left(y_n \mid \boldsymbol{\theta}, O = 1\right)$ conditional on the fact that an object is present is based on the POD_m values. The probability distribution for the sensor system outputs $p\left(y_n \mid \boldsymbol{\theta}, O = 0\right)$ conditional on the fact that no object is present is based on the PFA_m values.
In case the sensor outputs are statistically independent, the probability of the sensor system output $p\left(y_n \mid \boldsymbol{\theta}, O = 1\right)$ and $p\left(y_n \mid \boldsymbol{\theta}, O = 0\right)$ can be described as a product of the POD_m and the PFA_m, respectively. Given an object is present in a certain area of the FOV, the conditional probability for a certain sensor system output y_n can be derived from the individual sensor probabilities of detection POD_m in the following way:

$$p\left(y_n \mid \boldsymbol{\theta}, O = 1\right) = \prod_{m=0}^{M-1} POD_m^{S_{mn}} \cdot \left(1 - POD_m\right)^{1 - S_{mn}} \tag{2.5}$$

Given no object is present, the conditional probability for a certain sensor system output y_n can be derived from the individual sensor probabilities of false alarm PFA_m:

$$p\left(y_n \mid \boldsymbol{\theta}, O = 0\right) = \prod_{m=0}^{M-1} PFA_m^{S_{mn}} \cdot \left(1 - PFA_m\right)^{1 - S_{mn}} \tag{2.6}$$

Thus, when assuming statistical independence between the sensors, the considered sensor parameters are $\boldsymbol{\theta} = \{POD_0, \ldots, POD_{M-1}, PFA_0, \ldots, PFA_{M-1}\}$.
However, in general sensor outputs are not statistically independent. As an example, several sensors might be affected by harsh weather, which leads to an increase in the probability of FN and FP errors for all these sensors under such conditions. In order to account for dependent sensor outputs, an extended model that incorporates the statistical dependence between sensors is proposed in [16]. In order to account for the statistical

dependence between the sensors, the dependent model utilizes the Gaussian copula.
In case an object is present, the conditional probability for sensor system output y_n is then described by:

$$p\left(y_n \mid \boldsymbol{\theta}, O=1\right) = \quad \Phi_{\mathbf{R}_{POD,+}}\left(\begin{bmatrix} (2S_{0n}-1)\cdot\Phi^{-1}\left(POD_0\right) \\ \vdots \\ (2S_{M-1n}-1)\cdot\Phi^{-1}\left(POD_{M-1}\right) \end{bmatrix}\right) \tag{2.7}$$

Φ^{-1} corresponds to the univariate inverse normal CDF. $\Phi_{\mathbf{R}_{POD,+}}$ is the multivariate Gaussian cumulative distribution function (CDF) with correlation matrix $\mathbf{R}_{POD,+}$ and a mean value of zero. The plus-sign indicates that the sign of component ij of the correlation matrix $\mathbf{R}_{POD}$ is changed if the results of sensor i and sensor j are in disagreement. Switching the signs simplifies the integration that is included in the evaluation of the multivariate Gaussian CDF. The term $(2S_{mn}-1)$ switches the sign depending on the output of the sensor that can either be $S_{mn}=0$ or $S_{mn}=1$. Due to the symmetry of the Gaussian probability density function (PDF), switching signs allows to use the multivariate Gaussian CDF for evaluating the probabilities of the sensor system y_n instead of changing the borders of the integral over the multivariate Gaussian PDF with correlation matrix $\mathbf{R}_{POD}$.
The conditional probability for the sensor system output y_n in case no object is present is described by analogy to the above equation, interchanging the POD with the PFA and the correlation matrices.

$$p\left(y_n \mid \boldsymbol{\theta}, O=0\right) = \quad \Phi_{\mathbf{R}_{PFA,+}}\left(\begin{bmatrix} (2S_{0n}-1)\cdot\Phi^{-1}\left(PFA_0\right) \\ \vdots \\ (2S_{M-1n}-1)\cdot\Phi^{-1}\left(PFA_{M-1}\right) \end{bmatrix}\right) \tag{2.8}$$

$\Phi_{\mathbf{R}_{PFA,+}}$ corresponds to the multivariate Gaussian CDF with correlation matrix $\mathbf{R}_{PFA,+}$ in case no object is present.
The full parameter set is $\boldsymbol{\theta} = \{POD_0, \ldots, POD_{M-1}, \mathbf{R}_{POD}, PFA_0, \ldots, PFA_{M-1}, \mathbf{R}_{PFA}\}$.
The number of free parameters for equation (2.4) in combination with the dependent model from equation (2.7) and (2.8) is M^2+M+1. In order to fit the distribution, the number of free parameters needs to be smaller than 2^M, which is the number of distinct sensor system outputs y_n. From the inequality $M^2+M+1<2^M$ follows that at least $M=5$ redundant sensors are required in order to apply to the model.
For M sensors, the multivariate Gaussian CDF is based on an M-dimensional integral. Solving the integral for large values of M numerically is computationally demanding, as the computational effort increases exponentially with the number M. Therefore, the full-rank correlation matrices $\mathbf{R}_{POD}$ and $\mathbf{R}_{PFA}$ are replaced with rank one Dunnet-Sobel class matrices [16, 58]. Using the Dunnet-Sobel class matrices reduces the M dimensional integral to a 1-dimensional integral. The rank one Dunnet-Sobel class matrices are of the form:

$$\rho_{ij} = \begin{cases} \lambda_i \cdot \lambda_j, & \text{for } i \neq j \\ 1, & \text{otherwise} \end{cases} \tag{2.9}$$

We refer to λ_j as Dunnet-Sobel coefficients. With the Dunnet-Sobel class matrices the sensor parameters are $\boldsymbol{\theta} = \{POD_0, \ldots, POD_{M-1}, \lambda_{O=1,0}, \ldots, \lambda_{O=1,M-1}, PFA_0, \ldots, PFA_{M-1}, \lambda_{O=0,0}, \ldots, \lambda_{O=0,M-1}\}$.

The likelihood of the sensor parameters $\boldsymbol{\theta}$ and the hidden variable p_{obj} can be derived from equation (2.4) together with either the independent or the dependent model. According to Bayes' rule, the posterior of the parameters is derived from the likelihood and the prior distribution:

$$f(\boldsymbol{\theta}, p_{obj} \mid \{y_1, \ldots, y_N\}) \propto f(\boldsymbol{\theta}, p_{obj}) \cdot \prod_{n=1}^{N} p(y_n \mid \boldsymbol{\theta}, p_{obj}) \tag{2.10}$$

Here, $f(\boldsymbol{\theta}, p_{obj} \mid \{y_1, \ldots, y_N\})$ is the posterior distribution for the model parameters $\boldsymbol{\theta}$, $f(\boldsymbol{\theta}, p_{obj})$ is the prior distribution and $\prod_{n=1}^{N} p(y_n \mid \boldsymbol{\theta}, p_{obj})$ is the likelihood. In order to obtain an estimate for the sensor parameters, the maximum likelihood estimate (MLE) is evaluated.

In the following, the indices IM and DM will be attached to the model's probability mass function (PMF) $p(y_n \mid \boldsymbol{\theta})$ to indicate whether the parameters were estimated using the independent model from equations (2.5) and (2.6) or the dependent model from equations (2.7), respectively. For the reference distribution, which represents the PMF $p(y_n)$ based on the number of occurrences of every sensor system output, the index ref is used. We incorporate both models, the independent and the dependent model, into our pipeline. The focus of the study, however, lies on the pipeline with the dependent model. The independent model is not expected to be capable of describing the data.

3 Aspects of association measures and error definitions for the perception evaluation in individual time frames

Assessing the reliability perception of the environment perception of automated vehicles is based on an evaluation of the recorded dataset. Its output may be based on a statistical analysis of the number of times the detection was not in agreement with reality and the number of times it was erroneous. To evaluate large datasets, one requires a mathematical measure that allows a conclusion about whether sensor detection and reality are in agreement. In other words, one must define what is an error in the environment perception and what corresponds to a correct detection.

Due to the prominence of object detection in the environment perception for automated vehicles, it has become increasingly customary to employ object-based data for assessing its perceptual capabilities. In contradistinction, an alternative methodology involves evaluating environmental perception based on free space detection.

In the context of object detection delineated in section 2.4.4, a measure can be established by comparing each discrete object within a detection dataset against every object in a reference dataset, taking into account factors such as position, size, and/or velocity. A close match serves to link a detected object with a reference object, as they are assumed to represent the same entity. The process of linking detections with reference objects or other sensor detections is commonly referred to as association. Consequently, association yields an assignment of each object in one dataset to either none or one object in another dataset.

The term "association measure" is utilized in this work with two distinct connotations: firstly, to represent a continuous distance measure between two object instances originating from disparate datasets; and secondly, to signify a discrete, threshold-based measure that allocates a binary value to each object within one dataset, reflecting the presence or absence of the corresponding object in an alternate dataset. Given that the objective of association entails categorizing outcomes as either correct or incorrect, a threshold is imposed on the continuous distance measure to facilitate the derivation of the binary classification.

Object detection algorithms, aside from physical properties such as position and size, also yield confidence values τ for every detection. The association of detection with a reference object typically disregards the confidence value τ, given that the reference's presence is assured.

Instead of associating detections with reference objects, an object association between various sensors can be executed if multiple sensors are deployed. This association of detections from multiple sensors aligns with the initial stage of sensor fusion at the object data level, as briefly outlined in section 2.4.6. The association between multiple sensors is also requisite for implementing the method described in section 2.5.2 to estimate sensor reliability without reference truth.
Comparing the association of object detections from a single sensor with the reference truth to the association of object detections from multiple sensors reveals increased complexity in the latter. While a comparison with reference truth imparts information about the sensor, an association between sensors necessitates supplementary and intricate data pertaining to individual sensors. For instance, the association of object detections from multiple sensors incorporates weightings to prioritize detections from more reliable sensors over less reliable ones. Furthermore, object-based multi-sensor fusion may integrate confidence values derived from object detection algorithms. Approaches for the association between multiple sensors are introduced in sections 3.2 and 3.3. For the sake of simplicity, in subsequent sections, an object is considered present or absent, unless otherwise specified, if its detection surpasses a defined confidence threshold of $\tau = 0.5$.
This chapter addresses the various methods for comparing detections with either a reference reality or other sensor detections, elucidating the relationship between these tasks and providing a comprehensive understanding of their distinctions and interrelations.
The present chapter demonstrates various methodologies for comparing detections against a referential reality or alternative sensor data. The primary objectives of this chapter include:

- A comparison between existing association measures that are used for object detection in automated driving in section 3.1.
- An introduction of two association measures in sections 3.2 and 3.3 that were utilized to make the model from section 2.5.2 applicable with object data.
- An extension of object detection association in section 3.4 that allows a larger deviation between detection and reference object at distances further away from the ego vehicle. Larger deviation at larger distances is a common sensor property and, thus, may not be a perception error.
- A measure for assessing a perception performance from free space approaches in section 3.5. Analogue to 3.4 it incorporates a relation between the distance to detections and the possible deviations.
- An analysis of possible deviations in the human perception in section 3.6 as properties of the human perception may provide reference values for a sufficient environment perception.

Figure 3.1 highlights the focus of this chapter in the detection evaluation pipeline in blue.

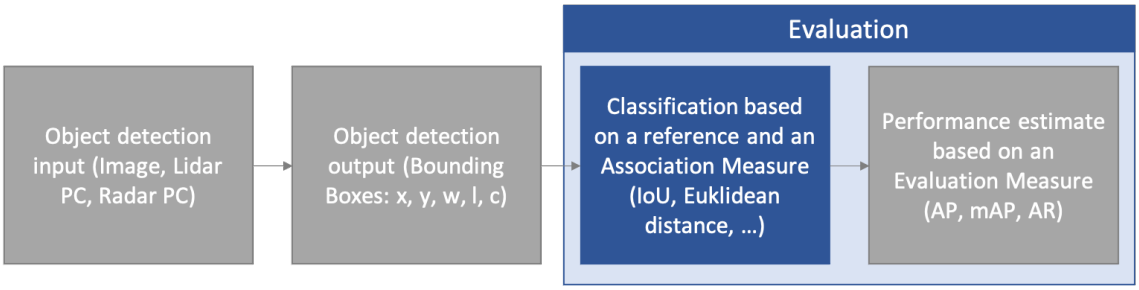


Figure 3.1: Basic representation of object detection and evaluation pipeline. Raw sensor data is provided to the object detection algorithm. The object detection algorithm yields bounding boxes as output. Data classified into TP, FP, FN and TN is obtained in the first step of the evaluation. From the classified data, a performance estimate is generated by applying one or multiple evaluation measures.

3.1 Existing association measures

Many association measures that classify detections into correct or false detections exist. In automated driving, the evaluation of object detection algorithms should relate to the vehicle's safety. Therefore, the association measure, being part of the evaluation of object detection algorithms, needs to differentiate between safety-critical and non-critical situations. Thus, to determine an interpretable evaluation approach, which is necessary for the perception validation of automated vehicles, one should start by finding an interpretable association measure that performs the differentiation into safety-critical and non-critical [59]. This motivates the comparison of existing association measures that are used for the evaluation of object detection for automated vehicles with respect to their interpretability.

3.1.1 Investigation outline

The association between detections and the reference truth commonly yields the following results: An association between a detection and a reference object classifies the detection as TP. Detections that are not associated with any reference objects are classified FPs. All remaining reference objects to which no detection has been assigned to are classified as FNs.
TNs, cases where objects are neither detected nor present, are usually not considered in object detection due to the fact that this usually accounts for the largest fraction of the FOV of the sensors. For such datasets where the ratio between the area that is occupied by vehicles and the area that is unoccupied is small, not taking the TN cases into account proofed to be beneficial [60]. As a result, TN cases are rarely defined in the perception for automated driving.
We analyze the association based on the following specifications.

- The association measure should remain interpretable which requires that defined values in the measure can be derived from persistent mathematical and physical principles. Introduced thresholds are often set arbitrarily without any physical or

Measure	Reference	Equation	Abbr.
Jaccard index/Intersection over union	[15, 49, 50] [51–53]	(3.1)	IoU
Dice coefficient F1-Measure	[62]	(3.3)	DICE
Generalized intersection over union	[55]	(3.4)	GIoU
Euclidean distance	[61, 63]	-	ρ
Distance intersection over union	[53]	(3.5)	DIoU
Complete intersection over union	[53]	(3.6)	CIoU
Support distance error	[64]	(3.8)	SDE

Table 3.1: Summary of existing association measures that are mentioned in section 3.1.

mathematical reasoning, which leads to non-interpretability. The aim of some association measures is, therefore, to keep the number of thresholds low [18, 61].

- The classification outcomes of the association measure should correlate with potentially hazardous situations. Margins of TP detections from their associated objects should be interpretable and safe to accept. A higher number of TP detections and less FN and FP detections should correspond to a safer system. Subsequently, a higher number of FN and FP should correlate with a higher number of potential accidents, corresponding to a system that is less safe.

- The association measure should be easy to evaluate in order to handle large amounts of data which does not allow individual computationally expensive operations.

The covered association measures will be discussed with respect to these three specifications.

3.1.2 Investigation of existing association measures

The association between the detections and the reference objects corresponds to the first block of the evaluation of object detection algorithms as shown in Figure 3.1. Table 3.1 lists all association measures investigated in this paper, which are used in automated driving. However, it is far from complete considering measures from other disciplines. A detailed list of measures used for 3D image segmentation beyond automated driving can be found in [62].
In the following, this section investigates different association measures. Based on examples it demonstrates the largest possible deviations for TP detections according to the different association measures and discusses when such detections are safety critical for the ego vehicle. As these illustrations indicate the largest possible deviation, we refer to them as corner cases in the following.

Measure	Fig. 3.2	Fig 3.3(a)	Fig 3.3(b)	Fig 3.4(a)	Fig 3.4(b)	Fig 3.5	Fig 3.6
IoU $\in [0,1]$	0.70	0.72	0.70	0.70	0.70	0.19	0.00
DICE $\in [0,1]$	0.82	0.84	0.82	0.82	0.82	0.30	0.00
GIoU $\in [-1,1]$	0.70	0.72	0.70	0.70	0.70	0.02	-0.25
$\rho \in [0,\infty)$ m	2.25 m	2.11 m	0.00 m	0.00 m	0.00 m	0.00 m	5.43 m
DIoU $\in [-1,1]$	0.68	0.70	0.70	0.70	0.70	0.19	-0.11
CIoU $\in [-1,1]$	0.68	0.70	0.70	0.70	0.70	-0.16	-0.13
SDE $\in [0,\infty)$ m	4.40 m	4.12 m	2.20 m	1.24 m	1.23 m	5.80 m	0.00 m

Table 3.2: Values obtained from the different association measures for the different examples in the indicated Figures. All association measures are unitless except for the Euclidean distance ($\rho(\mathbf{r}, \mathbf{r}_{ref})$) and the SDE. The values highlighted in green are larger than the threshold value of 0.7 for unitless measures or smaller than 0.5 m for distance-based measures. All examples are based on a reference object with dimension width $w_{2D,ref} = 2.5$ m and length $l_{2D,ref} = 15$ m. The evaluation of the SDE requires a position of the ego position and a driving direction.

Table 3.2 lists the investigated association measures together with the sets of their output domain. Table 3.2 also lists the output of the association measures for the different corner case examples. A chosen threshold value categorizes whether a detected object corresponds to the reference object. Here, we set the threshold to a value of 0.7 for unitless measures and a value of 0.5 m for distance-based measures. The same threshold values are utilized in [49] for the IoU and in [48] for the Euclidean distance. In the case of unitless measures, values greater or equal than the defined threshold of 0.7 correspond to a match between detected object and reference object. In the case of distance-based measures, values smaller than the defined threshold of 0.5 m correspond to a match between detected object and reference object. Table 3.2 highlights matches between detected objects and reference objects according to the different association measures in green.

Intersection over union/Jaccard index

The IoU is the most commonly used association measure [15, 49–53]. For two sets (shapes, volumes) $A, B \subseteq \mathbb{S} \subseteq \mathbb{R}^n$ the IoU is defined as:

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{3.1}$$

In the case of object detection for automated vehicles, the sets A and B usually refer to all points within the 2D area of the bounding box projection on the driving plane or the 3D volume of the bounding box for the detected object and the reference object, respectively. Figure 3.2 demonstrates a possible corner case for the IoU that can correlate with a potentially hazardous situation. The green bounding box corresponds to the reference truth

while the bounding box in red with the dashed line corresponds to the detection. For the demonstration a length of 15 m is chosen which is approximately the length of a truck. A detection of an object that is 15 m in length can be up to 4.5 m shorter and still fulfill the requirement of $IoU \geq 0.7$ for a match between detected object and reference object as utilized in [49]. A detection with a deviation of 4.5 m may be sufficient for an object that is far away, however, a detection that is 4.5 m further away than the actual object can cause an accident if the object is right in front of the ego vehicle. Furthermore, the IoU only considers the orientation of an object to a certain extent. The orientation of the vehicle may, however, be crucial, e.g., it makes a big difference whether a vehicle is considered to stay in a neighboring lane on the motorway or whether the vehicle changes into the lane of the ego vehicle.

In [64] it is shown that the IoU of two associated objects does not correlate with the number of potential crashes. In [64] a crash is defined as an overlap of an object bounding box with the ego-vehicle's bounding box that is increased by 80 %. The average IoU of a detected object and the reference object turned out to be approximately the same for cases where both, the reference and the detection, are involved in a crash and for cases where either the reference or the detection are involved in a crash. [64] also refers to crashes of the hypothetically increased ego-vehicle with a reference and also its associated detection as TP crash while crashes of the hypothetically increase ego-vehicle with only a reference object or only a detection are referred to as FN crashes and FP crashes, respectively. The conclusion from Figure 3.2 and the observations in [64] is that, especially for close distances, the IoU may not be a good measure for collision avoidance and, therefore, it may not correlate well with the number of hazardous situations. To account for differently sized objects in images due to their distance from the camera, [65] introduces an object size dependent threshold α instead of a constant. The threshold depends on the width w_{img} and height h_{img} of the reference bounding box B_{img}. As [65] focuses on object detection in images, the parameters w_{img} and h_{img} of bounding box B_{img} are measured in pixels. The threshold is adjusted for objects that are smaller than 10×10 pixels.

$$\alpha\left(B_{img}\right) = \min\left(0.5, \frac{w_{img} h_{img}}{(w_{img}+10)\cdot(h_{img}+10)}\right) \tag{3.2}$$

An object size dependent threshold could also be developed in 3D. Otherwise common thresholds α for the association measure are the constant values 0.5 and 0.7 [15, 47, 66, 67]. Especially for rectangular bounding boxes but also for generic pixel-wise evaluations the IoU is fast to evaluate which also makes it a common association measure in many evaluations.

Dice coefficient

The Dice coefficient is a measure that is frequently used in semantic segmentation [62, 68, 69]. For two sets (shapes, volumes) $A, B \subseteq \mathbb{S} \subseteq \mathbb{R}^n$ the Dice coefficient is defined as [62]:

$$DICE = \frac{2 \cdot |A \cap B|}{|A| + |B|} \tag{3.3}$$

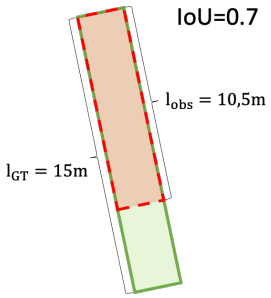


Figure 3.2: A match between a detection (red) and a reference object (green) according to the IoU with a threshold of $\alpha = 0.7$. The detection would be associated with the reference even though there is a deviation of 4.5 meter at the back side of the truck. A deviation of 4.5 meter can be essential when the object is right in front of the ego vehicle.

Like the IoU, the dice coefficient is equal to zero in case of no intersection between the two sets A and B and equal to one in case the two sets A and B are equal. In all other cases the dice coefficient is larger than the IoU as

$$DICE = IoU \cdot \frac{2 \cdot |A \cup B|}{|A| + |B|} > IoU \cdot \frac{2 \cdot \max(|A|, |B|)}{|A| + |B|} > IoU$$

As a result of the scenario in Figure 3.2 the dice coefficient returns a value larger than 0.7 as shown in Table 3.2. The question of how to choose the threshold α for associating two objects remains. The same threshold α as for the IoU of 0.7 or 0.5 from [49], respectively, yields a less restrictive measure for the association between objects in case of the Dice coefficient. Like the IoU the Dice coefficient will also not directly correlate with potentially hazardous situations. The calculation effort is comparable to the IoU and allows an evaluation for large-scale datasets.

Generalized intersection over union

The GIoU is an extension to the IoU that addresses an issue that can arise when dealing with non-overlapping bounding boxes. This issue, known as the vanishing gradient problem, makes the learning process of neural networks difficult or impossible. By incorporating an additional term, GIoU can help to avoid this problem and improve the performance of these algorithms [55]. For two sets (shapes, volumes) $A, B \subseteq \mathbb{S} \subseteq \mathbb{R}^n$ the GIoU utilizes the smallest convex set $C \subseteq \mathbb{S}$ that contains the sets A and B.

$$GIoU = \frac{|A \cap B|}{|A \cup B|} - \frac{|C \setminus (A \cup B)|}{|C|} \tag{3.4}$$

For close to perfectly aligned sets A and B the GIoU tends towards 1 as the first term corresponds to the IoU and the subtracted term tends towards 0 because the smallest convex set C becomes equivalent to the union of A and B. For two sets A and B that do not intersect, i.e. $A \cap B = \emptyset$, the GIoU tends towards -1 as the IoU term becomes 0 for no intersection between the sets A and B and the further the sets A and B are apart, the larger is the smallest convex set C, such that the second term tends towards -1 when C becomes much larger than the union of A and B. Analog to the IoU a threshold α needs to

be chosen such that detected objects are identified as correctly or falsely detected objects when being compared with the reference objects. When using a threshold α, the GIoU yields the same difficulties as the IoU as shown in Figure 3.2. Therefore, the GIoU may not correlate well with critical situations analog to the IoU. An advantage of the GIoU in comparison to the IoU is the faster decrease in the GIoU for example for slightly rotated objects due to the increasing second term that is subtracted from the IoU value in the GIoU.

Distance intersection over union

Similar to the GIoU, the Distance Intersection over Union (DIoU) extends the IoU with an additional term in order to avoid the vanishing gradient problem for non-overlapping bounding boxes [53]. Its introduction aims to optimize the training of object detection algorithms based on neural networks by building upon the advancements of the GIoU [53]. However, the focus in [53] does not lie in the interpretability of the measure in terms of safety and reliability. The DIoU is defined as

$$DIoU = IoU - \frac{\rho^2\left(\mathbf{r}, \mathbf{r}_{ref}\right)}{c^2}. \tag{3.5}$$

Here, $\rho : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}, (\mathbf{r}, \mathbf{r}_{ref}) \mapsto \rho\left(\mathbf{r}, \mathbf{r}_{ref}\right)$ denotes the Euclidean distance between the center points of the detection $\mathbf{r}$ and the reference $\mathbf{r}_{ref}$. c is a normalization constant and corresponds to the diagonal of the smallest possible bounding box that includes the detection and the reference box [53]. In [53] it is shown that optimization of a neural network with a loss based on the DIoU converges faster in comparison to using the GIoU. In the following, without loss of generality, consider A to be the detection bounding box and B to be the reference bounding box. For non-overlapping sets A and B an optimization based on the GIoU results first in an increase in the detection bounding box A to minimize the second term in the GIoU. Only when the increased detection bounding box starts to overlap with the reference bounding box, does the IoU term get optimized [53]. In comparison, the DIoU prioritizes the minimization of the Euclidean distance between the center points of the detected bounding box and the reference bounding box during the regression process, while maintaining the dimensions of the fitted bounding box.
For a final decision process, a threshold α is required analogous to the aforementioned association measures. Setting the threshold α results in a loss of interpretability as pointed out previously. However, in comparison to the IoU and the GIoU, the DIoU would rate the example from Figure 3.2 worse as the bounding boxes are not aligned on their center points, which yields a non-zero second term in the DIoU and, therefore, a worse fit in comparison to a detection where the center point would be aligned with the reference. However, in case the center points line up, as shown in Figure 3.3 (b), the object's boundaries can still deviate between reference and detection by up to 2.25 m. And in case the center points do not line up even larger deviations at some boundaries can occur as shown in Figure 3.3 (a). Thus, the distance to the back side of the object in front can still be crucially misestimated even though the measures would consider it a TP in case the threshold α is set to 0.7. As a conclusion, there can be corner cases in which the association measure does not correlate

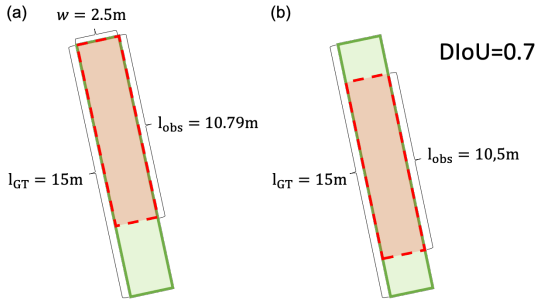


Figure 3.3: Two possible matches between a detection (red) and a reference object (green) according to the DIoU with a threshold of $\alpha = 0.7$. Even though the overlap of the reference and detection bounding box is larger compared to the case of the IoU, as shown in Figure 3.2, the deviation can still be large even though the association measure demonstrates a match. (a) demonstrates a match with the maximum possible deviation of the detection from the reference object on only one side of the object. (b) demonstrates a match with the maximum possible deviation in the area between detection and reference object.

with potentially hazardous situations. Like the GIoU, the DIoU also requires determining the smallest convex bounding box that encircles both sets A and B and is, therefore, more computationally expensive in comparison to the IoU.

Complete intersection over union

Besides the DIoU, [53] proposes the Complete Intersection over Union (CIoU). The CIoU includes an additional term that accounts for the aspect ratio of the detected object's dimension.

$$CIoU = IoU - \frac{\rho^2\left(\mathbf{r}, \mathbf{r}_{ref}\right)}{c^2} - \frac{v^2}{(1 - IoU) + v} \tag{3.6}$$

Here, the parameter v describes a function that accounts for the consistency of the aspect ratio between the length and the width of the detected object. v is assessed in the following way:

$$v = \frac{4}{\pi}\left(\arctan\frac{w_{2D,ref}}{l_{2D,ref}} - \arctan\frac{w_{2D}}{l_{2D}}\right)^2$$

For a perfect aspect ratio v becomes 0 while in case the object width and length are interchanged v becomes equal 1. By adding the additional aspect ratio term to the DIoU, the measure becomes more restrictive. For example Figure 3.3 (a) does not fulfill the requirement that the CIoU is equal or greater than 0.7 while it does fulfill the requirement for the DIoU. Figure 3.4 demonstrates two examples where the CIoU returns a value of 0.7. In the two examples, both the center distance term and the aspect ratio term are equal to 0. A maximum deviation of 1.22 m in the example from Figure 3.4 between the boundaries of detection and reference is smaller in comparison to the maximum deviation in previously mentioned association measures for a fixed threshold of $\alpha = 0.7$. In comparison to the aforementioned association measures, the CIoU is more restrictive for the same association threshold α. Thus, with the same threshold α, it classifies fewer detections as TP while

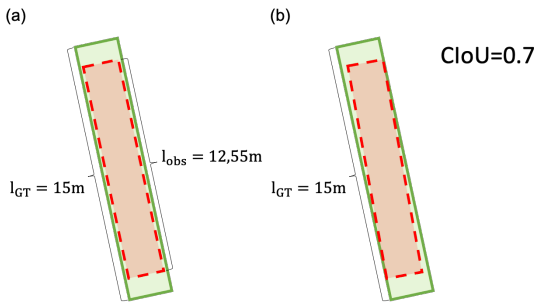


Figure 3.4: Two examples for a match between a detection (red) and a reference object (green) according to the CIoU when the threshold is set to $\alpha = 0.7$. In these two examples, both terms, the center distance and the aspect ratio term, are equal to 0. (a) demonstrates a match of the maximum possible deviation in the area between detection and reference object. A rotation of the detection relative to the reference object does not influence the value of the CIoU as long the area-wise overlap between detection and reference object is the same as shown in (b).

classifying a greater number of detections as FP. Moreover, more reference objects will be classified as FN. An TP detection is, therefore, in better agreement with the reference object. However, the same difficulties in the interpretability of the classification as for the other association measures arise. A direct correlation with the number of potentially hazardous situations is not expected likewise to the IoU.

Euclidean distance

The Euclidean distance between the center points of objects may also be used as an association measure instead of the IoU [48, 61, 63]. In order to classify the detections in either of the binary cases TP, FP and FN, a distance threshold α has to be chosen.
In the appraisal of the nuScenes challenge [63], the employed thresholds for the evaluation are 0.5 m, 1.0 m, 2.0 m, 4.0 m. The rationale behind the selection of these specific thresholds remains elusive. It is essential to consider that, in a real-world context, a deviation of 4.0 m proves unsuitable for accurately assessing nearby objects. In situations such as approaching a vehicle halted at a red traffic signal or during the process of parking even a deviation of 0.5 m may prove inadequate, especially, as one needs to take into account that the center distance does not include information about the object's size. On the one hand, this can be an advantage in particular for small objects, for example pedestrians, which in case of small deviations already show 0 IoU [48]. On the other hand, small objects can be associated with large objects due to a small Euclidean distance and, thus, imply a good fit while it may not be a good fit as demonstrated in Figure 3.5. In particular for large object large deviations may occur. Therefore, the Euclidean distance may not allow a proper interpretation after the data was classified into TP, FP and FN using a predefined threshold α. In [64] it is shown that the IoU does not correlate well with crashes which can be explained by the fact that the deviation of the object's corner closest to the ego vehicle can be quite large as shown in Figure 3.2, even though the IoU is greater than the applied threshold α. The distance measure between center points faces the same difficulty

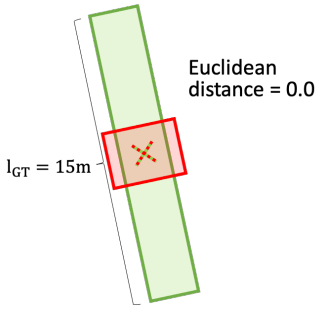


Figure 3.5: A perfect match between a detection (red) and a reference object (green) according to the Euclidean distance of the center points. The Euclidean distance does not take the object size into account.

that the side of a detected object which is close to the ego-vehicle may not be properly detected even though the center points of detection and reference are in good agreement. Therefore, the Euclidean distance of the center points will, like the IoU, not correlate well with the defined crashes either. The Euclidean distance is less expensive in comparison to the IoU in terms of computational requirements.

Support distance error

The Support Distance Error (SDE) accounts for the object contours that are closest relative to the ego vehicle [64]. Therefore, [64] describes the association measure to be ego-centric. All before-mentioned association measures are considered to be object-centric as these association measures are independent of the position and orientation in the ego-vehicle's coordinate frame. The support distance (SD) to a detection or reference bounding box corresponds to the distance to the closest point of an object along the x (longitudinal) and y (lateral) coordinates, respectively.

$$SD_i\left(B_C\right) = \min_{\mathbf{p} \in B_C} \left(p_i\right), \quad i = x, y \tag{3.7}$$

$B_C \in \mathbb{R}$ represents the set of points on the contour of a 2D bounding box from the bird's-eye perspective. In this context, p_i represents the coordinate component i of point $\mathbf{p}$, where i corresponds to either the x or y dimension. The SDE corresponds to the difference in the SD for a detection and a reference object along the coordinates x and y from the ego-perspective.

$$SDE_i = SD_i\left(B_{C,ref}\right) - SD_i\left(B_C\right), \quad i = x, y \tag{3.8}$$

The SDE corresponds to the minimum of the absolute value of the lateral and longitudinal SDE [64].

[64] shows that, unlike the IoU, the SDE shows a correlation with the number of hypothetical crashes. The SDE turns out to be on average lower for TP crashes and higher for FP and FN crashes. Therefore, in the case of the SDE, a threshold α would allow a better distinction of the detections into a dangerous and safe object than the IoU, as the IoU may classify an object as TP even though the detection would pretend to avoid a crash with

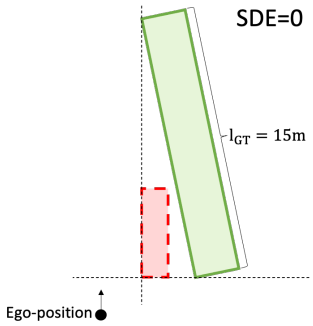


Figure 3.6: A perfect match between a detection (red) and a reference object (green) according to the SDE from [64]. The SDE is equal to zero even though the boundaries of the reference and the detection do not line up. The SDE depends on the position of the ego vehicle. Therefore, the ego vehicle position is shown in the lower left corner.

the object while the ego-vehicle is crashing into the object on its path.
Moreover, the SDE provides a physical distance measure with units in meter which allows a physical interpretation of how much closer/ how much further away the detection is allowed to be from the ego-vehicle in comparison to the distance to the reference object. However, the SDE may also lead to corner cases where the object may not be detected properly. Especially for tilted objects this can be the case. Figure 3.6 demonstrates two bounding boxes that are in perfect agreement according to the SDE as the SDE is equal to zero. As the SDE depends on the ego-vehicles position, the ego-position is shown in the lower left corner while the current driving direction is indicated with an arrow. As discerned from Figure 3.6, it is plausible for the detection and reference to exhibit no overlap, despite the SDE indicating perfect alignment. Consequently, although the correlation with potentially hazardous situations in the immediate surroundings has been shown in [64], the measure may still permit corner cases of utmost relevance for the safety assurance of autonomous vehicles. Consequently, interpreting the results obtained through the SDE may prove challenging, as exemplified in Figure 3.6. The measure may overlook pertinent cases, inadvertently inflating overall performance metrics, despite the actual performance being subpar.
The evaluation of the SDE uses non-expensive operations like subtraction and finding the minimum of two values. It is less computationally expensive compared to the IoU.

Extension to non-rectangular objects

Object detection algorithms usually yield bounding boxes that encircle the objects as closely as possible. However, real-world objects cannot be perfectly described by rectangular boxes. Especially, at the corners, the real-world objects usually differ significantly from the bounding box which can make a crucial difference whether an object is hit or not. Therefore, a representation that covers the actual object dimension may be beneficial. [64] proposes two methods for a more detailed representation of the objects in combination with the SDE. Both methods are based on the raw LiDAR data.
The Convex Visible Contour (CVC) uses the convex hull of all LiDAR points from a

single time frame within the object bounding box after removing the LiDAR points on the ground [64]. In particular the object contours that are in the LiDAR sensor's FOV and are not occluded by other objects are commonly most relevant in terms of crash avoidance. In [64], a better detection performance is observed for nearby objects when utilizing a CVC representation of the objects instead of using a bounding box representation.
The CVC is only a viable option for measures that account for the nearest object's contour rather than a full contour of the objects as opposing sides of objects are occluded from the LiDAR sensors perspective and cannot be detected. Object interpolations in occluded regions are not considered when employing the CVC in a detection evaluation. Consequently, CVC detections are not suitable for an evaluation that is based on an area or volume-based measure, such as the IoU. An evaluation of the nearest object's contour is justifiable by the fact that the distance to the side of the object facing the ego vehicle typically holds greater significance than the precise dimensions of the detected object as emphasized in [64]. When used in conjunction with the SDE, the CVC offers a more accurate measure by acknowledging that objects are not strictly rectangular in shape. Nevertheless, apart from accounting for the non-rectangular forms of real-world objects, the combination of the SDE and CVC may still encounter situations similar to the one depicted in Figure 3.6, thereby limiting the interpretability of the results. Moreover, evaluating the partial point cloud within the bounding box imposes a greater computational burden compared to assessing the IoU. Additionally, in adverse weather conditions, a LiDAR point cloud may not be consistent, which may make an approach that is based on a LiDAR generated CVC become inapplicable.
This argumentation can also be applied to the second approach by [64] of the name Starpoly. Starpoly also takes into account that objects are seldom rectangular and, like CVC, it is LiDAR-based. Starpoly utilizes a star-shaped description of the objects. Analog to CVC, in combination with the SDE the interpretation of the results is limited as the SDE can classify detections as a perfect match between reference and detection when they are not, as demonstrated in Figure 3.6. Furthermore, it increases the difficulty in labeling the data compared to bounding boxes. A more precise contour of the detected objects, however, allows a better interpretation of the scene. Thus, as pointed out by [64], crashes that would be observed with rectangular bounding boxes may indeed be no crashes as the ego vehicle can pass the detected object when taking into account that the object has round corners. Therefore, the introduction of non-rectangular boxes can increase the interpretability of the result for all previously mentioned association measures.

Conclusion about existing association measures

The section provides an overview of association measures that have been utilized in the context of object detection in automated driving. The association measures are analyzed quantitatively concerning (1) their interpretability, (2) their correlation with potentially hazardous situations and (3) in terms of the computational requirements. As the interpretation of an evaluation is based on whether a situation is safe or potentially dangerous based on physical properties of the object states, (1) and (2) are commonly related and

discussed together.
Usually, a better agreement of detection and the reference object for an association is achieved with more restrictive association measures or more restrictive thresholds. More restrictive association measures and thresholds will lead to a decrease in the number of detections that are classified asTP and, simultaneously, to an increase in the number of FN and FP detections. However, a more restrictive association hardly provides better interpretable results as the number of TP detections for such measures does not relate to the physical properties of the vehicle and, thus, its safety. This argument holds for most of the discussed association measures. Except for the SDE, which is proposed by [64], none of the above-mentioned association measures incorporate the safety aspect.
In order to account for possibly hazardous situations [64] introduces the terms object-centric and ego-centric. Ego-centric association measures incorporate the position of objects relative to the ego-vehicle in the evaluation.
The computational requirements may become relevant when dealing with very large datasets where computationally expensive approaches can take infeasible amounts of time. However, to this extent, this has not been a problem in the evaluation of object detection algorithms. Even computationally more expensive algorithms can be applied to the datasets that are available nowadays. In case the amount of data increases to an extent when the computational requirements become relevant, SDE in combination with bounding boxes as proposed by [64] provides a lightweight evaluation.
The computational demands may emerge as a significant factor when confronting vast datasets, wherein computationally intensive association measures could lead to impractical processing durations. Bounding boxes are frequently employed as an object representation in object detection, offering a rudimentary approximation of reality. Despite their simplicity, bounding boxes provide an accessible mathematical description with a limited number of parameters, facilitating computationally efficient evaluation. Conversely, more intricate shapes, though potentially yielding increased accuracy, necessitate greater computational resources for evaluation.
Up to this point, such concerns have not been a challenge in the evaluation of object detection algorithms, even with more intricate shapes requiring higher computational complexity. However, in case the amount of data escalates to levels where computational demands become more relevant, the SDE in conjunction with bounding boxes, as proposed by [64], presents an effective evaluation solution.

3.2 Grid-based association measure

In the evaluation of object detection, not detecting objects when none are present is often overlooked due to data imbalance. Data imbalance refers to the uneven distribution of instances across different classes or regions, in this case, between regions containing objects and those without them. In object detection, addressing this imbalance is crucial, as an object detection model may otherwise predict no objects by default, as regions with objects are usually less frequent.

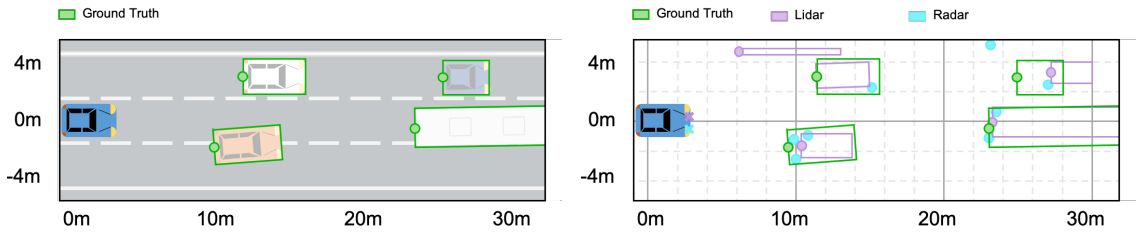
For the reliability analysis, however, it is essential to consider regions correctly identified as unoccupied when no objects are present, which constitute the TN cases. The model outlined in section 3 for determining sensor reliabilities necessitates the definition of TN cases for proper application. Furthermore, this model is dependent on an association of multiple detections, rather than the association of a detection with a reference object. In the absence of a reference, the challenge arises in determining which sensor to trust when faced with disparate detections resulting from varying physical principles and uncertainties in sensor measurements and processing procedures.
The association measures from section 3 do neither define TN cases nor do these association measures deal with sensor-specific sensor properties to perform an association of object data from multiple sensors. Thus, in the context of assessing sensors' object-detection-based on a redundant sensor system, a different association measure is required. A grid-based association offers an intuitive approach as starting point for the evaluation of large-scale datasets using the model from section 2.5.2.
Subdividing the FOV into a discrete grid for the association is briefly described in [8, 70]. One can use the grid-based association to apply the model from section 2.5.2. The model from section 2.5.2 requires a definition of TN in order to evaluate the PFA. In the following, we describe the grid-based association in more detail.

3.2.1 Association measure

The presented association measure is a discrete measure that assumes a value of one for detections from different sensors that are in the same region, or zero otherwise. The measure relies on the discretization of space by applying a grid on the bird's eye view. The measure only uses the position of a reference point from surrounding objects. Additional parameters like the size of the object, its velocity and its rotation relative to the ego vehicle are not considered. This work uses the center point at the rear of the objects as reference point.
Objects from different sensors are considered to be the same in case their reference points are within the same grid cell. Otherwise, the objects are considered to be different objects. The grid size is a free parameter.
Figure 3.7 demonstrates the association on an artificially generated frame. In Figure 3.7 the cell size is to 2 m x 2 m and is indicated by the grey dashed lines. Figure 3.7 (a) shows the reference objects by the green bounding boxes. The dots at the bounding boxes indicate the rear center of the objects. The reference point of the ego vehicle is at position $(0\,\mathrm{m}, 0\,\mathrm{m})$. Figure 3.7 (b) demonstrates the reference objects together with the detections from different sensors.

3.2.2 Results and discussion

Figure 3.8 shows the resulting binary data that corresponds to either an object being present or not. Figure 3.8 (a) shows the binary data of the reference truth. There are four objects present in four different cells of the grid indicated by a small rectangle within each cell that contains the reference point of an object. Figure 3.8 (b) shows the result for all

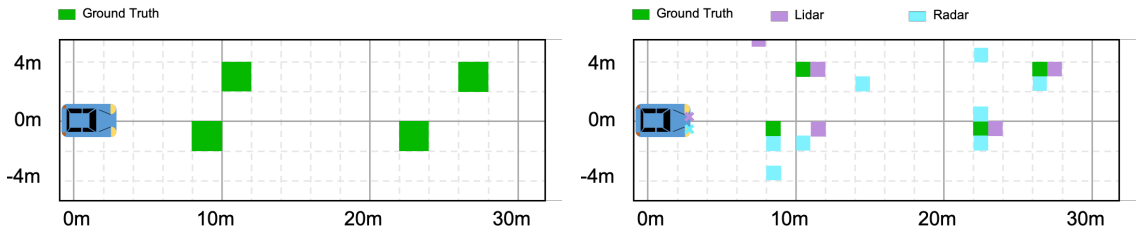


(a) (b)

Figure 3.7: Illustration of the grid-based association. A grid is used to subdivide the environment from the bird's eye perspective. (a) shows the reference truth objects, (b) shows the sensor data in addition.

sensors for the example frame. A comparison of the binary detections and reference data allows a classification of detections in any of the four cases of the confusion matrix from Table 2.1, namely TP, FP, TN and FN. In comparison to the association approaches in section 3.1, the grid-based association also defines TN cases which is necessary to apply the model from section 2.5.2.

While the approach is straightforward to implement it has major drawbacks. Firstly, it only considers the position of the reference point of the surrounding vehicles neglecting the size of these vehicles. Consequently, a large object may be associated with a detection of a much smaller object. Moreover, the detection may have an entirely different orientation than the actual object. Secondly, the acceptable tolerance for an object's reference point is defined by the rectangular cell and varies in different directions due to the squared-shaped grid cells. In the worst-case scenario, a detection's rear center, chosen as the reference point here, may be up to 2.83 m apart from an object it is associated with while still being classified as a correct detection. This length is determined by the diagonal of a 2 m x 2 m cell. Such a tolerance might be sufficient in the far distance. However, it is certainly not sufficient in the near field and adds up to the error introduced in (1). Thirdly, the approach classifies detections into a FP and a FN when reference object and detection are separated by the cell border, even if the reference truth object and a detection are close together. Lastly, the size of the grid cells is a free parameter without any particular rationale for a specific value. A cell size that is too large increases the probability of multiple objects gathering within the same cell. However, with decreasing cell sizes the number of TNs increases which can modify the PFA arbitrarily. We propose, therefore, to choose a cell size that is in the order of the size of the objects of an investigated class. Objects of a certain class have usually roughly the same size. Studies in this work mostly focus on cars only, disregarding other object classes. We chose 2 m x 2 m as the reference points, usually the rear center points, of neighboring cars get hardly closer than 2.83 m while the size of cars is in the same order of magnitude. We are aware that cars are usually larger than 2 m x 2 m, but a more extensive grid cell size would exacerbate the issue of multiple cars congregating in the same cell. However, the chosen 2 m x 2 m cell size remains somewhat arbitrary; some might argue that a 1.5 m x 1.5 cell size would be more appropriate to prevent multiple cars from occupying the same grid cell.



(a) (b)

Figure 3.8: Binary data obtained from the grid-based association for the frame of Figure 3.7. (a) shows the binary reference truth data and (b) shows the sensor data in addition.

From the drawbacks of the measure, we conclude that the resulting classifications may not correlate well with potentially dangerous situations when using the grid-based association measure. Subsequently, the resulting number of detection errors does not allow a proper interpretation as the detection errors can be benign or malign which cannot be derived from the number of errors. However, the measure is fast and easy to apply on a large dataset.

3.3 Trajectory-clustering-based association measure

In the preceding section, we presented an association measure that is based on a grid to associate detections from multiple sensors, even in the absence of a reference truth. While the measure is easy to implement it comes with major drawbacks, leading to a weak or no correlation between the number of errors and potentially hazardous situations. One issue is the potential lack of association between slightly shifted detections at grid cell borders. Conversely, depending on the grid cell size the accepted tolerance for objects within the same grid cell can be considerably large. Moreover, sensor-specific short-term errors, which could be eliminated by employing a Multi-Object-Tracking (MOT) algorithm, contribute to the overall error count. For instance, short-term ghost objects can appear in object detections that are based on LiDAR recordings during rainy weather. These ghost objects are often random occurrences in individual time frames. The subsequent association measure addresses these limitations by introducing an association measure that is based on the Euclidean distance and is employing a MOT approach. Nonetheless, a grid is still employed to define the TN cases, a requisite step for classifying the associated objects into either of the four cases of the confusion matrix. The measure is also presented in [57].

3.3.1 Association measure

The association measure functions in both spatial and temporal dimensions, offering a more refined association approach in contrast to the measures previously examined in this work. A MOT algorithm associates the object detections in time for each sensor by evaluating their trajectories. Subsequently, the trajectories of objects from different sensors are compared using a combination of the Euclidean distance for spatial aspects and the Fréchet distance for temporal aspects, instead of relying solely on the position of the actual

measurements.
The procedure can be subdivided into three key steps, aiming to provide a robust and comprehensive evaluation of object detection performance: First, the objects detected in an individual frame are tracked over time. Second, the obtained trajectories for all objects and sensors are clustered between different sensors and the reference truth. Third, a grid is utilized to obtain binary data from the object data either saying an object is present or not. While the binary data is necessary for the reliability analysis by [16], the association itself does not rely on binary data.
Figure 3.9, which we present in our publications [71] and [57], summarizes the details of the procedure. Figure 3.9 (a) visualizes the three subsequent steps of the pipeline in a flowchart. The pipeline takes object lists obtained from an object detection algorithm as input and outputs binary data for a subsequent reliability analysis. The association between the sensor detections and the reference truth is performed by the MOT procedure and the trajectory clustering as indicated by the red box in Figure 3.9 (a). The binarization represents an additional step necessary for the reliability analysis.
The pipeline employs the MOT algorithm from [50], which incorporates a Kalman filter for object tracking. As a well-studied approach, the Kalman filter facilitates the interpretation of the results, whereas MOT algorithms based on neural networks often exhibit a deficiency in interpretability.
The MOT algorithm is used to avoid short-time artifacts like sudden misdetections or short-time occlusions. Short time FPs might for example be detected by an object detection algorithm that is based on LiDAR when it rains. Using the MOT algorithm, one can eliminate these errors for a subsequent reliability analysis.
Figure 3.9 (b) visualizes the MOT algorithm from [50]. It starts by assigning an object ID to every object (bottom left box in Figure 3.9 (b)). Subsequently, a Kalman filter prediction is performed for every object. The subsequent step associates predictions and measurements using the well-established and frequently used IoU, as outlined in section 3.1.2, and the Hungarian algorithm [55, 67, 72]. The association between detections and predictions leads to three possible outcomes: Some predictions coincide with some of the detections, some predictions have no corresponding detection and some detections may not have been predicted by the Kalman filter as the object may not have been present in the previous frame. Whether prediction and detection coincide is defined by a minimum IoU threshold. The Hungarian algorithm optimizes overall associations of one frame in order to avoid that multiple detections are associated with the same reference truth object. A list stores predictions that are not associated with a detection for two more time-frames. Furthermore, the algorithm only considers objects as being present if they are detected for at least three subsequent time frames.
After tracking the object detections over time, the pipeline continues by clustering the obtained trajectories. The pipeline applies a variation of Density-Based Spatial Clustering of Application with Noise (DBSCAN) for clustering the trajectories and utilizes the Fréchet distance with threshold of 1 m [73, 74]. Trajectories from different sensors whose Fréchet distance is smaller than the threshold are clustered together and are assumed to relate to

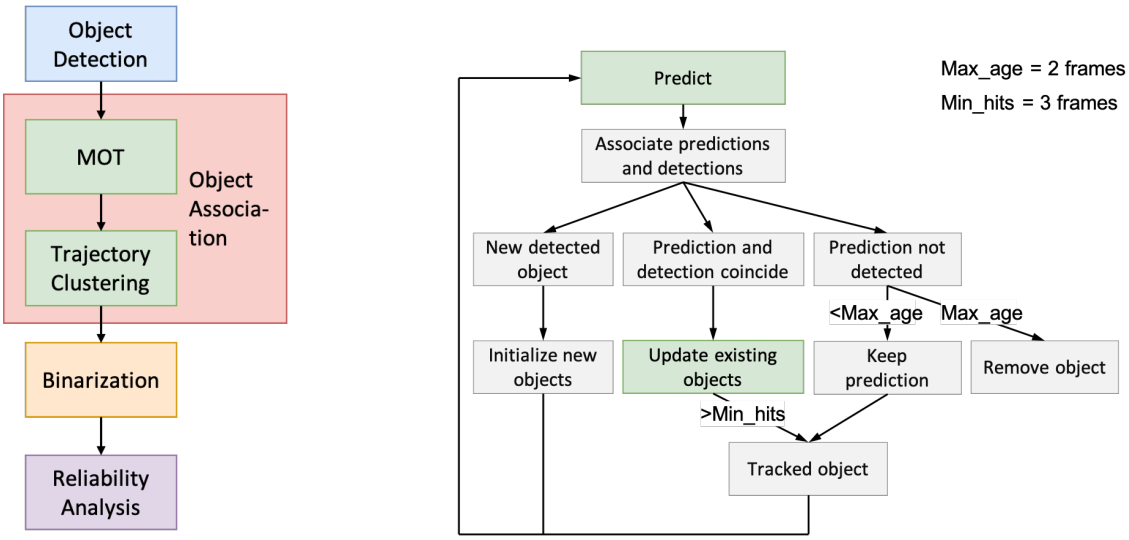
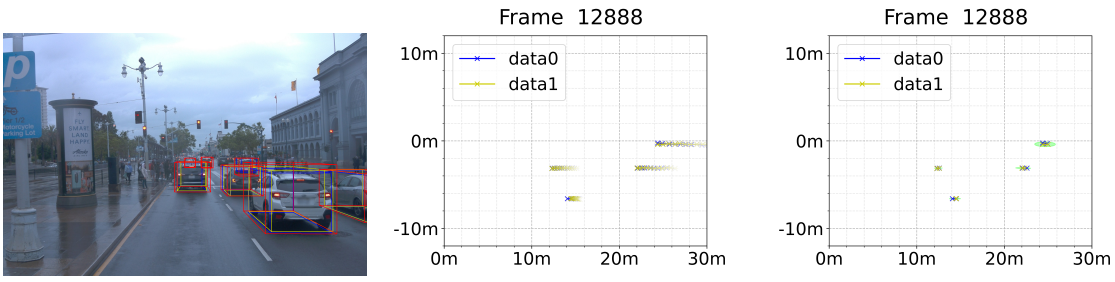


Figure 3.9: (a) Pipeline for associating and analyzing object data obtained from different sensors based on trajectory generation using the MOT algorithm from [50] and a DBSCAN-based clustering. (b) demonstrates the MOT algorithm by [50] in a flowchart. Figures are taken from our publication [71] and [57].

the same object. In this manner, the algorithm can be concurrently applied to multiple sensors and the reference, integrating their data for the proceeding analysis.
Finally, the data is put into a binary format. The number of TP, FP and FN are known from the clustering. Similar to the approach in section 3.2, we suggest that the grid cell size should be roughly determined by the minimum distance between two objects of the same class. For cars, we recommend using a grid cell size of 2 m x 2 m at the largest.

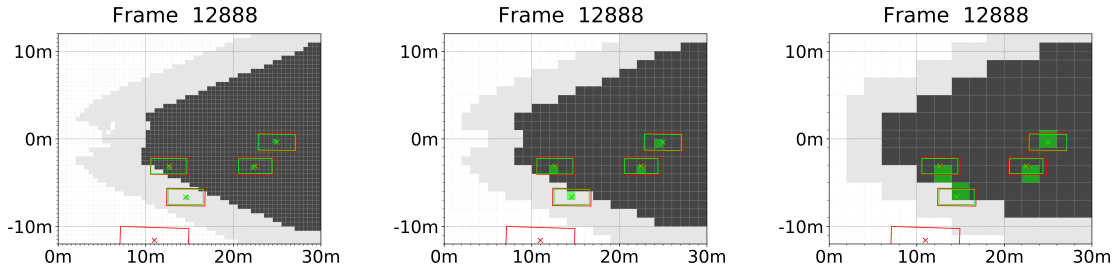
3.3.2 Results and discussion

Figure 3.10 (c) demonstrates the output of the measure for a sequence of the Waymo dataset [1]. Multiple sensors are obtained by sub-diving the LiDAR data of the Waymo dataset and applying an object detection algorithm on each LiDAR data subset. Figure 3.10 (a) shows the camera image of the current frame together with the projected bounding boxes of the reference and the object detection algorithm operating on two individual subsets of the LiDAR data. Figure 3.10 (b) demonstrates the trajectories obtained from two LiDAR data subsets over the past ten frames. The trajectories are relative to the ego vehicle coordinate frame. The reference point of the ego vehicle is positioned at $(0\,\mathrm{m}, 0\,\mathrm{m})$. Figure 3.10 (c) shows the positions of the objects for the current and the previous frame. The 3σ deviation of the clustered data is shown by the green ellipses that highlight a cluster.
Figure 3.11 demonstrates the resulting binary data for the specific frame of the scene from Figure 3.10 for different grid cell sizes of $0.5\,\mathrm{m} \times 0.5\,\mathrm{m}$, $1\,\mathrm{m} \times 1\,\mathrm{m}$ and $2\,\mathrm{m} \times 2\,\mathrm{m}$. The clustered data is demonstrated by the green rectangular boxes while the red boxes indicate the reference object data. The crosses indicate the center points of the bounding boxes. The bounding boxes of the clustered data is obtained by taking the mean of all values from



(a) (b) (c)

Figure 3.10: Visualization of the pipeline from Figure 3.9 for one frame of the Waymo dataset [1]: (a) visualizes the object data by projected bounding boxes on the camera image while the reference truth bounding boxes are shown in red and the detected bounding boxes in blue and yellow; (b) shows the evaluated trajectories of the detected objects; (c) indicates the clustered objects by green ellipse. The figure is taken from our publication [57] and modified.



(a) (b) (c)

Figure 3.11: Representation binary data obtained from object data of multiple sensors. The grid size for the discretization in space is a free parameter and can be chosen differently. Here, Figure (a) is based on a grid size of $0.5\,\mathrm{m} \times 0.5\,\mathrm{m}$, Figure (b) is based on a grid size of $1\,\mathrm{m} \times 1\,\mathrm{m}$ and Figure (c) is based on a grid size of $2\,\mathrm{m} \times 2\,\mathrm{m}$. As cars get hardly closer than $\sqrt{2} \cdot 2\,\mathrm{m}$, a grid size of $2\,\mathrm{m} \times 2\,\mathrm{m}$ may sufficiently represent the data. (c) is taken from our publication [57].

all sensors that participated in the cluster.

In comparison to the grid-based association, this approach does not encounter the problem that a close detection might be classified as FP and FN because the detection and the reference objects are in neighboring grid cells. Whether two trajectories contribute to the same cluster is defined by the chosen threshold for the Fréchet distance between the trajectories rather than the grid cell. The grid is only utilized to obtain a binary representation of the data including the TN cases.

By using the Fréchet distance for associating trajectories, the chronological order of the detections is not taken into account. The applied algorithm is derived from general trajectory clustering where no chronological order of the trajectories exists. A better approach might be to use the mean distance or the max distance obtained from the distances between the objects at each point in time.

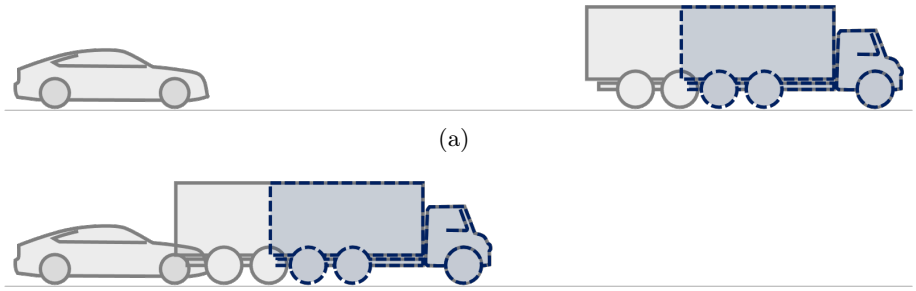
The grid cell size remains a free parameter with limited interpretability, as previously discussed in section 3.2. If not otherwise stated, all subsequent evaluations based on this

association approach use a grid cell size of 2 m x 2 m, which is in the order of the size of cars, here the investigated class of objects. Although in this approach the grid size does not affect the number of detections that participate to one cluster, multiple clusters can accumulate within the same grid cell in case of a too large cell size. In this case, the counted number of detections and reference objects might vary with the grid size, leading to different numbers of TP, FP and FN detections. However, for grid cell sizes small enough the number of TP, FP and FN detections stay the same as these are defined by the clusters which do not depend on the cell size. However, with decreasing the grid cell size leads to an increasing number of TN cases, which arbitrarily alters the probability of false alarm.
Similar to the association measure from section 3.2, a drawback of this association measure is that only the center points of the objects are considered for the association. Like in the measure from section 3.2 other parameters like the size and rotation of the objects are not considered. Thus, a detection could be associated with a reference object even though it is rotated by 90° or a detection could be of a different size compared to the reference object which can make a crucial difference in the path planning of an automated vehicle.
In addition, a deviation up to the defined threshold for clusters based on the Fréchet distance can occur. The deviation is independent of the distance. Here, the chosen threshold was set to 1 m. A deviation of 1 m might be too restrictive for faraway objects while it can be too loose for nearby objects.
While retaining some limitations of the purely grid-based approach, the trajectory-based association measure effectively addresses evaluation errors arising at grid cell boundaries and short-term errors. Another advantage of the trajectory-based association measure is the consideration of temporarily occluded objects due to continued tracking, ensuring that such objects, which are often present in the reference data, do not contribute to FN errors. Consequently, the trajectory clustering-based association is superior to the grid-based association.

3.4 Distance-weighted association using threshold-based measures

Without the need for the exact location and distance of the other participants in the traffic scenario, humans are also eligible to steer a vehicle safely. Human driving is often taken as a reference in safety concerns for automated driving [4, 8, 75, 76]. Human estimation of distances to surrounding objects that are further away is prone to misestimation. Consider an example in which the ego vehicle approaches the tail end of a traffic jam on the highway assuming a moderate speed of 120 $\mathrm{km\,h^{-1}}$ for the ego vehicle. A human will most likely not evaluate whether the last car or truck of the traffic jam is 150 m in front of the ego vehicle or 140 m in front of the ego vehicle. Nonetheless, a human will start to decelerate the vehicle. In case the last car/truck at the end of the traffic jam is right in front of the ego vehicle, however, a tolerance of 10 m is not acceptable. One may conclude that the human precision in distance estimation to surrounding objects is partially dependent on

the distance to these objects. In this context, the required precision for object detection algorithms could be related to distance. Just as with human drivers, pinpoint accuracy may not be essential for safely navigating a vehicle on public roads, especially at large distances to another object.
The correlation between the precision of a detection with its distance from the ego position can be assumed for most ego centric sensors. Such sensors usually have an angular resolution. As an example, consider a LiDAR sensor: an object of the same size is hit by more laser beams of the LiDAR if it is closer to the ego vehicle. Thus, a nearby detection is represented by more measurements in comparison to a measurement at faraway distances. Besides the reduced spatial resolution of ego centric sensors, a decrease in the signal intensity of individual measurements is expected with increasing distance to objects for active sensors [36]. Considering the LiDAR again, the laser beam is not perfectly parallel and, therefore, diverges over distance. Thus, the further an object is away from the LiDAR, the less intense is the returning signal.
To the best of the author's knowledge, to this date, no properly justifiable performance requirements for environment perception sensors and their object perception exist for the use in automated vehicles. A major focus in the development of object detection algorithms for the environment perception of automated vehicles has been the improvement of these algorithms. This is achieved by aiming for higher numbers of TP detections with association techniques described in section 3.1, 3.2 and 3.3. However, hardly any research describing perception specifications that may be sufficient for automated driving exists.
One may achieve requirements sufficient for automated driving by drawing parallels to human perception and adjusting them on a distance-dependent basis. Consequently, for an interpretable and justifiable evaluation one may need to drop current association approaches that classify detections independent of their distance into TP and TN as well as FP and FN, meaning correctly or falsely identified. Such requirements may not just allow an interpretation of the results by a comparison with the human perception but at the same time, they may be better achievable due to the described sensor properties. To the best of the authors' knowledge, a distance-dependent association in the field of automated driving perception does not yet exist.
Concerning requirements, no approaches have been found by the authors. Distance dependency, however, found its way directly or indirectly into the performance evaluation of object detection performance. [65] indirectly incorporates the distance to objects by their size in images is, using equation 3.2. Furthermore, an approach that considers the distance in the evaluation of the perception performance of automated vehicles exists [64]. However, this approach weights the detections depending on their distance to the ego vehicle after the association between the detections and the reference.
Many measures used for the association between detections and reference are derived from image processing. One of the most commonly used association measures in the perception evaluation of automated vehicles is IoU as described in section 3.1 [51, 62]. Figure 3.12 provides a one-dimensional illustration for two correct detections according to the IoU. For the illustration assumes an IoU ≥ 0.7 [49]. Detections with an IoU greater or equal



(a)

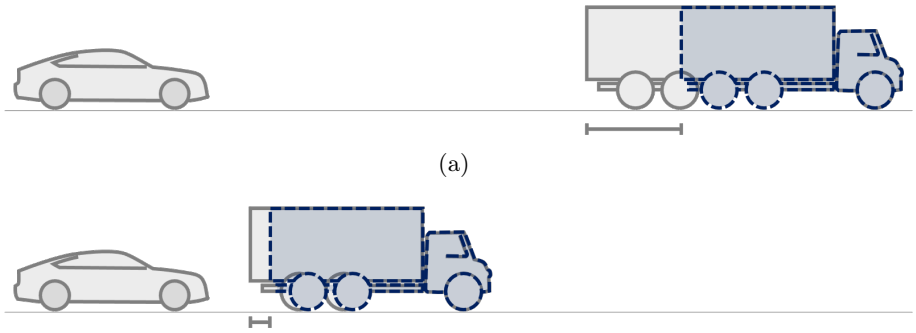
(b)

Figure 3.12: One-dimensional illustration of a problem with the most common approach for associating object detection and reference truth, which consist of the IoU as association measure and a predefined threshold, usually 0.5 or 0.7 [15, 49–53]. The detection is illustrated in blue while the true scene is shown in grey, on the left-hand side is the ego vehicle and on the right-hand side is a truck which is an object for detection. A fraction of 0.7 of the reference truck and the detected truck are in agreement. This corresponds to an IoU value of 0.7. If the truck is at far distances, the IoU of 0.7 might be sufficient as demonstrated in Figure (a). However, if the object is close, a detection with an IoU value of 0.7 might not be sufficient as demonstrated in Figure (b).

0.7 with a corresponding reference object are considered as correctly detected. Figure 3.12 highlights the detections in blue while the reference truth is shown in grey. In both Figure 3.12 (a) and (b), the IoU value is equal to 0.7. For large objects, however, the deviation in the detection of the object's frame can be large even if the requirement for the IoU is fulfilled. In the case of a truck of a length of 10 m, the deviation can be up to 3 m which can lead to an accident as illustrated in Figure 3.12 (b). Thus, for the reliability analysis of automated vehicles, the IoU might be too conservative in far distances on the one hand but too speculative in the near field of the ego vehicle on the other hand.
For the evaluation of the object detections in automated driving one may, therefore, need to find a set of rules that classifies the detections into correct or false in another way, e.g., by incorporating the distance into the measure.

3.4.1 Concept behind a distance-weighted association

As illustrated with a truck at different distances, a distance-dependent association based on a distance-dependent acceptable tolerance can be beneficial in the evaluation of the environment perception from an ego centric perspective. Detections that deviate more than the acceptable distance-dependent tolerance from a reference object are classified as false detections, objects that lie within the acceptable tolerance are classified as correct detections. The evaluation only focuses on the edges of objects which are in sight from the ego perspective. Therefore, only the objects that are closest to the ego vehicle in any radial direction covered by the sensor are considered for the evaluation. We propose to scale the allowed tolerance linearly with distance. This fulfills the requirement that the acceptable tolerance should be close to zero for nearby objects. The acceptable tolerance ΔR for an obstacle detection at distance r_{2D} to be classified as a TP is then defined by the distance $r_{2D,ref}$ of the corresponding reference obstacle and a defined error constant ΔR.



(a)

(b)

Figure 3.13: One-dimensional illustration of a distance-dependent allowed deviation. Most relevant in the detection of other objects are the object faces closest to the ego vehicle. For the evaluation of the perception, it might be sufficient to allow large deviations for detections at far distances. However, unlike demonstrated in Figure 3.12 nearby detections are only allowed to have small deviations from the reference object in order to be counted as a correct detection.

$$\Delta R = \alpha_{dd} \cdot r_{2D,ref} \tag{3.9}$$

Here, α_{dd} is a relative error constant that defines the allowed deviation between the detection r_{2D} and the reference $r_{2D,ref}$. Thus, a detection of an object that is located at a distance $r_{2D,ref}$ from the ego vehicle is classified as a TP detection in case it is lying at a distance $r_{2D} \in [r_{2D,ref}\,(1-\alpha_{dd})\,, r_{2D,ref}\,(1+\alpha_{dd})]$.
Figure 3.13 illustrates the approach in 1D. At small distances, the error in the distance estimation should be close to zero as shown in Figure 3.13 (a). For larger distances, a larger tolerance is sufficient as shown in 3.13 (b). This way we avoid adding events as shown in Figure 3.12 (b) to the number of correct detections in the reliability evaluation.
The unit-less error constant α_{dd} is a free parameter. Similar to the selection of threshold values for numerous in the other association measures, a rigorous theoretical derivation for the determination of the error constant α_{dd} is missing here. As described above, one may consider a tolerance of 10 m at a distance of 150 m as being sufficient to navigate a vehicle safely. This corresponds to an error constant of $\alpha_{dd} = 6.67 \times 10^{-2}$. For the illustrations in the following, we utilize a slightly smaller error constant of $\alpha_{dd} = 5 \times 10^{-2}$.
The 1D illustration of Figure 3.13 lacks the second dimension of the driving plane of the vehicle. As most sensors are recording with a certain angular resolution, we propose to add a constant angular error additional to the relative radial error. In the following calculations, we allow an angular deviation of $\Delta\varphi_{dev} = \pm 1°$.
Figures for the demonstration of the measure in this section are based on LiDAR scanner data. The assumption of a certain angular resolution is satisfied by the LiDAR scanner. The study uses a scene of the nuScenes dataset for the illustration [27]. The front and rear camera images of the scene are shown in Figure 2.1.
We introduce two approaches that account for a distance-dependent deviation and a constant angular deviation. The first approach limits the allowed deviation to the maximum angular deviation. The second approach is less restrictive at the corners of bounding boxes

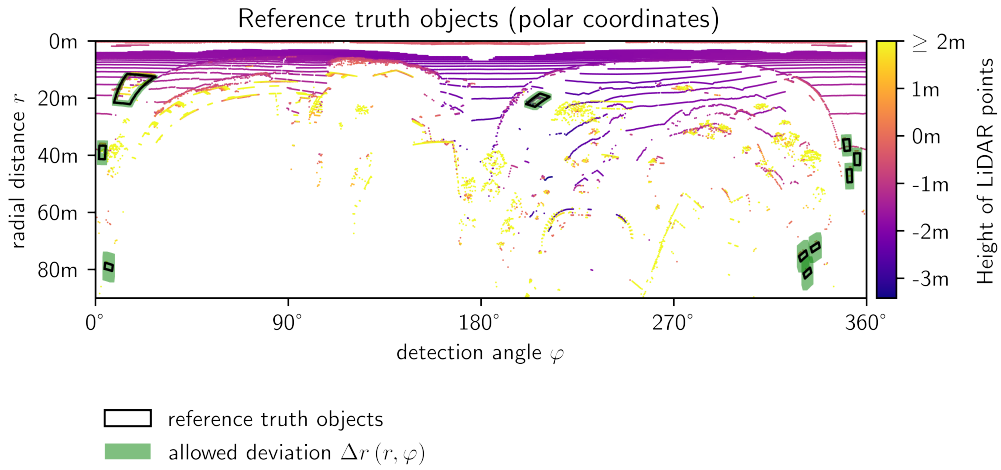


Figure 3.14: Illustration of the determined allowed deviation frames for an individual scene of the nuScenes dataset in polar coordinates. The scatter plot indicates the LiDAR recordings, indicating the height of the recordings with the colormap. The reference bounding boxes are shown in black. The green frames indicate the allowed deviation of detections. Detected bounding boxes that fall within the green deviation frame can be considered as TP detections. The allowed deviation is determined under the assumption that the deviation increases linearly with the distance to the detections in combination with a constant angular deviation. The front and rear camera images of the scene are shown in Figure 2.1.

as bounding boxes are a rough approximation of objects. For many objects, the deviation of bounding boxes is largest at the box corners if the objects do not entirely fill the bounding box.

3.4.2 Threshold-based association measure in polar coordinates

This association method relies on a transformation of the bounding box into polar coordinates. Both, angular and radial deviation, are added using a polar representation of the scene.

Figure 3.14 shows one scene of the nuScenes dataset in polar coordinates. The LiDAR data is added as a scatter plot. The height of the LiDAR recordings are indicated by the colormap. The reference bounding boxes are shown in black. Due to the transformation into polar coordinates, the bounding boxes are distorted. The allowed deviation of the bounding boxes is illustrated by the green frames. The detection bounding boxes are supposed to lie within the green areas in order to be accounted as TP detection.

The method first uses image processing for the transformation into polar coordinates and the consideration of the angular deviation. One starts by transforming the bounding boxes into binary images. The binary images are obtained by limiting the considered region around the car and discretizing it. Here, the region along the x and the y axis was limited to the interval $[-90\,\mathrm{m}, 90\,\mathrm{m}]$ with a discretization of $\Delta x = \Delta y = 20\,\mathrm{cm}$. One can then transfer the image from the Cartesian coordinate system to polar coordinates. One can add/subtract the angular deviation by applying a dilation/erosion with a kernel that is only one pixel in width along the radial dimension and N pixel in width along the angular

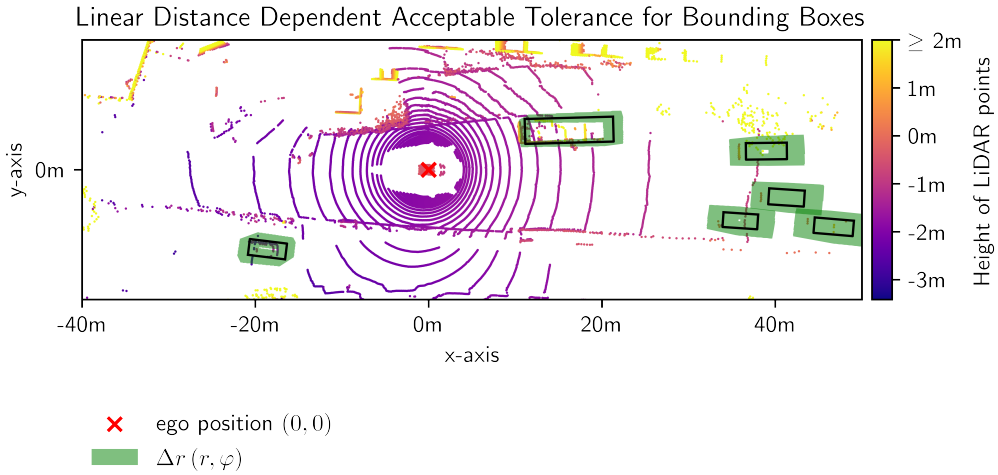


Figure 3.15: Illustration of the approach for determining the allowed deviation. The allowed deviation is indicated by the green frames around the reference bounding boxes which are shown in black. The scatter plot shows the LiDAR data while the height of the LiDAR recordings are visualized by the colormap.

dimension while, here, N is defined by the resolution of the angular axis and the allow deviation for the specific sensor. For this analysis, the resolution of the angular axis was set to $\Delta\varphi_{img} = 0.01°$ and the deviation of the LiDAR sensor was set to 1°. This resulted in a width of the dilation/erosion kernel of $N = 201$.
In a second step, the method derives the lines of the bounding boxes in polar coordinates from the polar images and adds the radial deviation. When determining the lines of the bounding box, one can separate between the lines outside and inside the FOV of the sensor. For the bounding boxes obtained from dilation of the initial bounding boxes, the lines inside the FOV are multiplied by $(1 - \alpha_{dd})$ and the lines outside the FOV are multiplied by $(1 + \alpha_{dd})$. This results in the maximum outline within which the approach requires a TP bounding box to be. For the bounding boxes obtained from dilation of the initial bounding boxes, the opposite is applied: the lines inside the FOV are multiplied by $(1 + \alpha_{dd})$ and the lines outside the FOV are multiplied by $(1 - \alpha_{dd})$. This provides a minimum outline. A TP detection bounding box is required to be larger than the minimum outline.

Results

Figure 3.15 illustrates the allowed deviation from Figure 3.14 in Cartesian coordinates. The position of the sensor is indicated by the red cross. In Figure 3.15 one sees that the corners of the green deviation frame appear to be cut off. This is because we assume a constant deviation in the angle. Thus, the corner of a bounding box can at most deviate by the maximum angular deviation of $\Delta\varphi_{dev} = 1°$. The allowed deviation in the distance to the other objects, however, can be larger which can be seen be the increasing frame width for objects further away.
Cutting off the edges of the deviation frames is caused by the rigid definition of the angular

deviation and is, thus, one feature of this definition. This is a legitimate approximation if the object corners would be in perfect agreement with the corner of the bounding box. As an example, consider a camera image with a pedestrian in the image. A human would be able to draw the edges of the visible part of the pedestrian within plus or minus one pixel. However, when drawing a bounding box around the pedestrian the task requires more assumptions: For example, if the pedestrian stretches out his arms, do the entire arms have to be within the bounding box? In this case, the entire bounding box might be larger than twice the size of the pedestrian herself. In case the pedestrian is partly occluded the definition of the bounding box becomes even less intuitive. Some studies restrict the bounding box on the visual parts of the object only [14]. This argumentation demonstrates that the definition of bounding boxes itself is not trivial. Also, more rigid bodies like cars and trucks are not rectangular. Thus, bounding boxes are just a rough approximation also for these objects. As a result, one might want to allow a larger deviation at the corners of bounding boxes instead of the cut off frames due to the small angular deviation. The small deviation might still be valid for the type of sensor, however, not for the bounding box approximation.
The following section presents an association measure that is based on a deviation frame that is not as restrictive at the bounding box corners.

3.4.3 Threshold-based association measure with interpolation at bounding box corners

Likewise, to the previous association measure, this association measure is based on distance-dependent deviation within which the detected bounding boxes need to be. The figures illustrate the allowed deviations by green frames around the reference bounding boxes. The implementation of the association starts with extending the lines of the bounding boxes to infinity as illustrated in Figure 3.16. Likewise, to the previous association measure, the calculations are performed in polar coordinates.
Figure 3.17 shows the extended lines of the bounding box in polar coordinates for the truck in the scene. One can evaluate the radial distance r in dependence of the angle φ. The function incorporates the tangent of the angle φ, which is periodic every 180°. The periodicity of the tangent leads to additional lines that are mirrored around the ego position. Excluding these lines has to be accounted for in the evaluation which can be obtained by checking the y-intercept and the slope of every line. For lines where the y-intercept and slope are both positive or both negative, one needs to consider a different interval in φ in comparison to lines where either the y-intercept or slope are negative. In Figure 3.17 this is already considered.
In polar coordinates, one can add the allowed deviation to the lines. The constant angular deviation is obtained by shifting the lines along the angular dimension by the constant deviation of $\pm 1°$. The radial deviation is added by multiplying the lines with $(1 \pm \alpha_{dd})$ while α_{dd} represents the error constant.
Figure 3.17 illustrates the minimum and maximum deviation for each of the lines which define the bounding box. From Figure 3.17 one can derive the different widths of the

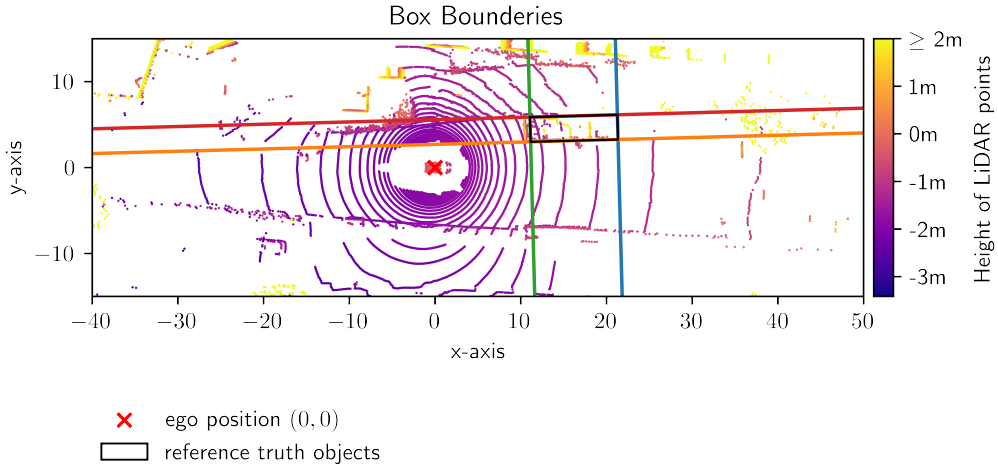


Figure 3.16: First step of the implementation for the distance-dependent association measure that interpolates at bounding boxes corners. In the first step, the lines of the bounding box are extended to infinity. Here, this is exemplarily shown for the truck in the scene that is in front of the ego vehicle.

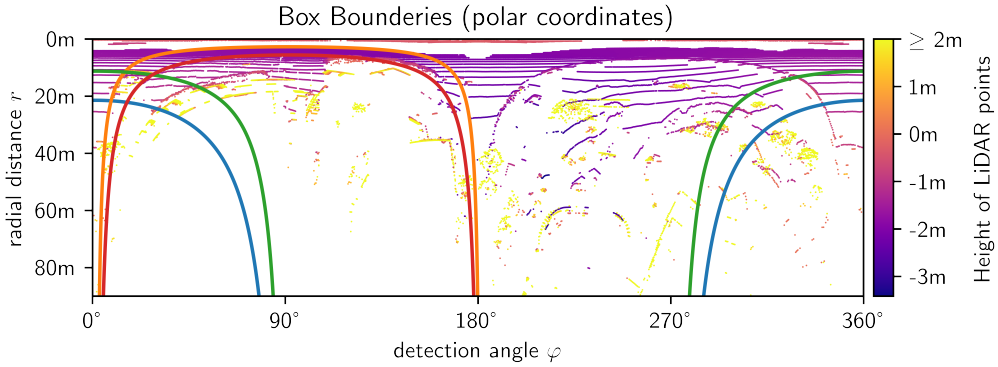
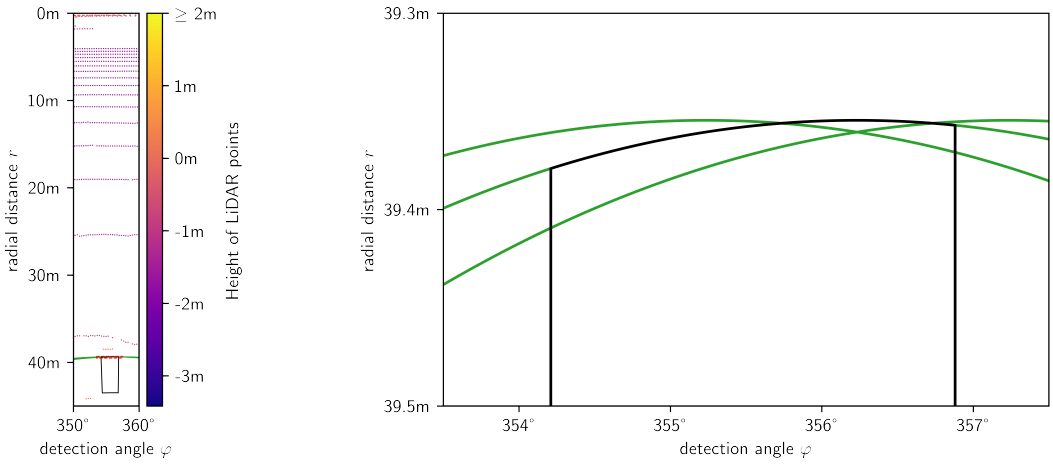


Figure 3.17: The lines of the bounding box as shown in Figure 3.16 transferred to polar coordinates. Additionally, for visualization the LiDAR data is added to the figure.



(a) (b)

Figure 3.18: If only one side of a bounding box faces the ego vehicle, the lines obtained by adding/subtracting the angular deviation intersect. This Figure shows an example within the considered nuScenes scene. (a) shows the bounding box within the LiDAR point cloud. (b) shows a zoomed-in extract of the bounding box side that faces that ego vehicle. The green lines correspond to the bounding box side that faces the ego vehicle and the two lines that are achieved by shifting the line by the angular deviation in either direction. The extract shown in Figure (b) is highlighted in Figure (a) by the red dashed rectangle.

deviation corridors for lines of the object that are further away from the ego vehicle.
In case only a single side of the bounding box is observed from the ego centric perspective, one cannot define the minimum and the maximum deviation by a single line that has been shifted by $\Delta\varphi_{dev}$ or $-\Delta\varphi_{dev}$ as the shifted lines intersect as shown in Figure 3.18. In order to overcome this effect, nine lines are evaluated out of every line. Three lines are obtained by adding the angular deviation $\pm\Delta\varphi_{dev}$ and keeping the initial line itself as shown for the relevant bounding box side for the object in Figure 3.18. For all of these three lines, the procedure is repeated with adding and subtracting the radial deviation. In the end, the maximum and the minimum of the nine lines define the outer bound of the allowed deviation. Figure 3.19 shows the resulting outbounds for the nearby truck in the scene as shown in Figure 3.16.
Figure 3.20 illustrates the lines defining the minimum and maximum deviation in Cartesian coordinates. One obtains a minimum bounding box and a maximum bounding box by truncating the lines at their intersections.

Results

Figure 3.21 shows the resulting deviation frames that are extended at the corners of the bounding boxes. In comparison to the deviation frames from the association measure that is solely based on polar coordinates, the extended frames are less strict at the corners of the bounding boxes. As discussed for the solely polar-coordinate-based association measure, bounding boxes are just a rough approximation of the objects and the interpretation of a perfect bounding box can vary. The deviation at the corners of bounding boxes can,

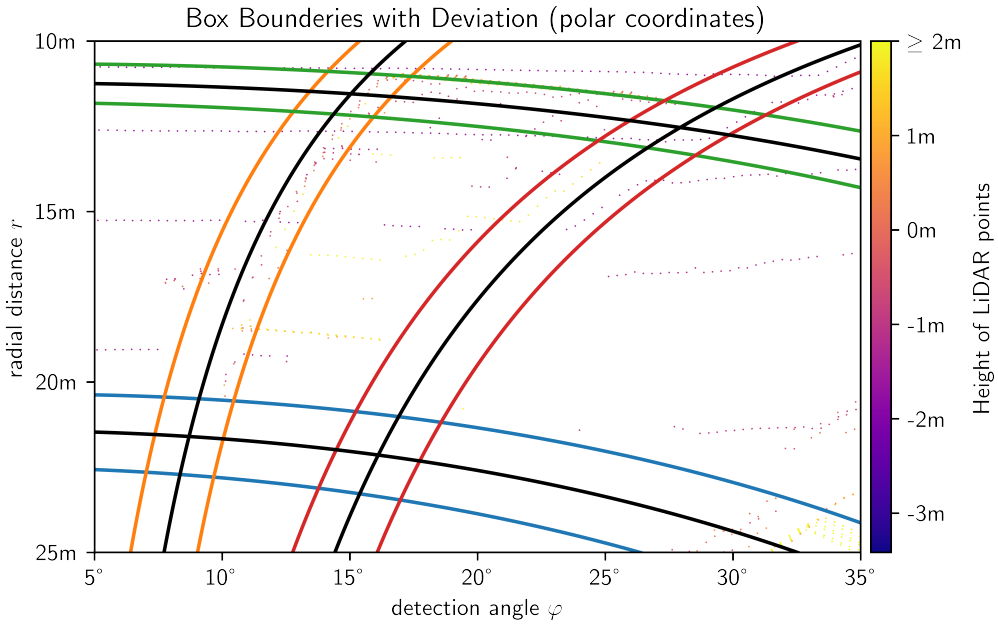


Figure 3.19: The lines of the bounding box after adding the allowed radial and angular deviation. The angular deviation is added by shifting the graph to the left and to the right by the angular deviation of $\Delta\varphi_{dev} = 1°$. The radial deviation is added by multiplying the radial distance from the ego vehicle with $(1 \pm \alpha_{dd})$. The points refer to the measurements by the LiDAR scanner. The height of the measurements is indicated by the color map.

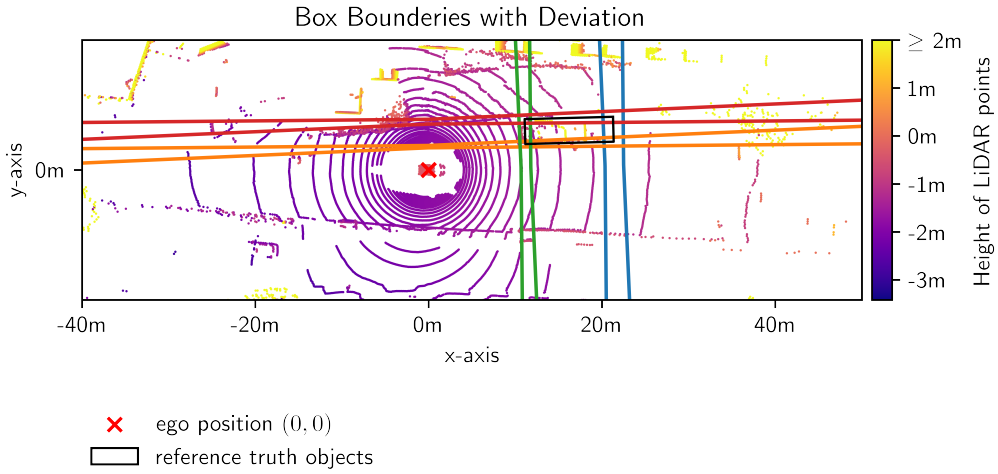


Figure 3.20: The lines of the bounding boxes after adding the allowed radial and angular deviation and the back transformation to polar coordinates. The LiDAR point cloud is added to the figure while the color map indicates the height of the LiDAR measurements.

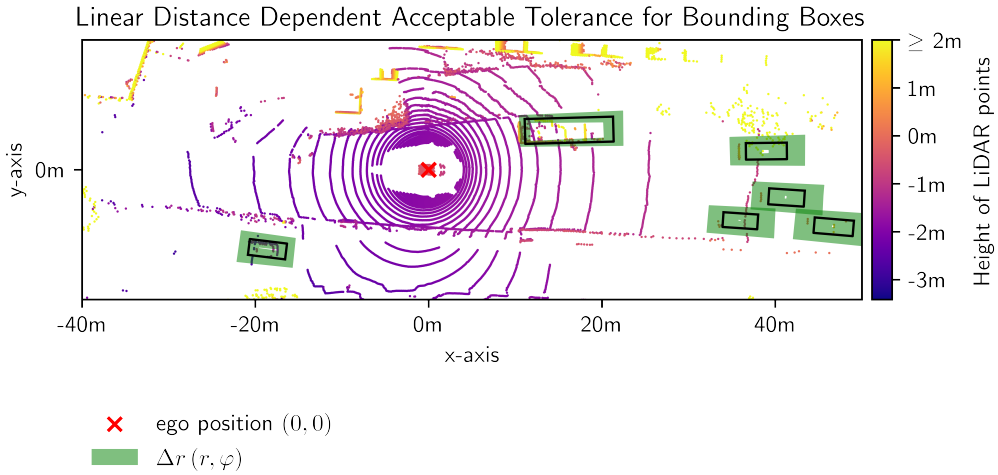


Figure 3.21: The figure shows the resulting frames for the allowed deviation in green around the reference truth bounding boxes which are shown in black. In addition, the LiDAR point cloud is added to the figure while the color coding indicates the height of the LiDAR measurements.

therefore, be indeed larger than the angular deviation of the sensor which is defined by the maximum angular resolution of the sensor. One can account for this larger deviation using interpolation at the corners of the deviation frames. Like the solely polar-coordinate-based association measure, the allowed deviation still varies in distance and the accepted detection deviation of nearby objects reduces to zero the closer surrounding objects get to the ego vehicle.

3.4.4 Results and discussion

Figure 3.15 and Figure 3.21 illustrate the results of the two different threshold-based association measures. Both association measures rely on the assumption of a radial recording pattern of the data. Radial recording is the default case for most ego centric sensors like camera and LiDAR scanner. Most of the calculations are, therefore, performed in polar coordinates.

The two introduced association measures demonstrate a way to classify bounding box detections in TPs or FPs based on the allowed deviation around the reference bounding box. However, there is no clear and unique interpretation of bounding boxes, which are just a rough approximation of real-world objects. One difficulty for a clear definition of bounding boxes is that from the ego centric perspective at most two faces of an object are observable. In some cases, only one face of surrounding objects may be observed, e.g., of the car or a truck in front of the ego vehicle in a traffic jam. Thus, object detection algorithms extrapolate the bounding boxes in regions that are not observable by the sensor. However, the extrapolation is ill-defined as one does not know what an object looks like when not seeing it from the other side. As an example, one cannot see if the co-driver door of the truck in Figure 2.1 is open. A bounding box that includes an opened co-driver door of the truck would be wider than a bounding box that does not include an opened co-driver

door. Bounding boxes that do not include this case assume that the door is always closed as neither a human nor a sensor is capable of detecting the co-driver door of the truck from the perspective in Figure 2.1. However, both approaches correctly describe the object with a bounding box.

To summarize, one has to be aware that corners of surrounding objects which are not within the FOV of the sensor due to occlusion may not be relevant in the evaluation of the sensor performance. At worst including occluded regions in the evaluation may lead to an underestimation of the environment perception reliability of the sensors and the sensor system. An underestimation of the perception reliability can have a crucial influence on the validation of the environment perception and, thus, on the release of automated vehicles. This, however, is an effect of bounding boxes. The occluded object faces are defined by the faces that are not occluded.

In order to circumvent the evaluation of the occluded faces of the object, one could only investigate the sides that can be observed by the sensor. The first approach differentiates between the object sides that are inside and outside the FOV. The second approach determines the allowed deviations of all object sides individually, which also allows to extract only the object sides inside the FOV. Thus, it would be possible to only investigate the detected lengths and positions of the object faces inside the FOV. This does make a difference in case only a single object face is detected. In this case, only one side of the rectangle would have to match the detection. However, it does not change the evaluation for the cases where two object faces are detected as two sides of the bounding box rectangle already define the entire rectangle/box.

Another difficulty of bounding boxes, besides the fact that bounding boxes do not represent the objects properly and that occluded object faces underlie the discussed assumption, is that bounding boxes are associated with a classification of the object. However, there might be objects that cannot be classified in any of the finite set of classes, e.g., a combination of objects like a bicycle on the back of a car or objects that are rare on the street like tractors or harvesters. Such objects might not be included in the evaluation of the environment perception. In most studies, we only focus on cars and avoid the discussion about what objects to include in the evaluation. Bounding box detections of objects are not able to cover the infinite number of different objects that are possible on public roads. In addition, bounding box detections do not account for the detection of the driving lane on the street. Street and lane detection, however, is also an important part of the environment perception. Instead of focusing on object detection for the evaluation of the environment perception, one may require a more general approach that can account for all different types of objects and that also takes the detection of the street into the evaluation.

Instead of an allowed deviation that increases linearly with the distance, one can also use other functions to account for deviations. First, some sensors may perform differently in different directions due to the limited FOV. Therefore, one may want to account for the difference in the directions in the evaluation of the sensors. Second, in some directions, a precise detection might not be that relevant compared to other directions. For example, it may be more relevant to detect the car in front of the ego vehicle precisely than the car to

either side of the ego vehicle. Third, the FOV of the sensor is also limited in the distance from the sensor. Therefore, one might want to perform the evaluation only for limited distances. Besides a clear threshold, one could also use a function where the increase in the deviation has a singularity at a certain distance indicating the maximum range of the sensor. One such approach is introduced in section 3.6.

[77] argues that the misdetections by the perception are velocity dependent due to the fact that misdetections in far distances can become relevant at high speed. This is a function-specific approach, e.g., for the adaptive cruise control on the motorway. However, as car manufacturers do not know whether their customers will drive on motorways or in cities with their automated cars, these cars and, thus, their environment perception, need to perform reliably on motorways as well as in cities. Therefore, we suggest not making the analysis dependent on the mission profile for highly automated vehicles. Thus, we would also not incorporate the ego velocity in the evaluation of the environment perception. If a sensor has a maximum range of 100 m one should not expect the sensor to detect objects at distances greater than 100 m. As described in the previous paragraph the introduced approach could also account for this effect by choosing a function where the acceptable tolerance goes to infinity for distances greater than the maximum sensor range.

3.5 Drivable area-based error definition

The previous sections focus on defining the error on the object data level. Research in the field of the perception for automated vehicles often focuses on object detection [18, 27, 49]. Therefore, validation of object detection is currently the most intuitive approach. It also allows comparing different sensors with each other as the sensor's raw data is transformed into the same data format. A comparable data format is required for some validation techniques [16].

This section introduces an error definition based on the drivable area rather than on object detection and association. The drivable area is the frame-wise area that can be accessed by the vehicle using a translation in any radial direction. This includes the first order approximation of the vehicle's driving trajectory. In case the drivable area is utilized to determine the reliability of perception sensors, the drivable area has to be derived from sensor raw data instead of object bounding boxes. As with object detection algorithms, a good approximation of the drivable area might be derived using neural networks. While the drivable area focuses on the closest obstacles only, trajectories of other objects might be implicitly included in future predictions of the drivable area.

This section is structured as follows: Section 3.5.1 performs a comparison of the drivable area with free space and occupancy grid calculations. Section 3.5.2 provides one method for determining the drivable area-based on a radial investigation of LiDAR data. Section 3.5.4 introduces the error definition based on the drivable area, followed by the result and discussion in section 3.5.5.

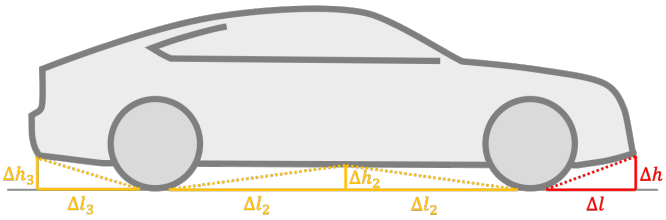


Figure 3.22: A car can only surpass a certain slope without touching the ground. Actually, the change in the slope is relevant. However, from the ego perspective, we consider that the closest distance at which the maximum slope is exceeded does not belong to the drivable area anymore. In our analysis, we take a conservative measure of $12.5\,\mathrm{cm\,m^{-1}}$. This is lower than the value accessible by a Porsche 911, which is car with a relatively small height of the auto body above the ground [79].

3.5.1 Drivable area, free space and occupancy grid

Occupancy grid and free space approaches often incorporate the sensor information available from one or multiple frames in a single map [26, 42, 78]. The definition of the drivable area is similar to the definition from [44]. The drivable area does not provide an ever-increasing map but a frame-based linear approximation in every radial direction of the current situation which does not exceed any object that cannot be passed in the radial direction from the current ego position.
The following section is introducing an approach to get a rough approximation of the drivable area. Analogue to [44], the introduced approach is only based on LiDAR data. It does not utilize neural networks and is based on the gradient and height difference in any radial direction.

3.5.2 Deriving the drivable area from LiDAR data

This section introduces a procedure to obtain the drivable area in all radial directions from the ego position of the vehicle based on LiDAR data. For the determination of the drivable area, one may assume that the vehicle can exceed a threshold of Δh_t and a maximum slope $\Delta h_s / \Delta r_s$. The maximum accessible threshold and the maximum slope that is accessible are defined by the vehicle parameters of each vehicle as shown in Figure 3.22 and 3.23.
The approach excludes all LiDAR points that are more than 5 cm above the ego vehicle. The evaluation uses the height difference and the slope between subsequent LiDAR measurements along the radial direction. Neighboring radial lines are separated by an angular deviation of $\Delta\varphi_{dev}$. The angular deviation is defined by the angular resolution of the LiDAR sensor of the azimuth angle. The azimuth angle is the angle between the x and the y coordinates while the x, y-plane corresponds to the driving plane of the vehicle. In the case of the nuScenes dataset, the LiDAR sensor was calibrated and was not aligned parallel to the driving plane. Therefore, instead of choosing the LiDAR resolution, one may choose a predefined value for $\Delta\varphi_{dev}$.
All points of the LiDAR point cloud that are recorded in the same interval $\Delta\varphi$ are then used to generate the radial height profile. The LiDAR points within the same interval are ordered according to their radial distance from the ego position. Plotting the height versus

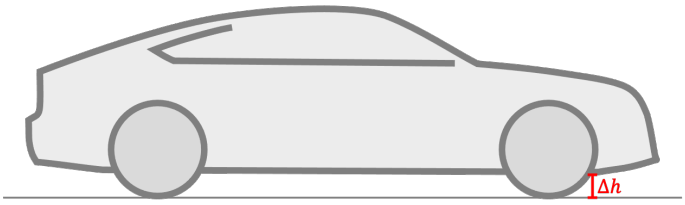


Figure 3.23: Maximum height that is surpassable by the vehicle without major damage to the car. This is just a rough approximation as, depending on the speed, such a threshold in the height might be devastating. However, at faraway distances, this height might provide a good approximation of heights that do not need major attention. In this study, we use a height threshold of 20 cm for subsequent measurements to be not considered as drivable area. 20 cm is larger than the indicated height for most cars. However, at faraway distances, for which we consider the approach more suitable, height differences smaller than 20 cm between subsequent measurements are not expected to be of major safety concern.

the sorted radial distances of all LiDAR recordings within the interval $\Delta\varphi$ provides the distance-dependent height profile along a radial line.
The height difference and the slope between radially subsequent LiDAR points are derived from the height profile. The height difference is obtained by subtracting the height of subsequent LiDAR points in a radial direction. The slope between subsequent LiDAR points is then obtained by dividing the obtained height differences by the difference of the radial distance of the same two subsequent points.
The distance r at which both, the height difference and the slope, exceed the threshold defines the border of the drivable area. The determination of the border of the drivable area is repeated for all azimuth angles φ with a discretization of $\Delta\varphi$ in order to determine the drivable area along all directions.
This analysis utilizes a height threshold of $\Delta h_t = 5\,\mathrm{cm}$ and a maximum slope of $\Delta h_s/\Delta r_s = 5\,\mathrm{cm\,m^{-1}}$. The discrete step in the azimuth angle is set to $\Delta\varphi = 1°$.

Results and Discussion:

Figure 3.24 demonstrates the drivable area for a single frame of the nuScenes dataset [27] shown by the grey area. The LiDAR point cloud is shown on top of the drivable area, with a color scheme that indicates the height of the recorded LiDAR points. The red line highlights all points along the radial line at an azimuth angle of 31°.
Figure 3.25 (a) demonstrates the height profile along the radial line at azimuth angle $\varphi = 31°$. The points indicate the recordings. The point at which the threshold exceeds 5 cm and where the slope exceeds $5\,\mathrm{cm\,m^{-1}}$ is indicated by the dashed line. It is observed at 12.07 m. The drivable area in Figure 3.24 roughly indicates the street. However, the drivable area exceeds the street at some angles, indicated by the spikes mainly in the lower part of the Figure.
The difference in the height of subsequent LiDAR points and the derivative of the height profile along a radial direction at an azimuth angle of 31° are shown in Figure 3.25 (b). Both curves are normalized by the maximum acceptable values of 5 cm and $5\,\mathrm{cm\,m^{-1}}$, respectively. Therefore, the border of the drivable area along this radial line is defined

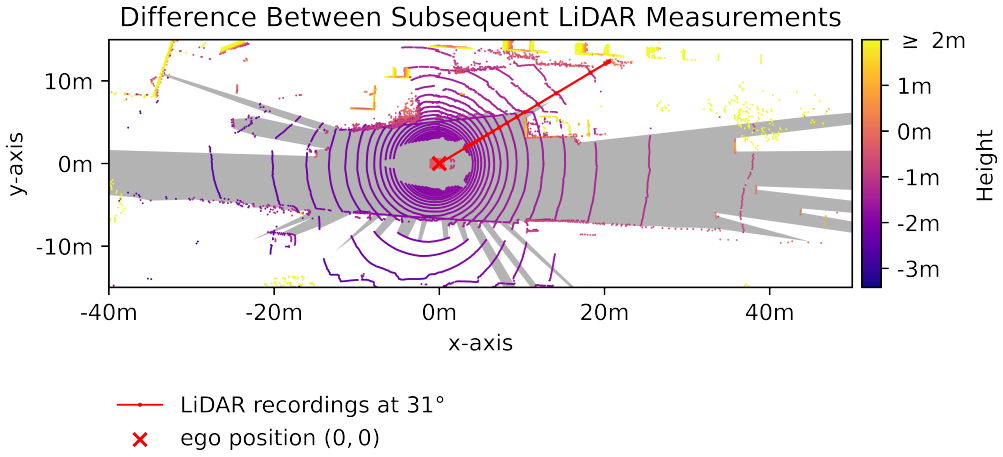
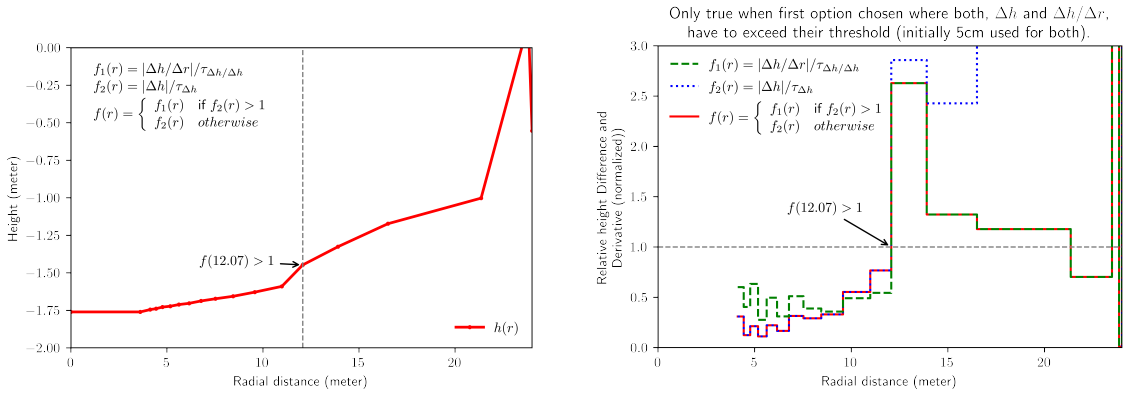


Figure 3.24: The Figure is showing the drivable area by the grey region together with the LiDAR point cloud. The corresponding camera images of the frame are shown in Figure 2.1. The drivable area is determined by the distance at which the height profile has a threshold greater than 5 cm and a slope greater than 5 cm m^{-1} in any radial direction. The red line highlights the LiDAR points at an azimuth angle of 31°. The height profile along this radial line is shown in Figure 3.25 (a).

by the smallest distance where both curves exceed the value of 1. The point where both profiles exceed the threshold is additionally shown by the arrow with the label $f(r) > 1$.
The presented approach to determine the drivable area performs reasonably well in finding the surrounding street in this particular frame. Except in some locations at individual azimuth angles the border of the drivable area is set to the border of the street or surrounding objects. However, the approach has two drawbacks:
First, the defined height threshold of 5 cm is restrictive enough in far distances. At far distances, subsequent LiDAR points in radial direction can be a few meters apart. In case of an incline or decline of the street, subsequent LiDAR points can differ more in height. However, for the near field, where many measurements are taken nearby, a threshold of 5 cm might be too loose. Imagine three subsequent LiDAR measurements that have a height difference of 5 cm between each other. As the approach accounts for the radial distance for the height threshold, these three measurements might be very close to each other. Thus, the effective height difference might be around 10 cm of the nearby LiDAR measurements. More LiDAR measurements could also occur at the same distance, leading to an even higher height difference for nearby measurements.
Second, the slope might be quite small at far distances even for objects that have a height in the order of 1 m. As in large distance subsequent LiDAR measurement can be quite far apart, the measurement of a surrounding object might be at a height of 0.5 m. Now consider that subsequent LiDAR measurements in the radial direction are 10 m apart from each other as measurements in the far distance get sparse as seen in Figure 3.24. In this case, the slope is 5 cm m^{-1} even though the object is at least 0.5 m high.
This section investigates the two limitations of the approach individually and provides another approach that combines the threshold and slope-based methods in a different way.



(a) (b)

Figure 3.25: (a) Height profile for the frame from Figure 3.24 along the radial line at an azimuth angle of 31° as indicted in Figure 3.24 by the red line. (b) Differences in the height profile and slope between subsequent LiDAR recordings. The height differences and the slope are normalized by the threshold which the car is expected not to exceed. The border of the drivable area is, therefore, defined by the closest distance at which both functions exceed 1.

The following section use image processing tools to exclude the spikes from the drivable area in Figure 3.24.

3.5.3 Adjustments of LiDAR-based drivable area using morphological operations

Figure 3.24 gives an approximation of the drivable area of a single frame of the NuScenes dataset. The drivable area demonstrates spikes that are not intended for driving and are an effect due to low height differences and slopes between subsequent LiDAR points along these radial directions. However, these regions might not be accessible by the car due to minor thresholds that are not detected by the used approach. Even if no threshold is observed in these directions, the regions are too narrow for the ego vehicle to pass. Morphological operations, specifically the morphological opening, allow the reduction of the occurrence of such protuberances within the LiDAR-based drivable area.

Background on morphological operations

Morphological operations are performed on binary images. A morphological opening consists, first, of a morphological erosion and, second, of a morphological dilation [80, 81]. Binary images can be described as a set of 2D-integer values in the $\mathbb{Z}^2$ space. The morphological erosion of binary image set A and structuring element B, denoted $A \ominus B$, is defined as the set of points $z \in \mathbb{Z}^2$ such that the set B translated by z is a subset of A [81]:

$$A \ominus B = \{z \mid (B)_z \subseteq A\} \tag{3.10}$$

In the following, this work adapts the terminology of the OpenCV package and refers to the structuring element as kernel [82]. The morphological dilation of a binary image set A

and a kernel B, denoted as $A \oplus B$, is defined as the points $z \in \mathbb{Z}^2$ for which the cardinality of the intersection of the set A with the set B translated by z is larger than zero. In other words, where the intersection is not equal the empty set [81].

$$A \oplus B = \{z \mid (B)_z \cap A \neq \emptyset\} \tag{3.11}$$

Morphological opening corresponds to the subsequent application of erosion and dilation. The morphological opening erases contours that are smaller than the set defined by the kernel. Therefore, morphological opening provides a method to erase the thin spikes from the drivable area from Figure 3.24.

Adjusting the LiDAR-based drivable area using a morphological opening

To mitigate the presence of the spikes in the drivable area one can apply a morphological opening on the LiDAR-based drivable area from section 3.5.2 which is characterized by a set of angular and radial distances. In order to apply a morphological opening, the process starts with converting the polar coordinates of the polygon into an image representing the drivable area. For image processing, we utilize the implementations of the OpenCV python package [82]. The size of the image is defined by a maximum and a minimum value in x- and y-direction, parallel and orthogonal to the driving direction. The evaluations in this section use an interval of $[-90\,\mathrm{m}, 90\,\mathrm{m}]$ in x- and y-direction for our calculations. A spatial resolution of $20\,\mathrm{cm} \times 20\,\mathrm{cm}$ is used. In order to apply the morphological operations, one has to transfer the polygon to a binary mask by filling the polygon in image coordinates. The subsequent step involves performing a morphological opening on the binary mask. A kernel with a length of 4.6 m and a width of 1.8 m approximately represents the size of the ego vehicle. The length and width of the kernel have to be adapted to the utilized image resolution.
Figure 3.26 demonstrates the drivable area after the morphological opening is performed on the drivable area from section 3.5.2 in Figure 3.24. The initial protuberances in the LiDAR-based drivable area, which the ego vehicle cannot pass as they are too narrow, are removed in Figure 3.26. The drivable area from Figure 3.26 provides, therefore, a more realistic approximation of the area where the ego vehicle can indeed go. However, it is noteworthy that a morphological opening can only provide a rough approximation of the vehicle movement in the 2D plane. It represents a translation of the rectangle while a rotation of the vehicle around its z-axis, the axis orthogonal to the driving plane, is not considered.

3.5.4 Error definition

The detection of the drivable area comes with the advantage that it incorporates all surrounding objects and obstacles. Object detection algorithms in comparison can only deal with predefined objects and neglect all objects of any other classes. In automated driving, usual object detection algorithms focus on non-stationary objects like other cars, trucks, buses, pedestrians, cyclists, etc. However, many stationary objects like for example

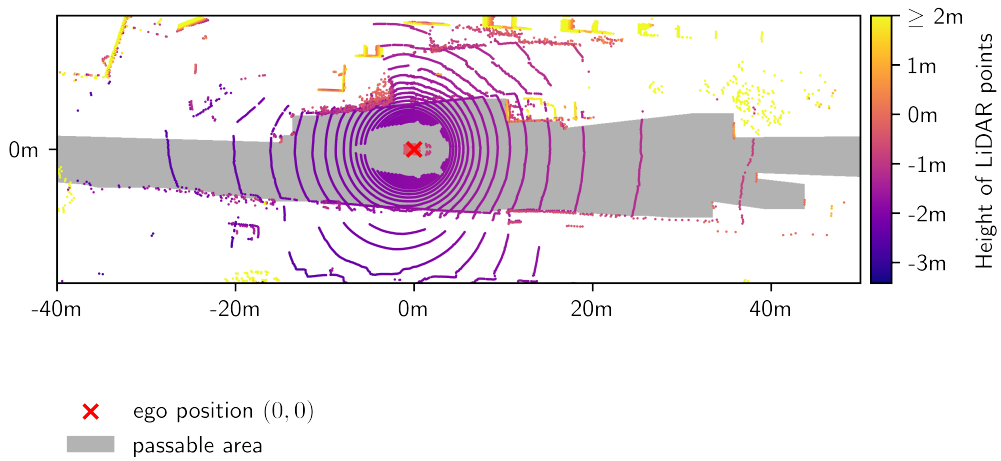


Figure 3.26: The drivable area obtained after applying a morphological opening on the drivable area of Figure 3.24. The kernel used for the morphological opening is a rectangle of the size of the ego-vehicle. The opening operation gives an approximation of where the ego vehicle can drive. It is just a rough approximation as it only considers a translation of the vehicle in any direction, but no rotation. However, it reduces the spikes of the drivable area from Figure 3.24 which are too narrow for the ego vehicle to pass.

trees are often not considered in datasets meant for automated driving. For instance the Waymo open dataset contains labels for vehicles, pedestrians, cyclists and traffic signs [1]. The KITTI dataset contains labels for 8 object classes and the nuScenes dataset contains labels for 23 object classes [27, 40]. The problem remains that a finite number of object classes cannot represent all possible objects on the street. Investigating the detection of the drivable area instead of the detected objects may, therefore, allow a better estimation of the sensor reliabilities and the underlying algorithms.

In addition, one should intend to define an error that accounts for the physical properties and limitations of the sensors. The sensors record the surroundings from the ego perspective in radial direction with a sensor-specific angular resolution. This accounts for all major types of sensors utilized in automated driving namely LiDAR, camera and RADAR. Obstacles at distances further away occupy a smaller solid angle in the FOV of these sensors. Thus, the number of detections of an object of the same size decreases with the distance, resulting in a less accurate detection of objects at locations further away from the ego vehicle. Figure 3.27 illustrates the number of detections for a car of the same size at distance $r_{2D,ref}$ and distance $2 \cdot r_{2D,ref}$. The beams in Figure 3.27 illustrate for example the angular resolution of a camera or a LiDAR sensor. The object that is located at a distance of $2 \cdot r_{2D,ref}$ is only detected at a solid angle that is half the size of the solid angle of the detection of the object at distance $r_{2D,ref}$. Subsequently, the object is only detected by half the number of pixels in an image or by half the number of LiDAR points in the point cloud. Figure 3.27 illustrates the example in the 2D plane. Here, the number of detections of an object scale with the inverse of the distance. In 3D the number of detections scales with the inverse of the distance squared. Thus, if an object is twice as far away, it only occupies one fourth of the image from a camera.

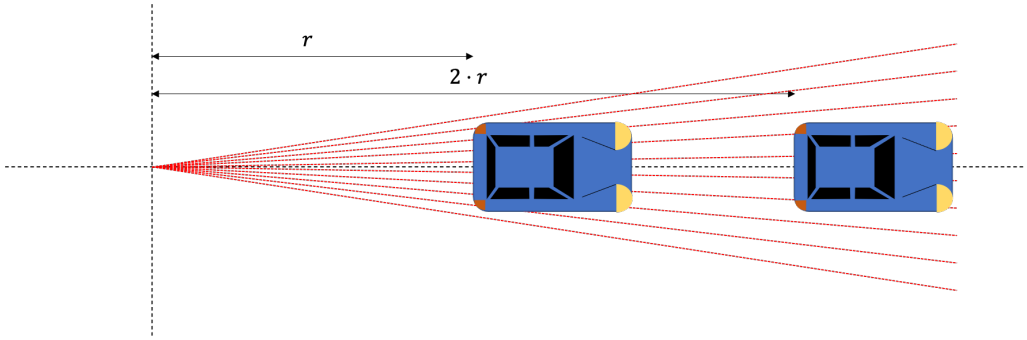


Figure 3.27: The Figure illustrates the number of recordings of an object at different distances. The red beams illustrate the discrete angles at which the sensor records the surrounding. For a LiDAR the discrete angles are determined by the angular resolution and for a camera, the angles are determined by the number of pixels and the objective lens. Only half the number of detections is observed for an object that is twice as far away from the ego vehicle in this 2D illustration (4 beams instead of 8 beams). For 3D the number of detections of an object decreases with the inverse of the distance squared.

Figure 3.27 also illustrates that the sensor recordings occur for discrete angles. The maximum resolution is defined by the angular difference $\Delta\varphi$ between two subsequent measurements. The deviation in the azimuth angle can be up to $\pm\Delta\varphi$. The translational deviation should, therefore, be dependent on the distance to the ego vehicle.
Moreover, the accuracy in the detection of the distance estimation can be worse for objects that are further away from the ego vehicle. This might be caused for example by a systematic, distance-dependent error in the LiDAR calibration or due to the worse performance of a stereo system for objects that are far away [30, 83]. Actually, an automated vehicle may not rely on perfectly accurate distance estimation of faraway objects. Therefore, this measure scales the allowable tolerance with the distance like in section 3.4 in addition to the angular deviation.
Starting with the mask of the drivable area in polar coordinates one can derive a lower limit of the drivable area and an upper limit for the drivable area based on the acceptable angular deviation. The lower limit is obtained by performing a morphological erosion on the binary mask of the drivable area in polar coordinates. The kernel for the erosion is chosen such that it only allows a deviation in the angular dimension with a width of $2\Delta\varphi_{dev}/\Delta\varphi_{img}+1$ while $\Delta\varphi_{dev}$ is the accepted angular deviation and $\Delta\varphi_{img}$ is the angular resolution of the image. $\Delta\varphi_{dev}$ should be a multiple of $\Delta\varphi_{img}$. In case the resolution is in agreement with the deviation as illustrated in Figure 3.27, the $\Delta\varphi_{dev}$ and $\Delta\varphi_{img}$ are in agreement. The upper limit is obtained by performing a morphological dilation on the polar representation of the drivable area. The kernel used for the morphological dilation is the same as the kernel that is used for the erosion according to the deviation of $\pm\Delta\varphi_{dev}$.
The binary mask in polar coordinates reaches from 0° to 360°. For the morphological operations, one is required to extend the boundaries of the binary mask on every side beyond the image boundaries at 0° to 360° to avoid artifacts. This is due to the fact that the polar coordinate frame is periodic. The extension has to be half the width of the kernel. After the erosion and the dilation, respectively, the mask of the drivable area in

polar coordinates should be reduced to the initial range from 0° to 360°.
The next step incorporates the acceptable deviation based on the distance from the ego vehicle. The measure accounts for the fact that the accuracy of the drivable area should be higher around nearby objects than it is required around objects that are far away. Thus, the proposed measure scales the acceptable deviation linearly with the distance from the ego vehicle. The linearly distance-dependent deviation around the boundaries of the drivable area is achieved by transforming the binary mask of the drivable area in polar coordinates into a polygon in polar coordinates. This is repeated for both, the upper and the lower limit that is obtained from applying the angular deviation. The radial distance of the polygons can then be multiplied with $(1 \pm \alpha_{dd})$ while α_{dd} determines the distance-dependent acceptable deviation. The radial distances of the polygon obtained from erosion of the drivable area mask are multiplied with $(1 - \alpha_{dd})$ to provide the lower limit. The radial distances of the polygon obtained from dilation are multiplied with $(1 + \alpha_{dd})$ to provide the upper limit, respectively.
A detection of the drivable area has to be within the lower and the upper limits of the reference area in order to be detected correctly. Deviations larger than acceptable can be interpreted as an incorrectly identified frame and the errors can be counted framewise. An alternative is to count the percentage of the azimuth angle for which the detection deviates larger than acceptable.
For illustration, the acceptable deviation for the drivable area is evaluated for one frame. The constant α_{dd} is set to 10 %, $\Delta\varphi_{dev}$ is 1° and $\Delta\varphi_{img}$ is 0.01°. The small value for $\Delta\varphi_{img}$ guarantees a high resolution for the resulting Figure. Thus, we differentiate here between $\Delta\varphi_{dev}$ and $\Delta\varphi_{img}$. The drivable area from Figure 3.26 is utilized for the evaluations as a reference area.

3.5.5 Results and discussion

Figure 3.28 shows in green the allowed deviation from the reference area. The reference for the drivable area is illustrated in grey. In addition, Figure 3.28 shows the LiDAR points of the recorded scene. The color bar indicates the height at which the LiDAR points were measured.
One can see that the allowed deviation increases at distances further away from the ego vehicle. The angular deviation is constant which can be observed at 1° and at 178°. These two angles mark the border of two detected objects. There is a detection of a close obstacle at angle φ and a detection of a far obstacle at angle $\varphi + \Delta\varphi$.
Figure 3.29 shows the drivable area together with the defined acceptable tolerance in Cartesian coordinates. The position of the ego vehicle is indicated by a red cross. Analog to Figure 3.28, Figure 3.29 shows the LiDAR point cloud with a color map that shows the height of the LiDAR measurements. At small distances, the deviation corridor is narrow while it is larger at distances further away from the ego vehicle.
The approach demonstrates an alternative to the validation of perception sensors based on object data from object detection algorithms. So far, the proposed approach comes with the difficulty that nearly no datasets with reference data of the drivable area exist. The

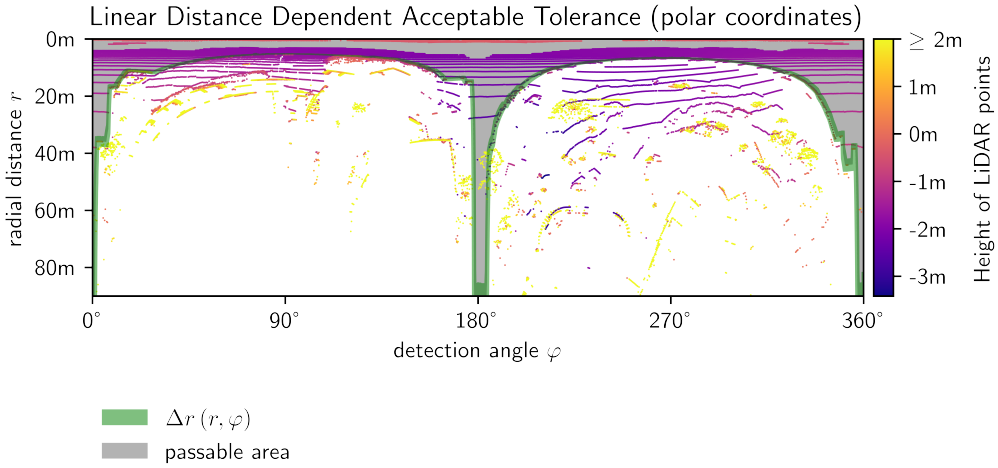


Figure 3.28: Drivable area (grey) and acceptable deviation (shaded green) polar coordinates. The angular deviation is set to $\Delta\varphi_{dev} = 1°$ and the acceptable deviation for the distance estimation is set to $\alpha_{dd} = 10\,\%$. The LiDAR point cloud is shown in order to obtain a height estimation of the surrounding.

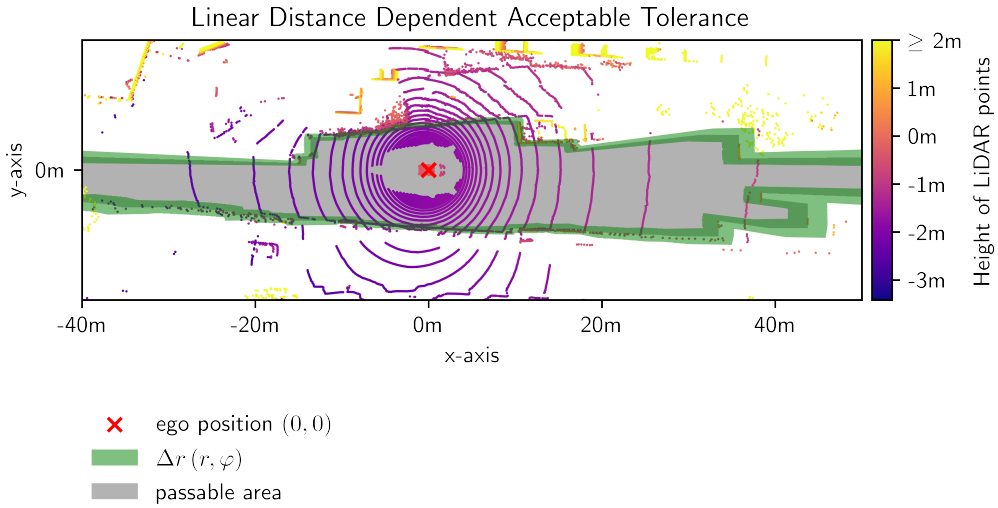


Figure 3.29: Drivable area (grey) and acceptable deviation (shaded green) in Cartesian coordinates. The angular deviation is set to $\Delta\varphi_{dev} = 1°$ and the acceptable deviation for the distance estimation is set to $\alpha_{dd} = 10\,\%$. The ego position is indicated by the red cross. The LiDAR point cloud is shown in order to obtain a height estimation of the surrounding.

nuScenes dataset is one of the only datasets that also contains information about the shape of the street [27]. This additional information is also used in the planner-centric measure for the validation of the perception of automated vehicles [22]. In contrast, [22] validates the perception based on predicted trajectories of the surrounding objects.
In this section we introduced two ways of determining the drivable area from highly resolute LiDAR data. However, most probably this can be further improved using neural networks. Neural networks could also be used for a method that derives the drivable area from camera data. Yet, the focus of this study is to provide a method for determining the reliability of perception sensors and not on determining the drivable area. Thus, only four approaches based on algebra and image processing are introduced here.
In comparison to commonly used association measures, this measure accounts for the distance dependence of the translational error. As all perception sensors in automated driving record the surrounding in radial lines for equally spaced azimuth angles, a detection that is far away cannot be as accurate as a nearby detection. Actually, detections at far distances may not even have to be as accurate as nearby detections. Therefore, this measure might provide a useful tool in the evaluation of perception sensors.
Instead of using a deviation interval that depends linearly on the distance, the measure can be extended by incorporating other dependences of the accuracy in the detection of the surrounding. For example, at the borders of the FOV of sensors the performance of the sensors may be less. So, one could also incorporate an angular dependence. One can also incorporate the fact that at some distance the sensor is not capable of any distance estimation. Section 3.6, which focuses on systems that come closest to the perception of humans, derives a function based on the equations of a stereo camera for the accuracy in the distance estimation. Thus, instead of using a linear scaling of the acceptable deviation, one could use the function derived in section 3.6 for the scaling of the acceptable deviation.
One could also evaluate future predictions of the drivable area using this approach. [44] introduces a method to extrapolate the free space into the future. The definition of the free space by [44] is similar to the definition of the drivable area except that they limit the definition of the free space to LiDAR data.
The recorded deviations provide a distribution for the difference between the reference area and the detected area. The distribution can also be generated for a specific distance and/or a specific interval of the azimuth angle. One could conclude from the distributions the performance of individual sensors depends on the direction and the distance to surrounding obstacles instead of obtaining a specific error rate.

3.6 Parallels and comparison with the human perception

As discussed in the introduction, human driving is often used as a benchmark for automated driving in terms of safety by utilizing the rate of fatal accidents [4, 77].
A perception error may not directly lead to a fatality. Thus, for the validation of the environment perception of automated vehicles, one has to base the error definition on other parameters. In order to obtain a benchmark for a sufficient environment perception,

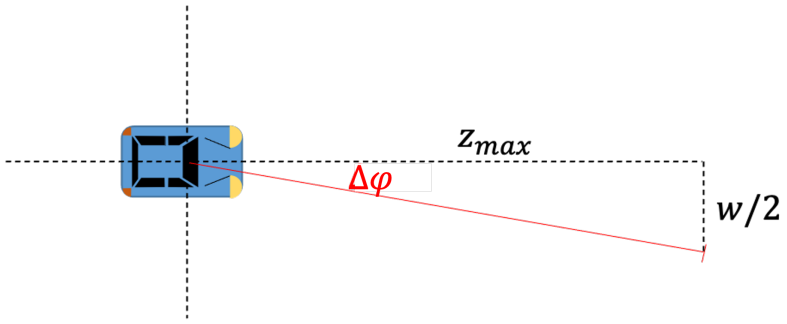


Figure 3.30: Visualization to evaluate the maximum distance z_{max} at which an object of width w_{2D} can still be observed given a minimum resolution of $\Delta\alpha$.

one can quantify the performance of human depth perception. Even though the human depth estimation is far from precise, especially at far distances, it is sufficient for driving. As a result, this section investigates the human depth perception. This work intends to find fuzzy rules that can be used to roughly estimate the performance of the human depth perception that could be used as a benchmark for a comparison with the environment perception of automated vehicles. The human perception is based on multiple principles that provide depth information [84]. This section focuses in particular on stereopsis and depth perception based on familiar object sizes.

3.6.1 Human resolution

The resolution of a sensor and of the human eye is one parameter that limits the distance for how far one can detect or see objects of a certain size. A human can approximately resolve one arcminute [85] which corresponds to $\Delta\alpha = 0.0167°$. For a primitive approximation for how far a car can be seen by a human, we made the assumption that a car is still observable if the car occupies at least a solid angle of two times the minimum resolution. Figure 3.30 provides a demonstration of the calculation. From the minimum resolution $\Delta\alpha$ and the width of the object, one can derive a maximum distance at which the object can still be observed. Here, due to the focus of object detection in traffic situations which include cars an object width of 2 m is assumed here which is approximately the width of a car. The resulting distance at which the object of 2 m in width can still be observed is $z_{max} = \frac{1\,\mathrm{m}}{\arctan(0.0167°)} \approx 3500\,\mathrm{m}$.
For comparison, one can repeat the calculations for example for the Puck LITE LiDAR from Velodyne. The Puck LITE has an angular resolution of 0.1° - 0.4° [31]. For simplification one may assume that the minimum number of two LiDAR point detections of an object in order to detect the object analog to the previous assumption. For a LiDAR with an angular resolution of 0.4° the maximum distance at which an object can be detected is 140 m.
The assumption that two LiDAR point detections might be an optimistic assumption to detect an object. It might be unlikely that an object detection algorithm can estimate an object from two single LiDAR points. Moreover, the LiDAR range is restricted by the Laser intensity. The maximum range of 100 m is neglected in the calculation [31]. However,

the assumption allows a rough but first quantitative comparison of human detection

3.6.2 Human depth perception

Human depth perception is based on many different principles [84]. For the evaluation of possible deviations in human perception, we focus on two principles: the stereopsis and the depth perception due to familiar sizes of objects. We utilize these two principles to quantify the possible deviation in the depth perception of humans. We focus on these two principles as they allow basic assumptions that are necessary for the estimation of the deviation in the depth perception. However, these two principles are only partially representative for the human depth perception and were used here to obtain a rough estimation which is a start for the comparison with the performance of depth perception of sensors.

The stereo system

Figure 3.31 demonstrates the principle for a stereo system which is a perception system which could be compared with the perception of a human. For the further estimations of the depth estimation of humans we start from the equation for a stereo system which can be derived using Figure 3.31 [30].

$$z = \frac{fb}{x_l - x_r} \tag{3.12}$$

Here, we are not interested in estimating the depth from an image where one has to estimate x_l and x_r from the image which is achieved by the correlation of small image segments along the epipolar line as described in [30]. Instead, we are interested in finding an estimate of the possible deviation of z based on the resolution of the recorded images. The resolution is provided by the difference in the angle that can be recorded $\Delta\varphi$. In order to find the deviation in z we, therefore, need to represent the image positions x_l and x_r in dependence of the angle φ_l and φ_r, respectively.

$$x_l = f \cdot \tan(\varphi_l) \approx f\varphi_l \tag{3.13}$$

The small angle approximation is often used for angles smaller than 5°. From equation (3.12) and (3.13) one can evaluate the depth based on the angles φ_l and φ_r.

$$z = \frac{b}{\tan(\varphi_l) - \tan(\varphi_r)} \approx \frac{b}{\varphi_l - \varphi_r} \tag{3.14}$$

This allows to estimate the possible deviation in the depth estimation given a resolution $\Delta\varphi$ in the angles φ_l and φ_r.

$$\begin{aligned} \Delta z &= \frac{b}{\tan(\varphi_l + \Delta\varphi) - \tan(\varphi_r)} - \frac{b}{\tan(\varphi_l) - \tan(\varphi_r)} \\ &\approx \frac{-\Delta\varphi b}{(\varphi_l - \varphi_r)^2 + \Delta\varphi(\varphi_l - \varphi_r)} \end{aligned} \tag{3.15}$$

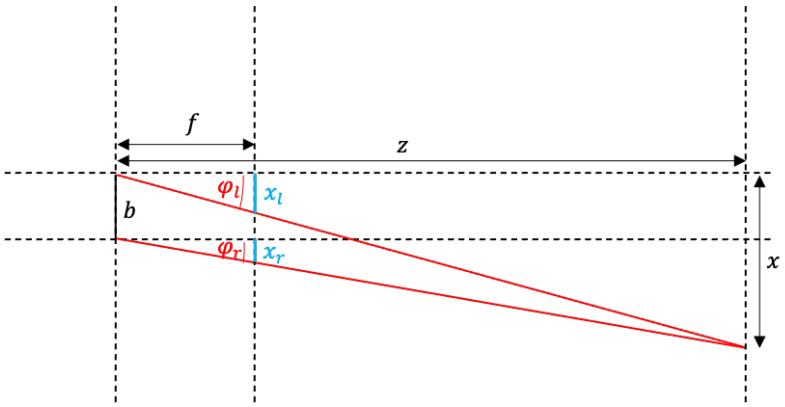


Figure 3.31: Determination of the distance based on the stereo system [30]. f represents the distance between the focal length of the camera or the human eye; the baseline b is the distance between the two cameras of a stereo system or the distance between two eyes; z is the distance to the object which is the subject of interest in a stereo system; x is the position of the object in focus; x_l and x_r are the distances of the object at position x on the camera chip or the human retina.

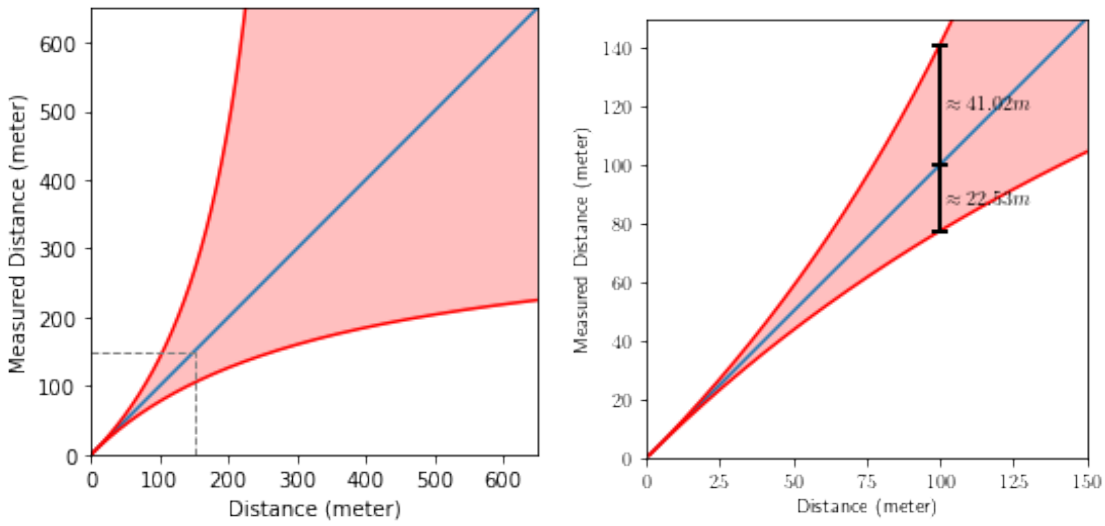
Under the assumption that the object is right in front of the ego vehicle on a central line so that $\varphi_r = -\varphi_l \approx b/(2z)$ we get a possible deviation in the depth estimation of

$$\Delta z = \frac{-\Delta\varphi z^2}{b + \Delta\varphi z}. \tag{3.16}$$

The maximum distance at which it may still be possible to estimate the depth can be derived from equation (3.16) as follows:

$$z_{max} = \lim_{z\to\infty} z + \Delta z = \lim_{z\to\infty} \frac{zb}{b + \Delta\varphi z} = \frac{b}{\Delta\varphi} \tag{3.17}$$

In case $\Delta\varphi$ is in the opposite direction corresponding to an inverse sign, the z_{max} is obtained due to the fact that for a negative $\Delta\varphi$ the fraction has a singularity for $z = b/|\Delta\varphi|$.
For the analysis of the depth estimation using the stereo system the parameter b from Figure 3.31 was set to 10 cm. 10 cm may be an upper estimation of the distance between the eyes of a human and may lead to an underestimation of the deviation in the depth estimation of a human. However, human depth estimation is also based on other principles which may lead to a better estimation than the estimation only obtained by stereopsis. Therefore, the overestimation of b may still lead to an overestimation of the deviation in human depth perception. For the evaluations the angular resolution was set to $\Delta\varphi = 1/60° = 0.0167°$ [85].
The error in the small angle approximation of $\tan(\varphi) \approx \varphi$ is smaller than 0.3 % for angles smaller than 5°. For a distance of $z = b/(2 \cdot \tan(5°)) \approx 0.57$ m are the angles φ_l and φ_r smaller than 5° for the assumptions made for equation (3.16).
Figure 3.32 shows the possible deviation of detections by a stereo system with the defined parameters. The possible deviation increases with distance according to equation (3.16). The distance to an object can be detected to lie anywhere in the interval of [87 m, 117 m] for an object at a distance of 100 m. With equation (3.17) and using the parameter values



(a) (b)

Figure 3.32: Error in distance estimation when only considering a stereo system with a distance between two eyes of 10 cm and a solid angle resolution of 0.0167° which is approximately the resolution of a human [85].

$b = 10\,\text{cm}$ and $\Delta\varphi = 0.0167°$ one obtains a maximum distance of approximately 340 m. Above this distance, the possible deviation expands to infinity meaning that it becomes impossible to make an estimation of the distance to the object except that it is further than 340 m away from the ego position.
The result of the previous section demonstrates that a human is capable of seeing an object of a width of 2 m, approximately the width of a car, for multiple kilometers. The 340 m are just the limitation of the stereo system to estimate the distance from the ego position. However, one can still differentiate between a car that is in the visible range or out of sight. Thus, even above 340 m one can estimate distances using other principles that are built on the intuition of humans. One such principle is based on familiar sizes of objects which can be compared with the size these objects occupy in the perceived image. In the following, we want to investigate the principle based on familiar sizes of objects in order to make depth estimations.

Familiar sizes

Another approach that allows humans to estimate depth, besides the stereo-based depth estimation, is based on the intuition about sizes of objects. The intuition is based on the knowledge obtained from seeing similar objects previously. The equations for the depth estimation based on familiar size objects are derived from Figure 3.33. For simplicity, we consider an object right in front of the ego-vehicle. The distance to the object can be derived from the size of the object and the solid angle that the object occupies in the visible space by the following equation.

$$z = \frac{w_{2D}}{2\tan(\gamma/2)} \approx \frac{w_{2D}}{\gamma} \tag{3.18}$$

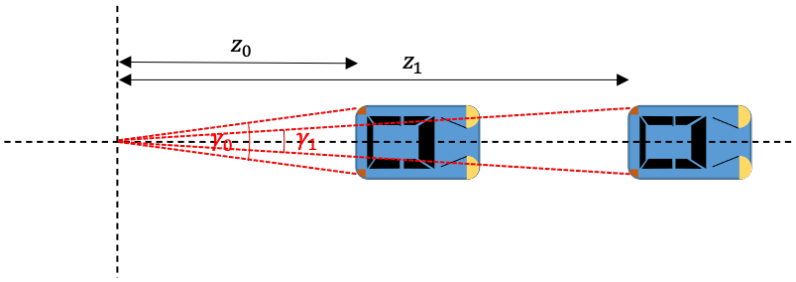


Figure 3.33: Determination of the distance based on familiar sizes.

For a provided angular resolution, we can derive the deviation in the distance that we can resolve based on the principle of the familiar sizes of objects.

$$\begin{aligned}\Delta z &= \frac{w_{2D}}{2\tan(\gamma/2)} \approx \frac{w_{2D}}{\gamma} - \frac{w_{2D}}{2\tan((\gamma+\Delta\gamma)/2)} \\ &\approx \frac{\Delta\gamma z^2}{w_{2D}+\Delta\gamma z}\end{aligned} \tag{3.19}$$

Equation (3.19) is similar to equation (3.16) for the stereo system. The difference between equation (3.16) and (3.19) is lying in the width w_{2D} of the observed object rather than the baseline b which corresponds to the difference between the two eyes or between the two cameras. Unlike the baseline b, which is the fixed distance between the two eyes or the two cameras of the stereo system, the width w_{2D} is object dependent. $\Delta\gamma$ corresponds to the minimal difference of the solid angle which the object occupies in the FOV. This minimal angle difference is defined by the resolution of the human eye or the sensor analogous to $\Delta\varphi$ in the equation for the stereo system (3.16).
Analog to the stereo system one can derive a maximum distance at which depth estimation becomes impossible.

$$z_{max} = \lim_{z\to\infty} z - \Delta z = \lim_{z\to\infty} \frac{z w_{2D}}{w_{2D}+\Delta\gamma z} = \frac{b}{\Delta\gamma} \tag{3.20}$$

The maximum distance at which the distance to an object can still be estimated is dependent on the object's size. For a car with a width of $w = 2\,\text{m}$ is the maximum distance $z_{max} \approx 3500\,\text{m}$. This is in agreement with the obtained distance from 3.6.1 that defines how far an object of width $w = 2\,\text{m}$ can be seen for a specified angular resolution $\Delta\varphi = \Delta\gamma = 0.0167°$.

3.6.3 Discussion

The safety of human driving is frequently utilized as a safety reference for automated vehicles [77]. The number of crashes or the number of fatalities per unit time driven or per kilometer driven is commonly used as benchmark [4].
However, hardly any data about the number of fatalities exist for automated driving and the industry is far from gaining the required amount of data for proving a failure rate close to the performance of human driving as the required number of drive kilometers to prove

the vehicle safety is immense [4, 75].
In the development of automated vehicles, the research focus is usually on one of three major tasks which are perceiving the environment, planning the driving path and finally steering the vehicle through the traffic. These three tasks are also described as sense, plan, act [8, 9, 86, 87]. A safe automated vehicle is required to perform all three tasks reliably. Here the focus lies on the perception of automated vehicles. The failure rates allow an intuitive and quantitative comparison between human driving with automated driving. However, with a focus solely on the perception of automated vehicles, a comparison with humans is not as intuitive. Therefore, we provide a first approach to quantify human perception. This would allow a straightforward comparison of human perception with the perception of automated vehicles.
The human perception is based on many principles [84]. This study focuses on two of these principles to allow a first estimation of the performance in depth estimation. These two principles are the stereo system and the depth perception by familiar sizes.
In both approaches the deviation in the depth estimation becomes larger for objects that are further away from the ego vehicle. While the depth estimation from a stereo system is independent of the object size, the depth estimation by familiar sizes of objects depends on the size of the respective object.
The depth estimation based on familiar object sizes allows an estimation as long as the object can be observed which is only limited by the size of the observed object and the resolution of the human eye or the sensor. Thus, the principle based on familiar sizes allows a longer range in comparison to the stereo system. However, while the stereo system depends on the physical parameters of the distance between the eyes/cameras and their resolution, the principle based on familiar sizes depends on the object size estimation by humans. This is not a quantitative measure and even though human intuition allows a depth estimation beyond the possible range of the human stereo system, the estimation in depth may not be better in the near field compared to the stereo system despite equation (3.19) let one first expect that this is the case.
Currently, most validation approaches for object detection do not take the distance to the objects into account and evaluate all objects in the same way. However, even though the deviation in distance estimations increases with an increasing distance to the objects, humans are capable to navigate vehicles safely on public roads. Therefore, one may conclude that one could implement an association that takes the distance to the objects into account and is less restrictive for faraway objects. Section 3.4 and section 3.5 provide two approaches that take the distance into account.
Unlike the deviation in section 3.5 and section 3.4 the deviation by the discussed principles of human depth perception is not linear. The function shows a maximum distance. The maximum distance corresponds to a singularity in the function for the upper bound and a constant maximum value in the function for the lower bound.
An extension of the approaches from sections 3.4 and 3.5 could utilize the deviation function from equation (3.16). Especially for camera-based stereo systems this function represents the capabilities of the system in more detail.

3.7 Conclusion

This chapter provides concepts that can be used to define an error in the environment perception of automated vehicles. Object detections that are associated with reference bounding boxes indicate a correct detection while detections without an associated reference object are classified as erroneous. Association measures for object detections are, thus, directly linked with an error definition for object detection.
The chapter first provides an overview of existing association measures between object detections and reference objects used in automated driving. Most association measures are based on the difference between the detection and reference from the object's point of view. Only one measure, the SDE, takes the ego position into account by focusing on the bounding box sides that are facing the ego vehicle's position.
Subsequently, the section introduces two other association measures that can be used to compare detections from multiple sensors instead of a comparison with the reference only. The first approach assigns detections to rectangles in a grid in order to associate them. The second approach is based on trajectory clustering for the association. These approaches are intended to be usable for an application of the model from [16] on real-world data.
In a reliability analysis, one may make the association dependent on the distance such that the impact of detection errors at different distances becomes comparable. In this case, association errors may be weighted the same independently of the distance. This chapter introduces two such association approaches that introduce a distance dependence for the evaluation of object detection.
Furthermore, the chapter addresses the limitations of bounding boxes obtained from object detection. Due to their rectangular shape, bounding boxes can only provide a very rough approximation of surrounding objects as they are commonly not rectangular. To account for some of the limitations this chapter also provides an approach based on the drivable area around the ego vehicle. Detections and evaluations of the drivable area have the advantage that they account for all possible objects in the vehicle's environment.
A benchmark for the safety of an automated vehicle is the human driver [1, 2]. The human depth perception might be utilized as a benchmark for necessary properties of the perception. Therefore, an investigation of human depth perception may allow to define sufficient properties for the environment perception of an automated vehicle. Thus, the last part of the chapter quantifies human depth perception based on a comparison with the stereo system and the familiar size-based depth estimation.
The depth estimation based on these two principles is quite rough at faraway distances. This is in agreement with the introduced distance-based association and the drivable area-based error definition. For future studies, one may adjust the distance-based association using the knowledge about human depth perception.

4 Reference-truth-based perception evaluation and reliability assessment aggregated across frames

Once a perception error is defined, an estimation for the perception reliability can be obtained by an evaluation measure that sets erroneous detections in relation to correct detections. For this, the errors are accumulated over recorded driving datasets.
Some perception errors may not have any influence on the driving behavior of the automated vehicle. Such perception errors may be excluded in the reliability evaluation of automated vehicles. Therefore, accumulating these errors in the evaluation of the environment perception is subject to discussion.
In object detection, association measures define corresponding errors for the case of object detection. The association of data from different sensors with a reference classifies the object detections as correct or wrong. The accumulation of the number of correct and wrong object detections grants insight into the performance of the sensors together with the underlying detection algorithms. The accumulation is achieved using an evaluation measure. The application of the evaluation measure is the second step of the object-based perception evaluation and is represented as the last block of the pipeline from Figure 4.1 that is highlighted in blue. Likewise, to the association measures, many measures for evaluating the detection performance exist.
Evaluation measures can be categorized into relative and absolute measures. Relative evaluation measures allow a comparison of the performance of different detection algorithms and sensors. However, such measures do not allow to draw a conclusion on whether the sensor system is sufficient for automated driving. Absolute evaluation measures should allow classifying the perception as being sufficient or non-sufficient for the use in automated vehicles.
This chapter provides an overview of perception evaluation techniques with its objectives being:

- An analysis of scenarios with perception errors that may not have a direct influence on the ego vehicle's driving behavior and a qualitative evaluation of the effect of such scenarios on the statistical assessment of the perception reliability in section 4.1.
- An inventory of existing evaluation measures that can be utilized for the evaluation of object detection is in section 4.2. The analysis lists the different evaluation measures and classifies them into absolute and relative association measures.

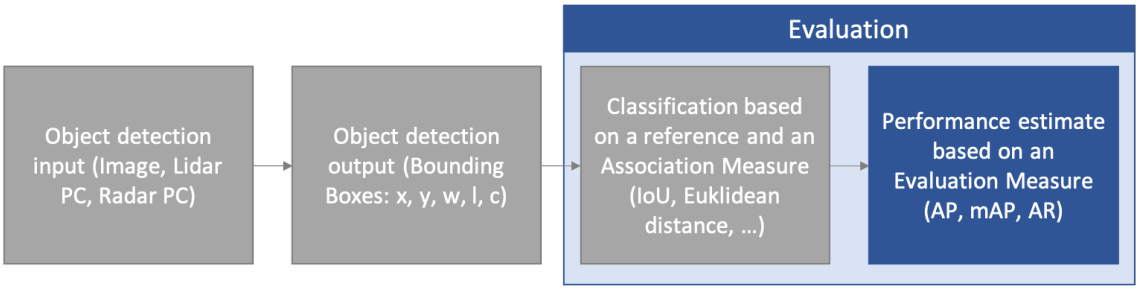


Figure 4.1: A modification of Figure 2.4. A basic representation of object detection and evaluation pipeline, starting from raw sensor data. The evaluation of the obtained object data is usually performed in two steps. The first step is the association between the sensor data and some reference data or other sensor data. The association with the reference usually classifies individual detections as correct or wrong. The second step is the evaluation which is obtained by the accumulation of some kind for all correct and wrong detections. The focus of this chapter is lying on the second step, the evaluation measures.

- An analysis of the combination of evaluation measure and association measure used for the performance evaluation of the environment perception in section 4.3.

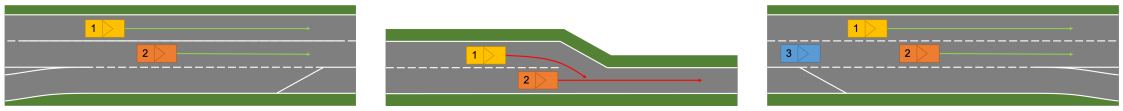
4.1 Situation-dependent safety considerations

Detection failures may not have an influence on the driving behavior of the ego vehicle. In particular, objects whose trajectory never passes the ego trajectory are safe not to detect. The following section demonstrates example scenarios in which detections may not have an influence on the ego trajectory. Based on the examples we discuss the use of scenario-based testing. Subsequently, scenario-based testing is compared with a statistical assessment of the perception reliability.

4.1.1 Impact of vehicle constellations on the perception performance

The following section investigates three vehicle constellations. A vehicle constellation describes the positions of multiple vehicles relative to each other. The same vehicle constellation can occur in different scenarios. This section demonstrates multiple scenarios for the three considered vehicle constellations. The different scenarios for the three vehicle constellations are illustrated in Figures 4.2, 4.3 and 4.4. Each vehicle is assigned a number. The ego vehicle corresponds to the vehicle with index 1. The illustrations are based on right-hand traffic.

In the first constellation Vehicle 1 drives on a trajectory that is parallel to the trajectory of Vehicle 2. Vehicle 1 is slightly behind Vehicle 2 while having a higher speed compared to Vehicle 2. Typical scenarios with such vehicle constellations are overtaking maneuvers. Figure 4.2 illustrates different overtaking maneuvers. In Figure 4.2 (a) the lane marking does not allow to interchange lanes. Assuming that both vehicles conform to public road regulations, a detection of Vehicle 2 does not have an effect on the trajectory of Vehicle 1. In the scenario of Figure 4.2 (b) and (c) this is different. Figure 4.2 (b) illustrates a



(a) (b) (c)

Figure 4.2: Three possible scenarios on (German) motorways. Vehicle 1 is considered to be the ego vehicle. The scenarios from left to right: (a) A continuous line separates the two lanes so the vehicles are not supposed to change lanes there. This is common in combination with a motorway access road. (b) End of a motorway or a two lane interval. (c) Common motorway interval with a dashed line. Thus, changing lanes is permitted. The scenario considers that the ego vehicle just passed another vehicle.

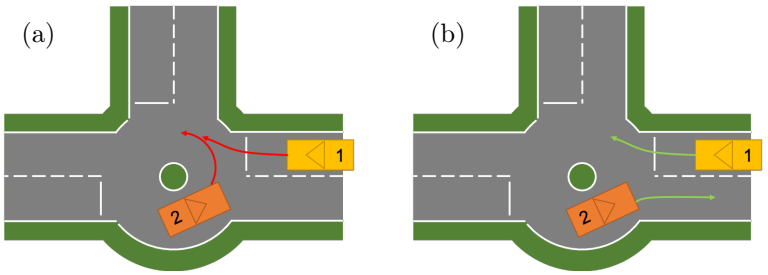


Figure 4.3: Two possible scenarios in a roundabout. Vehicle 1 is considered to be the ego vehicle. From left to right: (a) Vehicle 2 stays in the roundabout. (b) Vehicle 2 exits the roundabout.

scenario where Vehicle 1 has to change from the left lane to the right lane as the left lane is about to end. Detecting Vehicle 2 in this situation is safety critical as the trajectories of Vehicle 1 and Vehicle 2 cross due to the higher speed of Vehicle 1. Figure 4.2 (c) illustrates a similar scenario. Here, Vehicle 1 is on the left lane as it just passed Vehicle 3 and it can keep driving on the left lane to also pass Vehicle 2. Due to the obligation to drive on the right-hand lane according to public road regulations, Vehicle 1 is supposed to change from the left to the right lane if possible. Therefore, detecting Vehicle 2 is also safety-critical in this situation. If Vehicle 2 is not detected, Vehicle 1 will change to the right lane and the trajectories of Vehicle 1 and Vehicle 2 will cross, leading to an accident. Only if Vehicle 1 detects Vehicle 2 it will keep driving on the left lane as indicated by the green arrow. In summary, in two out of the three scenarios, the detection of Vehicle 2 by Vehicle 1 is safety-critical.

Figure 4.3 shows two different scenarios for another vehicle constellation. Vehicle 1 intends to enter a roundabout in both scenarios. However, in Figure 4.3 (a) Vehicle 2 stays in the roundabout while in Figure 4.3 (b) Vehicle 2 leaves the roundabout. In the scenario of Figure 4.3 (a) a detection of Vehicle 2 is safety-relevant. However, in the scenario of Figure 4.3 (b) a detection of Vehicle 2 does not have an influence on the trajectory of Vehicle 1. Thus, in this case, a detection of Vehicle 2 is not safety critical.

The third vehicle constellation with two vehicles is illustrated in Figure 4.4 which considers nine different scenarios at a four-way-intersection. The nine different scenarios are derived from the fact that each of the two vehicles has three possible directions to go if excluding the possibility of U-turns. In four out of the nine scenarios, the trajectories of the two vehicles

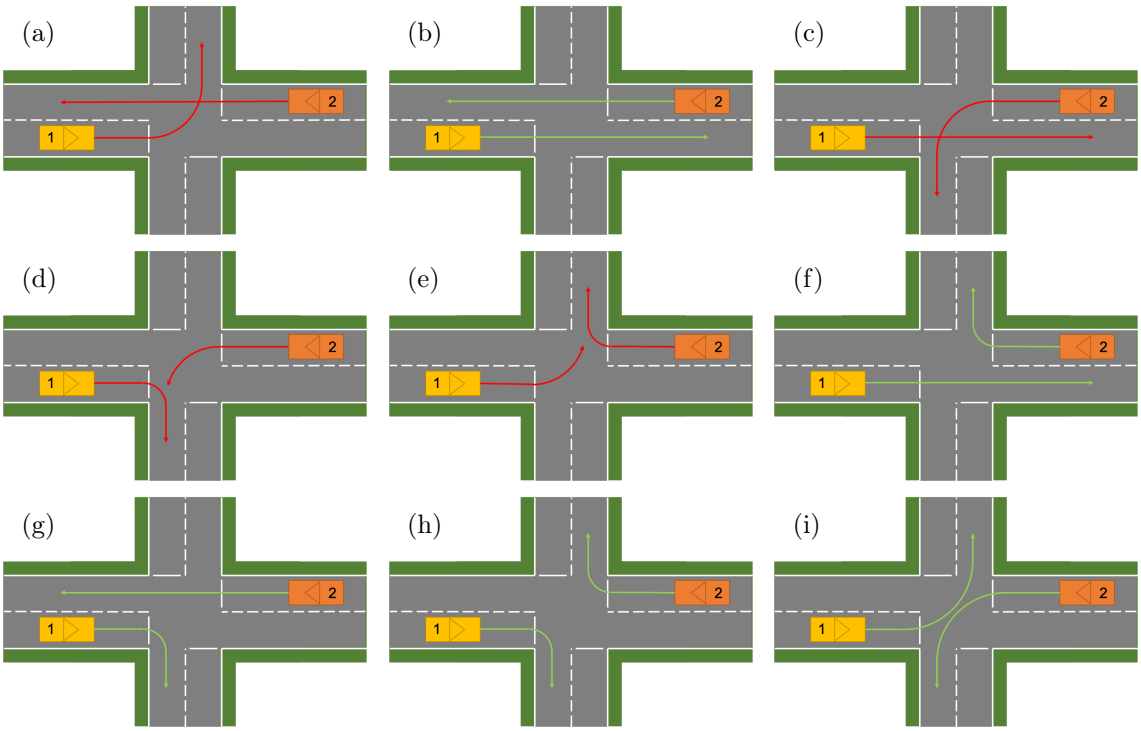


Figure 4.4: All nine possible scenarios with two vehicles at a four-way-intersection. The possibility of a U-Turn is not considered. Vehicle 1 is considered to be the ego vehicle.

intersect. These four scenarios all include left turns by one of the vehicles. Assuming that the vehicles drive according to public road regulations, in two out of these four trajectories a detection of Vehicle 2 by Vehicle 1 is safety critical. These two scenarios are shown in Figure 4.4 (a) and (e). In these cases, Vehicle 1 performs a left turn and has to give way to Vehicle 2. In Figure 4.4 (c) and (d) Vehicle 2 has to give way. Thus, the detection of Vehicle 2 is safety critical in two out of all nine scenarios.

4.1.2 Discussion

Scenario-based evaluation does not only consider the perception. As illustrated by Figures 4.2, 4.3 and 4.4, the scenario-based evaluation also takes the path planning of the automated vehicle into account and includes the actions and trajectories of surrounding traffic participants in the evaluation. Accidents do not occur in case the ego vehicle trajectory does not intersect with the trajectory of other traffic participants. This is also the case if the other traffic participants are not detected by the perception sensors. Therefore, scenario-based evaluation can provide an estimate of the risk introduced by an automated vehicle. The risk corresponds to the product of the rate of failures λ and the associated consequences C [8, 77, 88].

One may count as a consequence the number of accidents. Considering the vehicle constellation from Figure 4.3, the consequence of not detecting Vehicle 2 is an accident in the scenario from Figure 4.3 (a), thus, $C = 1\,\text{accident}$. The scenario from Figure 4.3 (b) leads to no consequences if Vehicle 2 is not detected, resulting in $C = 0\,\text{accidents}$. Assuming that the two scenarios are on average equally represented on public roads, the expected

value for the consequence is an accident half of the times the vehicle enters a roundabout if the perception is not working $E[C] = 1/2$ accidents. The expected value of the consequence $E[C] = 1/2$ accidents indicates that the scenario results on average every second time in an accident if the ego vehicle just follows its trajectory without taking other traffic participants into account.
The failure rate of the environment perception of the vehicle, which is directly related to its reliability, can be treated independently. Assuming the environment perception detects 99 instances out of 100 vehicle instances that are in a corridor of 30 m in length and 30 m in width in front of the vehicle, which covers the size of most roundabouts, one obtains a reliability of 99 %. A vehicle instance describes one occurrence of a vehicle in a frame. Moreover, an adequate underlying association measure is assumed here. A reliability of 99 % corresponds to a failure rate of 1 %.
Figure 4.3 only visualizes a single frame. Scenarios are not based on a single frame. However, there exists a final frame at which Vehicle 1 has to make a decision whether to enter the roundabout or to wait [89]. The risk introduced by automated Vehicle 1 in such a scenario is provided by $r = \lambda_{sys} \cdot C = 0.005$ accidents per time entering a roundabout).
Accident data for such specific road infrastructure scenarios is difficult to obtain. Recordings of a single scenario like entering a roundabout are often too sparse to determine a statistically valid error rate for this individual scenario. [90] investigates a dataset with 266 vehicle trajectories within a roundabout. The study, however, is not solely limited to entering the roundabout but also accounts for the exiting of the roundabout. Furthermore, an accident is not reported within the analysis.
[91] in comparison works on accident data and classifies the data into scenarios. However, scenarios in [91] specify the number of vehicles involved, the driving direction and the orientation of the vehicle relative to the driving lane instead of specific road infrastructure and a specific vehicle setup like the roundabout example. Scenarios without accidents do not exist in the analysis which does not allow a conclusion about the accident rate in a specific road infrastructure-based scenario.
[77] estimates the probability of safety-critical scenarios, so scenarios that lead to an accident, on German motorways using the HighD dataset [92]. The study is limited to situations on motorways where a vehicle stays in the same lane. Different situations correspond to a categorization of each vehicle into speed intervals. This approach categorically excluded lane-switching situations where the detection of neighboring objects also matters. [77] estimates the probabilities for different speeds and adds up the data using the total probability theorem with $\kappa = \sum_i p_i \cdot p_{S,i} = \sum_i Pr\left(Speed \in I_i\right) \cdot Pr\left(Scenario \in \{S2, S3\} \mid Speed \in I_i\right)$. $p_i = Pr\left(Speed \in I_i\right)$ describes the probability of driving with a speed in the range defined by the interval I_i; $p_{S,i}$ describes the conditional probability of having potentially dangerous scenarios in the speed range i. Potentially dangerous scenarios are defined as scenarios that can lead to collisions of S2 or S3 severity according to ISO26262 [77]. Based on an approximated mean time between failures (MTBF) of 1.3×10^5 h and the HighD dataset, which contains 150 h of drone recordings of German motorways, the study in [77] concludes that an FN rate sufficient for automated driving

should not exceed $1 \times 10^{-4}\,\mathrm{h}^{-1}$ for leading vehicles on the same lane on motorways.
Investigating safety-critical scenarios and estimating the probability of these scenarios is a research topic in itself. Further research scenario-based testing can be found in [93–96]. [93] presents a list of possible scenarios. Furthermore, within the scope of the PEGASUS project a database of relevant scenarios for the testing of automated vehicles was generated [94, 96].
Scenario-based testing performs a validation on the vehicle level rather than on the individual components of the vehicle like the perception. However, scenario testing stays always limited to recorded and/or defined scenarios [86, 93]. As indicated by the example situation in this section, the risk of individual situations where a misdetection may not directly lead to an accident is also represented in a statistical analysis of the environment perception.

4.2 Existing evaluation measures

Besides the probability of detection and the probability of false alarm, which are utilized by the model from [16] as introduced in section 2.5, other evaluation measures for the performance of sensors exist. The following section investigates evaluation measures utilized for the perception evaluation of automated vehicles.
The legibility of the evaluation is contingent upon not just the evaluation measure, but also the association measure. Thus, for the analysis of the evaluation measures we employ the IoU as association measure in the subsequent analysis.
Furthermore, the evaluation usually depends on the confidence score which is a measure provided by object detection algorithms to rate how certain an algorithm is about the presence of every detected object. An additional threshold τ for the confidence score is introduced. In order to indicate the additional threshold τ for each classification TP, FP and FN by the association, τ is used as a subscript in the equations.

4.2.1 Investigation outline

This work discusses different evaluation measures that can be used for the evaluation of object data that is associated with a reference truth. Object data that is associated with a reference truth is classified into TP, FN and FP cases based on the association measure. With regard to the release of automated vehicles, one requires a sufficiently reliable environment perception for the vehicle. Addressing a sufficiently reliable environment perception, there are two associated questions: first, how to evaluate the environment perception? To answer this question, we investigate the different evaluation measures. Second, based on the evaluation measure how could one determine that the perception is sufficient? We try to answer the latter question by classifying the evaluation measures into absolute and relative evaluation measures. Absolute evaluation measures can derive an actual error rate in failures per unit time that can be related to the failure rate of human driving whereas relative measures allow a comparison between sensor systems without allowing a determination of the failure rate that can be related to the vehicle's safety. This section proceeds

as follows:

- In the first step, it investigates if and how one can interpret the result of the evaluation measure. Moreover, it incorporates expectations obtained from human driving. The performance of human driving is often provided as fatalities or accidents per unit time of driving or per driven kilometers. For the release of automated vehicles, one needs to be able to obtain a measure that is understandable for the public in order to become acceptable.
- In the second step, the section evaluates whether the evaluation measure is better used as a relative measure or whether it might be usable as an absolute measure. The classification into a relative or an absolute measure correlates with the interpretability of the measure.

The subsequent section is subdivided into the following four parts: (1) the next section investigates individual association measures for the classification of the detection into correct and false according to the discussed procedure; the study continues with investigating different evaluation measures used for object detection evaluation in automated driving; (2) we investigate the combinations of association measures and evaluation measures while having a closer look at common combinations of the two; besides we discuss an approach that does not differentiate between classification and evaluation; (3) we discuss advantages and disadvantage and (4) we present a conclusion.

4.2.2 Investigation of existing evaluation measures

This section investigates different evaluation measures used in object detection for automated driving, which corresponds to the second part of the evaluation in Figure 4.1 besides the association. The evaluation measures accumulate FP and FN errors over the dataset and are supposed to provide information about the overall performance of the perception sensors. All evaluation measures investigated in this work are summarized in Table 4.1.

Recall

Section 2.5.2 introduces the recall value as defined in equation (2.1). One can derive the number of false negatives FN_τ from the recall value r_τ. Furthermore, one can estimate the average value for the number of objects $\mu_{o,\tau,t}$ of the specific class that are present in a data recording at the time point with index t and the number of data recordings n_t.

$$FN_\tau = (1 - r_\tau) \cdot (TP_\tau + FN_\tau) \approx (1 - r_\tau) \cdot \hat{\mu}_{o,\tau,t} \cdot n_t \tag{4.1}$$

The recordings are performed at a certain rate and the time between recordings is also referred to as a time frame which can, thus, be indexed by t. $TP_\tau + FN_\tau$ corresponds to the total number of objects present in all time frames. And the total number of objects present in all time frames can be approximated by the expected value of the average number of objects per time frame $\hat{\mu}_{o,\tau,t}$ times the number of time frames n_t. The number

Measure	Reference	Equation	Abbr.	Category
Recall	-	(2.1)	$\overset{p_\tau}{Pr}(O \mid D)$	absolut
Precision	-	(2.2)	$\overset{r_\tau}{Pr}(D \mid O)$	absolut
Average precision	[14, 15, 97] [60, 66, 98] [47, 53, 99]	(4.3)	AP	relative
AUC of the ROC	[60]	-	-	(relative)
distance weighted recall	[64]	(4.5)	r_D	relative
distance weighted precision	[64]	(4.5)	p_D	relative
distance weighted AP	[64]	(4.5)	APD	relative
nuScenes detection score	[22, 48]	(4.8)	NDS	relative
Average orientation similarity	[47]	(4.3)	AOS	relative
multi object tracking precision	[61]	-	MOTP	relative
multi object tracking accuracy	[61]	-	MOTA	relative
average multi object tracking precision	[67]	-	AMOTP	relative
average multi object tracking accuracy	[67]	-	AMOTA	relative
higher order tracking accuracy	[100]	(4.13)	HOTA	relative

Table 4.1: Evaluation measures used for object detection and object tracking algorithms.

of FN_τ errors per unit time correspond to $(1 - r_\tau) \cdot \hat{\mu}_{o,\tau,t}/\Delta t$ where Δt is the time difference between two subsequent time frames.
In conclusion, recall corresponds to our definition of an absolute evaluation measure as the error rate can be determined from the number of FN_τ for a specific period of time.
Recall has been considered for the reliability analysis of the vehicle's environment perception in previous studies where it was also referred to as the probability of detection (POD) [8, 16]. As an interpretable evaluation measure, recall or actually the derived error rate can be compared to human driving.

Precision

Section 2.5.2 introduces the precision in equation 2.2. The precision provides a value for how reliable an object detection is and how often ghost objects appear. In parallel to the concept of recall, a value close to 1 indicates a good agreement of the detections with the reference. All objects that are detected by the object detection algorithm are correctly identified and are indeed present. The number of false positive occurrences FP_τ within a certain time interval can be evaluated using the precision p_τ

$$FP_\tau = (1 - p_\tau) \cdot (TP_\tau + FP_\tau) \approx (1 - p_\tau) \cdot \hat{\mu}_{d,\tau,t} \cdot n_t \tag{4.2}$$

Here, the expected average number of detections $\hat{\mu}_{d,\tau,t}$ is considered instead of the expected average number of present objects $\hat{\mu}_{o,\tau,t}$.
Consequently, precision is an absolute measure evaluating the number of objects that were correctly identified versus the total number of objects. With an average of present vehicles in common driving scenarios, the number of errors per unit time or unit distance can be estimated.
Recall and precision are usually presented together as one measure alone does not ensure a good performance of an object detection algorithm. For example, a high recall can be obtained when the object detection algorithm recognizes many objects independent of whether the algorithms also detect many ghost objects. Therefore, the aim is to achieve a high recall as well as a high precision.

Average precision

The average precision (AP) is the most common measure for evaluating the performance of object detection algorithms once the detections are classified into TP, FP and FN by the association measure [14, 15, 47, 48, 53, 60, 66, 97–99]. For the evaluation of the AP, the confidence score for every detection is utilized besides the association measure in order to evaluate the goodness of the detections. While the number TP_τ and FP_τ can increase with an increasing threshold τ, FN_τ will decrease with an increase in τ for an algorithm that is based on a regression of the confidence values. The AP combines recall and precision in one measure and corresponds to the area under the precision-recall curve.
For the AP, precision and recall are evaluated at different confidence scores τ which corresponds to the functions $\tau \mapsto p(\tau) = p_\tau$ and $\tau \mapsto r(\tau) = r_\tau$. The average precision makes

use of the inverse function of $r(\tau)$ to eradicate the use of the confidence threshold in the evaluation. This provides a relation between precision and recall $r \mapsto p(r)$. An estimate of the AP is obtained by averaging over the interpolated precision values at certain recall levels. For interpolation, the maximum precision for any recall value higher than the recall level of evaluation is used [15].

$$AP = \frac{1}{N} \sum_{r \in S} \max_{\tilde{r}:\tilde{r} \leq r} \left(p\left(\tilde{r}\right) \right) \tag{4.3}$$

Here, S is the set of equally spaced recall levels with cardinality N. The AP may also be approximated using all calculated recall levels instead of only a set of equally spaced recall levels.

Some studies evaluate the AP for every class of objects individually [14, 52]. A comparison is performed for the individual object classes. The best object detection algorithm for a specific class of objects is the one that yields the highest AP. And the best overall object detection algorithm is the one that yields the highest AP for most classes as used in [14, 52].

The higher the AP for a certain class, the better the detection of objects for the specific class. An AP close to 0 means that independent of the confidence score threshold τ only very few objects are detected correctly. In comparison, an AP close to 1 means that nearly all objects are detected independent of the chosen confidence threshold τ. AP values between 0 and 1 show a dependence between precision and recall. The negative correlation between precision and recall is introduced by the chosen confidence threshold τ, as both, precision and recall depend on the confidence threshold. The number of correct detections depends on the confidence threshold τ as the recall tends to increase with a lower chosen confidence threshold τ as more objects are detected while the precision tends to decrease with a lower confidence threshold τ as usually more FP_τ are observed. The introduction of novel object objective detection algorithms entails a progressive elevation of the AP by aiming to achieve values closer to 1 which allows a comparison between different object detection algorithms and rating them relative to each other. However, the AP does not provide a value that can be related to a specific failure rate as the failure rate can vary with the chosen confidence threshold. The AP, however, does not rely on the confidence threshold τ.

The confidence threshold τ may still be use case dependent and may be chosen such that it optimizes the object detection output depending on whatever error is more relevant, e.g., in case FP is more relevant the confidence threshold τ may be increased and if FN errors are more relevant the threshold τ may be decreased. Choosing a specific confidence threshold τ is usually restricted to a specific training data set together with a specific association measure but may not be applicable to other datasets or association measures. The independence on the confidence threshold τ of the evaluation of the object detection is, therefore, beneficial. However, the question remains whether the AP can also be used as an absolute measure that is interpretable for the comparison to human driving which is taken into account when talking about the release of automated vehicles.

Figure 4.5 shows an example precision-recall curve. Precision and recall correspond to

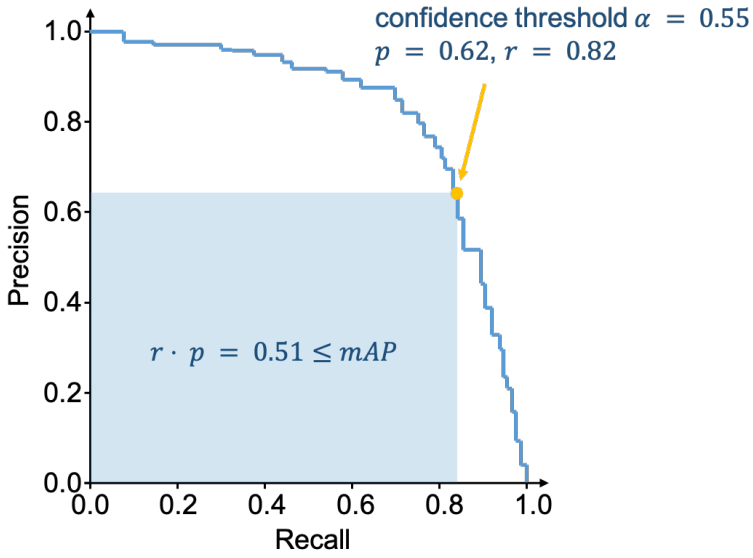


Figure 4.5: Example precision recall curve $\max_{\tilde{r}:\tilde{r}\leq r}(p(\tilde{r}))$. (4.3) corresponds to an approximation of the curve's integral. An even more basic approximation corresponds to the shown rectangle which corresponds to the multiplication of a certain precision-recall pair. A precision-recall pair is obtained for a certain confidence threshold τ. The multiplication of precision and recall corresponds to the multiplication of two conditional probabilities $Pr(O \mid D) \cdot Pr(D \mid O)$.

the two conditional probabilities $Pr(O \mid D)$ and $Pr(D \mid O)$, respectively. The intention is that both conditional probabilities approximate one for a reliable environment perception. Thus, the product of the two should also tend towards one. However, a product of the two conditional probabilities, which is exemplary indicated by the rectangle in Figure 4.5, does not allow any relation to the actual failure rate. Neither does the integral of one conditional probability in dependence on the other.
The actual number of errors corresponds to the additive measure $FN_\tau + FP_\tau$ instead of a multiplication or integration of precision and recall. As the number of errors per unit time or per distance is usually a measure taken for comparison, we consider an additive measure as more appropriate for the evaluation of the perception reliability of the automated vehicles such that it can be used as justification for their release.

Average precision for multiple object classes

Some references take the mean of all AP values obtained for the different classes, which is commonly referred to as mean average precision (mAP) [51].
In order to obtain a measure that becomes independent of the chosen threshold α for the association measure, other references average also the mAP obtained for different association thresholds α [53, 98]. The different approaches to evaluate the mAP are not consistently separated by distinct acronyms in the literature [98].
The aforementioned mAP fails to adequately address the issue of data imbalance among classes, as it assigns equal weight to all classes. Consider the following example: a class C_1 of objects with only a few instances occurring in the dataset and a class C_2 of objects with

many instances occurring in the dataset. The resulting mAP would be the same in case (a) all objects of class C_1 are observed and none of class C_2 or (b) no objects of class C_1 are observed but all of class C_2. The overall number of objects detected would be low in case (a) while it is high in case (b). To account for this problem [97] proposes to pool all objects from all classes and combine them in a single precision-recall curve which is used to evaluate the average precision AP^{Pool}. In AP^{Pool} all object instances are weighted equally rather than all object classes with different numbers of observed objects. However, like AP, AP^{Pool} lacks the relation to an absolute number of errors that allows the comparison with human driving capabilities. It is, therefore, also only applicable as a relative measure.

Association measure threshold independent average precision

While the AP measure is independent of the confidence threshold τ, it still relies on a chosen threshold α for the association measure. Many references use a predefined threshold α in combination with the association measure IoU as described in section 3.1 [14, 47] . However, with a single IoU threshold α, one encounters the issues described in section 3.1. An AP close to 1 would still not account for imperfections in the detection of the position of objects introduced by the association measure.

Distance-dependent average precision

The distance weighted average precision (APD) is proposed by [64]. The APD is based on inverse distance-weighted TP, FN and FP counts. For a set of detections, $B = \{b_i\}, i = 1, \ldots, N$ and a set of reference truth objects $G = \{g_i\}, i = 1, \ldots, M$ these counts are defined by the sum of the inverse distances to the power of a hyperparameter β that controls how much the distance is taken into account.

$$
\begin{aligned}
IDTP &= \sum_{b_i \in \mathbb{TP}} \frac{1}{d^{\beta}_{g(b_i)}}, \quad IDFP = \sum_{b_i \in \mathbb{FP}} \frac{1}{d^{\beta}_{b_i}} \\
IDG &= \sum_{g_i \in \mathbb{G}} \frac{1}{d^{\beta}_{g_i}}
\end{aligned}
\tag{4.4}
$$

The distance d corresponds to the Manhattan distance between the center point of the ego vehicle and the center point of the detection b_i or reference object $g_i / g\left(b_i\right)$. $g\left(b_i\right)$ denotes the reference truth object that is associated with the detection b_i. The distance weighted recall and precision are derived from the inversely distance-weighted TP, FP and IDG values.

$$
r_D = \frac{IDTP}{IDG}, \quad p_D = \frac{IDTP}{IDTP + IDFP} \tag{4.5}
$$

The APD is derived from the precision-recall curve while using p_D and r_D as precision and as recall, respectively [64].

Compared to any other evaluation measure in this section, the APD is the only measure that takes into account that objects that are further away from the ego vehicle may be

less relevant for the trajectory planning of the ego vehicle if not detected. The averaging, however, still allows that a high recall value r_D can be observed even though an object right in front of the ego vehicle may not be detected while many objects in the far distance are observed. Therefore, from the APD itself, one cannot conclude whether a few errors in the near distance occur or whether many errors in the far distance are observed.
Moreover, as the APD averages over the precision values p_D in dependence of the recall values r_D, one cannot know the specific confidence score threshold τ that yields the best precision-recall pair for the intended application with the used object detection algorithm. Likewise, to the AP, from the APD one can only extract which object detection algorithm performs best on average. Therefore, APD is also a measure that allows a relative comparison between different object detection algorithms. However, one cannot conclude an absolute number of critical situations per unit time or per distance. A high APD value only indicates that either a few errors in the close distance are observed or many in the far distance. Depending on the scenario, far-distance objects may also be relevant, e.g., on the motorway.

Average recall

The average recall (AR) is another measure used for the evaluation of object detection algorithms. Unlike AP, AR does not take the confidence score τ into consideration. Instead, AR represents the average value of recall values obtained for differently chosen values of the IoU threshold α [51, 101].

$$AR = 2 \int_{0.5}^{1} r(\alpha) \, \mathrm{d}\alpha \tag{4.6}$$

For the evaluation of the COCO challenges AR is defined slightly different. It is defined as the average of the maximum recall values for the defined set of 10 IoU thresholds $[0.5, 0.55, \cdots, 0.95]$. The evaluation of AR based on the set of predefined IoU thresholds provides an approximation of the AR from equation (4.6) [101].
Unlike AP, which incorporates the recall values by the integration of the precision in dependence of the recall, AR is an individual measure that does not incorporate the precision. However, it accounts for different IoU values and tends to be better for increasingly confident fits. Averaging over different IoU thresholds α, however, does not allow an extraction of the best IoU value for a specific application. It may, therefore, be used as a relative measure but, in case a specific decision about the objects' state has to be made, which requires the use of a threshold α, AR does not provide any information about the best choice of the IoU threshold α. At some point, it may be used as an absolute measure though. One knows that there exists an IoU threshold α for which the recall is at least as large as the AR. Therefore, by using AR in combination with equation (4.1) one can estimate the minimum number of possible FN failures. However, from AR one cannot conclude the IoU threshold α for which this number of FPs is achieved.
In the COCO challenges the term AR also incorporates averaging over the different classes [98]

while others refer to the averaging over the different classes as (mean average recall (mAR)). In case of data imbalance, averaging over multiple classes is in favor of object classes with only a few object instances. For object classes with many object instances, the object detection algorithm has to properly detect many objects in order to achieve a similar mAR while only doing few mistakes in the object class with few objects.

Receiver operator characteristic

The receiver operator characteristic (ROC) is an alternative to the precision over recall curve, which is used for the evaluation of AP. The ROC represents the TP rate in dependence on the FP rate. The TP rate corresponds to recall while the FP rate, also defined as PFA, is defined as $FP/(FP+TN)$. As a high recall and a low FP rate are preferred, the intention lies in finding algorithms that achieve values in the upper left corner of the diagram.
In case the dataset can be subdivided into a number of cases where an object is present and into a number of cases where no object is present, the ROC can be directly transferred into the precision-recall curve or vice versa [60]. A possible definition for the TN cases was introduced in [70] and [71] by subdividing the FOV into a grid. [60] demonstrates that a curve dominates in ROC space if, and only if the curve dominates in precision-recall space. Furthermore, the ROC allows to create a convex hull by linear interpolation between the points on the ROC. This is explained by the fact that an average TP rate and a FP rate between two classifiers can be achieved by randomly choosing between one or the other classifier with weighted probability. In this context, the term classifier is used for a differently chosen confidence score threshold τ. The precision-recall curve does not allow a linear interpolation as shown by [60]. In the case of the precision-recall curve, linear interpolation leads to an overestimation. The approximation with a step function by using the maximum precision at any higher recall value in equation (4.3) returns a lower approximation. The correct ideal approximation of the precision-recall curve requires the transformation from the linear interpolation in the ROC.
The ROC obtains the same information as the precision over recall curve in case TN cases are defined. In addition, it relies on the definition of the TN cases. A common practice in object detection is to not take the TN cases into account to avoid associated data imbalance between regions with and without objects. As mentioned previously, [70] and [71] introduce a definition for TN cases by subdividing the bird-eye view perspective of the FOV of an automated vehicle into a grid. However, this approach lacks a proper interpretation due to the fact that the chosen grid size is a freely choosable parameter that provides no clear definition except that it is required to be in the order of magnitude of the considered objects.

Area under the receiver operator characteristic

Like for AP the area under curve (AUC) of the ROC is intended to reach values close to 1. An algorithm that yields results with higher AUC of the ROC is, therefore, preferred.
The AUC of the ROC is not suitable for the evaluation of highly skewed datasets [60]. As

surrounding objects usually cover only a small area of the FOV of the sensors, the sensor data commonly corresponds to a skewed dataset between regions with and without objects. Subsequently, a proper definition of TN cases for object detection in automated driving does not exist. Furthermore, the reference and the object detection algorithms only yield objects that are present or detected, which only incorporates FP, FN and TP cases but no TN cases. And even with a definition of the TN cases, the AUC of the ROC remains a relative measure.
To sum up, the discussion in the previous section about the limitations of the ROC in the evaluation of object detection can also be applied to the AUC of the ROC. Without a proper definition of the TN cases, the resulting AUC of the ROC will remain difficult to interpret. Besides, also in case of a proper definition of TN cases, which is for example achieved in semantic segmentation, the AP is commonly the preferred measure as the AUC-ROC does not account for the imbalance between the actual positive and the actual negative cases according to the reference truth.

True positive measures

TP measures average over the difference between TP detections and their corresponding reference objects [47, 48, 61, 102]. The nuScenes challenges based on the nuScenes dataset introduce a set of five TP measures to account for multiple attributes in object detection [48]. The nuScenes set of TP measures $\mathbb{TP}$ includes:

- The average translation error (ATE) is the average of the 2D Euclidean center distance in the bird-eye view perspective. The multi object tracking precision (MOTP), used in [61, 102], is the average value of the association measure for all TP detections. It is equivalent to the ATE when using the Euclidean distance as an association measure. In comparison to the ATE, [102] proposes to use the MOTP with different association measures. For face, person and vehicle tracking in 2D images [102] proposes to use the IoU instead of the Euclidean center distance.

- The average scale error (ASE) is the average of the 3D IoU after correction of the orientation and the translation.

- The average orientation error (AOE) is the average difference in the orientation of the detection and the orientation of the assigned reference. [47] provides a comparable measure that is called the average orientation similarity (AOS).

- The average velocity error (AVE) is the average of the absolute velocity provided by the L2 norm of the difference in the velocity between detection and reference object.

- The average attribute error (AAE) averages the result of 1 minus the attribute classification accuracy.

The measures are only applied on the TP detections, so all detections that are associated with a reference truth object. The mean over all classes is evaluated for each of the five measures.

$$mTP = \frac{1}{\mathbb{C}} \sum_{c \in \mathbb{C}} TP_c \tag{4.7}$$

Classes, where attributes are not well defined, are omitted in the evaluation of the mean in nuScenes' TP measures. The evaluation measures, which can be larger than 1 (mean average velocity error (mAVE), mean average orientation error (mAOE), mean average translation error (mATE)), are normalized to the range of 0 to 1 [48].

In the case of the ATE and the MOTP, the chosen threshold α for the association measure defines a maximum possible (Euclidean) distance between detections and their reference counterpart. Association measures specific limitations as described in section 3.1 remain as shown by the following two examples:

An average of any association measure does not provide a measure that allows a statement about the vehicle's safety in terms of object detection errors per unit time or per distance. A MOTP value only allows a relative comparison: Depending on the chosen association measure, larger or smaller MOTP values show an improvement on average. E.g., in the case of the Euclidean center distance: the smaller the ATE/MOTP the closer the object centers of detection and reference on average which indicate a better fit only of the center points.

A variation of MOTP that is independent of a fixed confidence threshold τ exists. The average multi object tracking precision (AMOTP) is achieved by averaging over multiple confidence thresholds τ [67]. Like MOTP, AMOTP is difficult to interpret as good and bad fits average to medium performance. However, as demonstrated in Figure 4.7 one cannot exclude potentially hazardous situations from a medium performance or even a good performance in MOTP. The same applies to AMOTP. Further averaging over different confidence values decreases the interpretability as one no longer knows the performance for a fixed confidence threshold τ. A lower AMOTP value can be lower for some MOTP values while being higher for others. In the case where the threshold τ is relevant, e.g., when a decision is required for the upcoming time step, the optimum threshold cannot be derived from the AMOTP measure. The AMOTP measure allows a relative comparison as a lower AMOTP value indicates that the MOTP value is lower either for more recall values of much lower for individual recall levels. However, it does not allow to conclude what confidence threshold τ may be suitable. Like MOTP, AMOTP does not provide an error rate that can be translated to human driving. Thus, AMOTP is not suitable for the use in the reliability analysis of the environment perception and the release of automated driving and only allows a relative comparison of object detection algorithms.

Except for the ATE, MOTP and AMOTP, all other TP measures are not based on an association measure. These measures only compare individual state variables of objects. They are not suitable as individual measures for evaluating the environment perception performance of sensor and object detection algorithm [47]. There is no transformation that allows to associate the measures with the number of errors in perception that can possibly lead to an accident. Thus, they cannot be considered as absolute measures and only allows to draw relative conclusions.

nuScenes detection score

In order to also account for the suitability of the associated objects, the nuScenes detection score (NDS) summarizes the five TP measures and the mean average precision [48] in a single measure.

$$NDS = \frac{1}{10}\left[5mAP + \sum_{mTP \in \mathbb{TP}} \left(1 - \min\left(1, mTP\right)\right)\right] \tag{4.8}$$

As the NDS incorporates the AP, we refer to the discussion of the AP concerning its interpretability. We classify the NDS as a relative evaluation measure as the AP corresponds to a relative evaluation measure. Like the AP, the NDS does not allow a conclusion about an absolute number of dangerous scenarios per unit time or unit distance as required for the release of automated vehicles. Taking the mean of the mAP and the five TP measures does not allow a subdivision in the number of occurrences of the individual types of errors from the NDS. The resulting evaluation measures become, therefore, less interpretable. The reduction of all individual types of errors is still pursued. Thus, an increase in the NDS is preferred and shows an overall tendency towards the reduction in any type or all types of errors. However, an increase in the NDS does not allow a conclusion about the way in which the object detection results improved.

Multiple object tracking accuracy

Besides MOTP, the multi object tracking accuracy (MOTA) is proposed to be used in automated driving by [61] to also account for the consistency of the object detections in time.

MOTA summarizes the FP rate, the FN rate and the rate of mismatches, in one measure [61, 67, 103, 104]. While the FN and the FP rate can be in the interval of $[0, 1]$, the rate of mismatches lies in the interval between 0 and the maximum ratio of mismatches $\overline{mme}_{max}$. The maximum ratio of mismatches $\overline{mme}_{max}$ is defined by the maximum number of time steps $n_{t,max}$ for which the maximum number of objects $n_{o,max}$ is present.

$$\overline{mme}_{max} = \frac{\left(n_{t,max} - 1\right)\left(n_{o,max} - 1\right)}{n_{t,max} \cdot n_{o,max}} \tag{4.9}$$

Therefore, the MOTA returns a number in the interval $(-(1 + \overline{mme}_{max}), 1]$. From MOTA one can derive the total number of errors in all n_t time frames analog to equation (4.1) when an estimate for the number of objects $\hat{\mu}_{o,r,t}$ in each frame is available. However, the type of error remains unknown and can correspond to any of the three types. The question that needs to be addressed is whether the error caused by a FP, a FN and or by a mismatch can be weighted equally. Like the definition of the appropriate thresholds α for the association as discussed in section 3.1, the correct weighting of different types of errors remains an open question. Moreover, the number of errors is dependent on the definition of an error that corresponds to the chosen association. Additionally, the errors are weighted equally, independent of the distance from the ego vehicle. Thus, a mismatch of two objects which are 50 m away from the ego vehicle is weighted the same as a mismatch of two objects

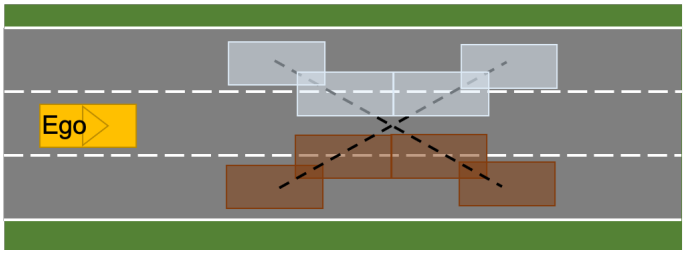


Figure 4.6: Example for a mismatch in the trajectories of two objects. Such a switch in the object IDs is weighted the same as a FP and an FN error by the MOTA measure.

which are 5 m away from the ego vehicle. However, a mismatch while tracking multiple objects is not as relevant as a FP error or a FN error. For example, one can consider two vehicles in front of the ego vehicle on the motorway as shown in Figure 4.6. The mismatch in the two object trajectories may not have a direct influence on the ego vehicle as, in any way, it needs to keep a distance from the two other vehicles. However, a FP in front of the ego vehicle may lead to an accident due to unexpected emergency braking. Based on these considerations MOTA can only be utilized as a relative evaluation measure for comparing object-tracking algorithms.

Average multi-object-tracking accuracy

MOTA does not consider the confidence score of the object detections which are provided by the object detection algorithms. Thus, MOTA are evaluated for a specifically chosen confidence threshold τ. Objects below the chosen threshold τ are neglected. In order to obtain a measure that is independent of a fixed confidence threshold τ, [67] proposes the AMOTP and the average multi object tracking accuracy (AMOTA). Like the AP and the AOS, the AMOTP and the AMOTA provide an average value for the MOTP and the MOTA for different recall levels.
AMOTA corresponds to the average value of the MOTA values obtained at different recall levels. By taking the average of multiple MOTA values for different recall values r one also loses interpretability in AMOTA. Like in AMOTP, the additional averaging does not allow to specify which confidence threshold τ yields better results. However, the confidence threshold τ may be necessary for a decision in path planning, making it necessary for the reliability analysis. An error in the detection can cause an accident due to subsequent incorrect planning. Besides, AMOTA faces the same difficulties as MOTA: it does not differentiate between the three different types of errors, FP errors, FN errors and mismatches. A small AMOTA value can be due to too many errors of one type or due to errors of all types with a lower incidence of each individual error type. Thus, AMOTA may be used as a relative measure but is not sufficient as an absolute measure that can be utilized for the reliability analysis of automated vehicles.

Scaled multi-object-tracking accuracy

[67] proposes an additional measure that they introduce as scaled multi object tracking accuracy (sMOTA) and scaled average multi object tracking accuracy (sAMOTA), respec-

tively. Commonly used measures, in particular the AP, return a value in the interval $[0, 1]$. In comparison, the AMOTA reveals at most a value of 0.5 as the value of MOTA is smaller than the recall r_τ. In order to normalize the AMOTA into the interval $[0, 1]$, the sMOTA is introduced.

$$sMOTA = \max\left(0, 1 - \frac{\sum_t \left(m_{t,\tau} + fp_{t,\tau} + mme_{t,\tau}\right) - (1-r) \cdot \sum_t g_t}{r \cdot \left(\sum_t g_t\right)}\right) \tag{4.10}$$

The notation is adapted from [61]: $m_{t,\tau}$, $fp_{t,\tau}$ and $mme_{t,\tau}$ are the number of FN, the number of FP and the number of identity switches within a single time frame t with are accumulated over all frames by summing them up. The numbers of FN and FP accumulated over all time frames of a dataset are represented by capital notation.
Besides the subtraction in the nominator, the number of TP detections is used in the denominator rather than the total number of reference truth objects as $r \cdot \sum_t g_t = r \cdot (TP_\tau + FN_\tau) = \text{TP}/\left(TP_\tau + FN_\tau\right) \cdot (TP + FN) = \text{TP}$. sAMOTA is the average of the $sMOTA_{r_\tau}$ values at different recall levels r analogue to the relation between AMOTA and $MOTA_{r_\tau}$.
By normalizing a measure into the range $[0, 1]$ the measure itself does not become easier to interpret. The limitations of MOTA remain for sMOTA. The sum of all missed objects is equivalent to the number of FNs $FN = \sum_t m_{t,\tau}$. The subtracted term in the nominator of the fraction also corresponds to the number of FNs as

$$(1 - r_\tau) \cdot \sum_t g_t = \left(1 - \frac{TP_\tau}{TP_\tau + FN_\tau}\right) \cdot (TP_\tau + FN_\tau) = FN_\tau.$$

Thus, one can reduce equation (4.10) by the term which describes the number of FPs.

$$sMOTA = \max\left(0, 1 - \frac{\sum_t \left(fp_{t,\tau} + mme_{t,\tau}\right)}{r_\tau \cdot \left(\sum_t g_t\right)}\right) = \max\left(0, 1 - \frac{FP}{TP} - \frac{\sum_t \left(mme_{t,\tau}\right)}{TP}\right) \tag{4.11}$$

In conclusion, sMOTA proposed by [67] does not consider the FN errors.
The ratio of the number of FPs and the number of TPs can be is related to the precision.

$$\begin{aligned} \frac{1}{1 + FP_\tau/TP_\tau} &= \frac{1}{TP_\tau/TP_\tau + FP_\tau/TP_\tau} \\ &= \frac{1}{\left(TP_\tau + FP_\tau\right)/TP_\tau} \\ &= \frac{TP_\tau}{TP_\tau + FP_\tau} = p_\tau \end{aligned}$$

The precision is a measure that can be transformed into the number of FPs as described in equation (4.2). Using the relation between the precision and the ratio of the number of

FPs and the number of TPs one can further rewrite equation (4.11).

$$sMOTA = \max\left(0, 2 - \frac{1}{p_\tau} - \frac{mme_{t,\tau}}{r_\tau \cdot (\sum_t g_t)}\right) \tag{4.12}$$

While precision allows an interpretation with respect to the automated vehicle's safety, the summation of the inverse of the precision and the number of identity switches only allows a relative comparison. sMOTA should tend towards 1. However, from the sMOTA one cannot conclude the number of individual types of errors.
Furthermore, the inverse of the precision tends towards infinity when the precision tends to zero. However, sMOTA is limited by the lower bound of 0 due to the max function used in equation (4.11). Cutting the value does not reduce the actual number of errors, it just increases the value of the sMOTA measure versus the MOTA measure. However, cutting of sMOTA at a minimum of 0 further reduces the interpretability as the error rate may be worse than actually expected.

Higher order tracking accuracy

[100] propose the higher order tracking accuracy (HOTA) for the evaluation of the tracking accuracy. HOTA is independent of the association threshold α which is obtained by taking the mean of the measure $HOTA_\alpha$ for different association thresholds α.

$$HOTA_\alpha = \sqrt{\frac{\sum_{c \in \mathrm{TP}_\alpha} \mathcal{A}_\alpha\left(TP_i\right)}{|TP_\alpha| + |FN_\alpha| + |FP_\alpha|}} \tag{4.13}$$

The association score $\mathcal{A}_\alpha\left(TP_i\right)$ is defined by [100] as the Jaccard index for the trajectory of a TP detection TP_i at a specific time point.

$$\mathcal{A}_\alpha\left(TP_i\right) = \frac{|TPA_\alpha\left(TP_i\right)|}{|TPA_\alpha\left(TP_i\right)| + |FNA_\alpha\left(TP_i\right)| + |FPA_\alpha\left(TP_i\right)|} \tag{4.14}$$

TPA, FNA and FPA are defined for the trajectory of an associated TP detection. For an TP detection in a time frame t, TPA are the number of times the detection trajectory and the reference trajectory are in agreement. FNA are the object instances at time frames where the reference trajectory of the TP detection does not correspond to the detection trajectory. FPA are the object instances at time frames where the detection trajectory of the TP detection does not correspond to the reference trajectory.
In case only a single object is present $HOTA_\alpha$ reduces to the Jaccard index $\frac{|TP_\alpha|}{|TP_\alpha|+|FN_\alpha|+|FP_\alpha|}$.
HOTA does not incorporate the confidence of the object detection and tracking algorithms. It is, therefore, required to set a defined threshold τ in case the object detection algorithm yields confidence values.
HOTA is in the range of $[0, 1]$ by default and does not require an artificial cut-off like sMOTA. It indirectly incorporates identity switches/mismatches mme by the association measure $\mathcal{A}$ as an identity switch either leads to an FNA and/or an FPA. The Jaccard index, which is used for evaluation of the trajectories in HOTA, is a frequently used measure due to its interpretation as a percentage of agreement between two sets, here the set of reference truth objects and the set of detected objects.

Integrating over the association threshold α, however, does not allow a conclusion about any specific association threshold that might be best for the intended application. For this case $HOTA_\alpha$ can be used. In addition, HOTA and $HOTA_\alpha$ do not allow a distinction between errors in tracking and errors in detection. A lower HOTA value indicates that either there are more FP or FN detections or the detection might be good but the tracking is quite bad, leading to a low value of $\mathcal{A}$. As a result, the suitable measure depends on the specific use case and cannot be summarized in HOTA. For automated vehicles, the detection of the surrounding objects may be more relevant for safe driving than proper tracking over time [100]. For automated driving the weighting of the tracking in HOTA is not necessary for the use case of automated driving. The egocentric perspective of an automated vehicle is also not considered in the HOTA. A detection error for an object that is far away is, therefore, weighted the same way as a detection error for an object right in front of the ego vehicle. While an error in the detection of a faraway object may not be relevant to the automated vehicle, an error for a nearby object may be crucial.

4.3 Combining association measure and evaluation measure

The evaluation of a dataset can be performed using any combination of an association measure and an evaluation measure. Any of the 7 association measures from Table 3.1 could be combined with any of the 13 evaluation measures that are listed in Table 4.1.
Based on the same association measure, all evaluation measures are in agreement in case the object detection algorithm performs perfectly. However, for imperfect object detection algorithms, the individual errors are weighted differently by the different measures.
The evaluation should be interpretable such that one can relate the resulting value of the evaluation measure to the human error rate which is a common census in the discussion about the release of automated vehicles [4, 20]. In case one cannot interpret either the association measure or the evaluation measure or both, the result remains non-interpretable. As an example, one can consider the combination of the IoU with a threshold of $\alpha = 0.7$ in combination with the precision as an evaluation measure. In case a precision close to one is observed, one can conclude that nearly every detection corresponds to a real object within the limits of the IoU given by the threshold of $\alpha = 0.7$. Here, the higher the value, the better. However, even for the limit that the precision reaches the value of one, one cannot ensure that none of the situations from Figure 3.2 occur. Especially in the case of large objects like trucks, a crucial deviation in the detection may be observed while the detection is still classified as correct detection. The release of automated vehicles requires to account also for such corner cases. By using the IoU in the evaluation, corner cases as shown in Figure 3.2 can occur while the evaluation measure indicates perfect agreement. The example demonstrates that the selection of the association measure plays an important role in the interpretability of the resulting evaluation.
The resulting evaluation can also be limited by the chosen evaluation measure. As an example consider the AP as an evaluation measure in combination with the IoU with a threshold of $\alpha = 0.7$ as association measure. A higher AP value is beneficial as the AP corresponds to the integral of one conditional probability, precision, depending on another

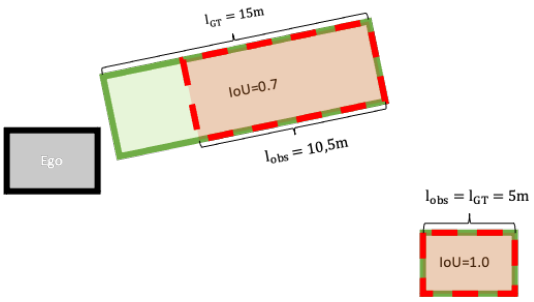


Figure 4.7: Example for a situation where all evaluation measures are not capable of indicating the misdetection when using the IoU with an association threshold of $\alpha = 0.7$ as association measure. The reference truth objects are shown in green and the detections are shown in red with a dashed contour. The ego vehicle is shown in black.

conditional probability, recall. The higher both conditional probabilities are, the better, thus it is aimed at a value close to one, meaning both conditional probabilities are high for all confidence values τ. However, the integral of precision in dependence on recall does not specify a value that can be interpreted in the sense that the result can be compared with the rate of accidents in human driving. In the previous example, one is able to relate the precision to the number of errors within the limitations of the IoU with a threshold of $\alpha = 0.7$. For the AP one cannot conclude the number of errors per unit time or per distance even within the limitations defined by the IoU with the threshold of $\alpha = 0.7$ as the AP itself cannot be referred to a number of errors as the number of errors is dependent on a chosen confidence threshold τ. The AP, however, averages over all precision values in dependence of recall which are obtained for different confidence values τ.
Figure 4.7 visualizes an example scenario for illustration purposes of the MOTP combined with the IoU. The evaluated value for MOTP is equal to 1.125 m. 1.125 m may be a good value for faraway objects. The distance and the associated criticality of the deviation are, however, not included in MOTP. Moreover, 1.125 m is just an average value. As shown in the example scenario from Figure 4.7 the deviation between the detection and the reference truth object can be much larger (4.5 m in this example). Especially, for close objects this can be critical as demonstrated in Figure 4.7. Similar examples can be found in any other combination of association and evaluation measures.
An introduced threshold in the evaluation is usually difficult to interpret as the selection of a threshold is often associated based on the use case, e.g., parking assistance where a false negative may be more acceptable than a continuous warning by the assistance system. In order to overcome the need for a specific threshold, a common approach is to average over a range of thresholds. However, for a specific use case, a threshold is usually required and an improved mean does not necessarily result in a better detection for a specific use case. Therefore, taking the mean value over a range of thresholds rather reduces the interpretability of evaluation measures. Almost all measures do not correspond to an absolute measure that could be compared with human driving.
A differentiation between the object-centric perspective and the ego-centric perspective is introduced in [64]. The distance-weighted average precision for the evaluation of ob-

ject detection algorithms in automated driving is introduced in [64] to account for the criticality of detection errors of nearby objects and objects that are far away. Previous object detection evaluations in image processing partition the objects in small, medium and large-sized objects to compare the performance of the algorithms for different object sizes which usually correlates with the distance [51].
So far, distance-weighting has only been introduced in the evaluation measure. The distance-weighting in the evaluation measure does not allow to differentiate between many detection errors in the far distance or very few detection errors nearby while the latter could be of more importance. Instead of introducing distance-weighting in the evaluation measure, one could also use an association measure where the acceptable tolerance increases with the distance. Besides the distance-weighted error definition presented in sections 3.4 and 3.5.5, one could for instance adjust the threshold for the IoU linearly with the distance. Similar attempts for the IoU are presented in [65]. An introduction of distance-weighting may account for the relevance of objects at different distances and, thus, may lead to a more intuitive evaluation whose values can be better interpreted.
In conclusion, the interpretability is lost or limited by either the result of the association measure, the evaluation measure, or both. If either the association measure or the evaluation measure is not interpretable, the overall evaluation is not interpretable, meaning that one cannot transfer it into a number of errors per unit time or per distance.

4.4 Conclusion

Firstly, the chapter emphasizes the evaluation of perception data based on object classifications as TP, FN, and FP. It acknowledges that not all FN and FP instances result in accidents, particularly when the trajectories of the other traffic participants do not intersect with that of the ego vehicle. To address this, this chapter analyzes the relation between scenario-based testing and a statistical evaluation of the environment perception of automated vehicles.
Furthermore, an analysis of existing evaluation measures is conducted. These measures are categorized as either relative or absolute evaluation measures. Absolute association measures enable the calculation of an error rate, which can be compared against the human failure rate of e.g. $1.52 \times 10^{-9}\,\mathrm{km}^{-1}$ [5, 6]. In contrast, relative measures only facilitate comparisons between different object detection algorithms without defining a safety threshold. An absolute measure is, therefore, a requirement to define what sensor system is sufficient for automated driving.
Only recall and precision possess the properties of an absolute measure. Other evaluation measures average over different types of errors, making the interpretation of their results challenging and limiting them to relative comparisons between object detections.
Furthermore, the interpretability of their values relies heavily on the error definition introduced by the association measure used. An ill-defined association measure may yield a favorable average performance despite the possibility of poor performance in specific regions within the field of view. This can be especially problematic for nearby detections.

To summarize, the conclusion encompasses the safety considerations of perception errors that do not impact the vehicle's driving behavior, an inventory of existing evaluation measures classified into absolute and relative association measures, and an analysis of the combination of evaluation measures and association measures for the performance evaluation of environment perception.

5 Assessing the perception reliabilities without reference truth

An approach to learning perception sensor reliabilities by exploiting the sensor redundancy in order to avoid the approval trap is proposed by [16]. According to [16], one can avoid the requirement for a reference truth by utilizing multiple redundant sensors. A summary of the model from [16] can be found in section 2.5.2. [16] did not deploy their model on real-world data. This work tests the model on real-world datasets with object data using two different association measures.
The model from [16] requires a binary representation of the surrounding: either an object is present or no object is present. Object data, however, is not binary. A subsequent question throughout the thesis project was, thus, how to extend the model in a way that also incorporates other parameters.
The chapter deals with:

- An application of the model from [16] in combination with the grid-based association from section 3.2 to estimate the sensor reliabilities without a reference truth.
- An application of the model from [16] in combination with the trajectory-clustering-based association from section 3.3 on a different dataset with more correlated data.
- An introduction of a model that also operates on non-binary data to also incorporate the confidence values that are provided by object detection algorithms in the evaluation and estimation of sensor reliabilities.

5.1 Learning the sensor reliabilities using the grid-based association

This section provides a first approach to learning the sensor reliabilities without a reference truth from object-data-based on the model which is summarized in section 2.5.2. The model was applied to a real-world dataset recorded for automated driving using object data. In order to build upon the required binary format of an object either being present or not, the grid-based approach from section 3.2 is applied to the dataset. The validation procedure, the results, and a discussion are summarized in the following sections. The work from this section is taken from our publication [70].

5.1.1 Validation

This section is taken from our publication [70] (©SAE International).

Using the reference truth, an estimate for the sensor reliabilities consisting of the POD_m and the PFA_m for each sensor m is derived, as well as the distribution of Y, which represents all sensors including their dependence as described in section 2.5.2. The conditional distributions of Y derived from the reference truth are denoted by $p_{ref}\left(y \mid \boldsymbol{\theta}, O=1\right)$ and $p_{ref}\left(y \mid \boldsymbol{\theta}, O=0\right)$ in the following. $p_{ref}\left(y \mid \boldsymbol{\theta}\right)$ describes the distribution of the 2^M sensor system outputs, also referred to as reference distribution, which is unconditional on the objects state. We approach the validation step-by-step as follows:

(I) We learn only the Dunnet-Sobel coefficients based on the reference truth data to assess whether the proposed dependence model can reflect the true dependence.

(II) We learn only the Dunnet-Sobel coefficients without reference truth data to assess whether the model stays persistent with the reference truth when the reference truth data is unknown. The Markov Chain Monte Carlo (MCMC) starting values are set to the values derived from (I).

(III) We learn all parameters of the dependent model without the reference truth data to assess whether the model finds the right values for the sensor reliabilities. The MCMC starting values are set to the values derived from the reference truth for p_{obj}, the POD_m values and the PFA_m values. The MCMC starting values for the Dunnet-Sobel coefficients are set to the values derived from (II).

(IV) We learn all parameters of the dependent model without the reference truth data. In comparison to (III) we set the MCMC starting values to the values obtained from the expectation maximization (EM) algorithm as explained in [16] to assess whether the sensor reliabilities can be learned when no prior knowledge is provided to the model.

Step (I) is included to verify that the model can in principle describe the data. The remaining steps are included to verify if and how well the proposed statistical models and MCMC algorithms are able to learn the sensor reliabilities without reference truth data. The NUTS algorithm is used to generate the Markov chains (MCs) [105, 106]. In every validation step 50 MCs with 500 tune steps and an additional 500 samples are generated each. A set of model parameters is obtained per chain by taking the mean value of the last 500 draws in analogy. The set of model parameters that yield the highest likelihood is used for the comparison with the reference values derived from the reference truth data.

5.1.2 Results

This section is taken from our publication [70] (©SAE International).

Model	MSE	Log-likelihood
Independent (O=1)	1.3588e-02	-2306634
Dependent (O=1)	3.5970e-04	-2232955
Independent (O=0)	1.2412e-05	-46653726
Dependent (O=0)	2.0699e-08	-45214280
Independent	1.4753e-05	-50028013
Dependent	3.1989e-08	-49542497

Table 5.1: Mean squared error and the logarithm of the likelihood for the PMFs from Figure 5.1 and Figure 5.2 (reprinted with permission from [70], ©SAE International).

Validation step (I)

Figure 5.1 (a) shows the PMF for the independent model p_{IM} and the dependent model p_{DM} in comparison to the reference PMF p_{ref} defined by the reference truth data for the case that an object is present $p\left(y \mid \theta, O=1\right)$ and Figure 5.1 (b) in case that no object is present $p\left(y \mid \theta, O=0\right)$. Figure 5.2 shows the resulting PMF $p\left(y \mid \theta\right)$ unconditional on the fact whether an object is present or not which is obtained by entering the conditional distributions from Figure 5.1 described by equations (2.5) and (2.6) of the independent model/equations (2.7) and (2.8) of the dependent model into equation (2.4). The three curves show the resulting PMF using the independent model, the dependent model and the output of the sensor system. The mean squared error of the PMF based on the independent model and the PMF based on the dependent model relative to the reference distributions conditional/unconditional on the presence/ absence of an object as well as the log-likelihood of the two models are listed in Table 5.1. The mean squared error (MSE) provides a measure for how well the resulting distributions of the independent model and the dependent model approximate the reference distributions based on the reference truth $p_{ref}\left(y \mid \theta, O=1\right)$, $p_{ref}\left(y \mid \theta, O=0\right)$ and $p_{ref}\left(y \mid \theta\right)$, respectively. As expected from Figure 5.1 and Figure 5.2, the MSE is much larger in the case of the independent model for the conditional PMFs as well as the unconditional PMF compared to the dependent model. Figure 5.1 (b) shows that in the case of the independent model, the fitted distribution is in agreement with the reference truth distribution only for some sensor system outputs. These sensor system outputs correspond to the most frequently recorded sensor system outputs when either no sensor recognized an object (index 1) or when only a single sensor recognized an object (indices 2, 3, 5, 9 and 17).

We also point out that the MSE for the dependent model distribution $p_{DM}\left(y \mid \theta_{DM}, o=1\right)$ conditional on an object being present is much larger than the MSE for the distribution conditional on no object being present. Figure 5.1 also shows that the distribution $p_{ref}\left(y \mid \theta, O=0\right)$ conditional on no object being present is better approximated by fitting the conditional distribution of the dependent model $p_{DM}\left(y \mid \theta_{DM}, O=0\right)$ than the distribution $p_{ref}\left(y \mid \theta, O=1\right)$ conditional on an object being present. This may be because the reference truth data contains fewer cases with an object being present compared to cases without an object being present in the region of interest of the rectangle with two-by-two meters squared. Therefore, the data without an object is better approximated by the distribution of the dependent model. Data balancing could reduce this effect in this validation

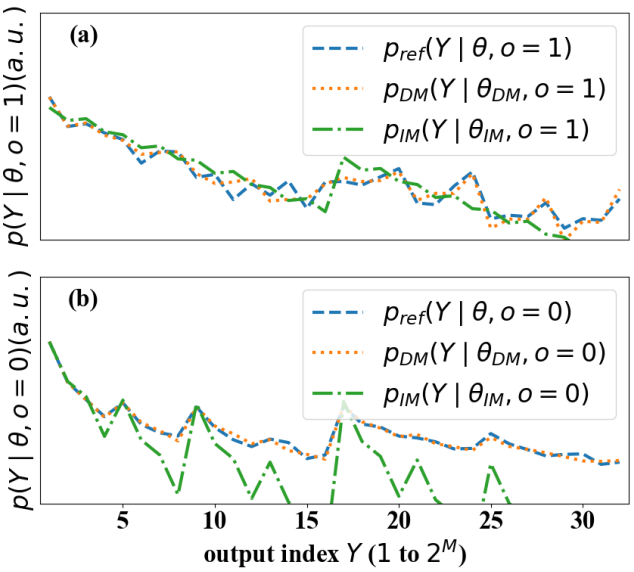


Figure 5.1: PMF conditional (a) on the case that an object is present $p(y \mid \boldsymbol{\theta}, O = 1)$ and (b) on the case that no object is present $p(y \mid \boldsymbol{\theta}, O = 0)$. The three PMFs correspond to the independent model, the dependent model and the reference distribution. The parameters of the models are learned from the data with reference truth. In case no object is present the independent model yields very small probabilities for lots of sensor system outputs. It becomes clear that these low values do not properly represent the reference distribution. Therefore, the distribution of the independent model is cut off so that the distribution of the dependent model and the reference distribution is better visible (reprinted with permission from [70], ©SAE International).

step. However, [16] proposes a model to learn the sensor reliabilities without a reference truth, as performed in validation step (IV). Data balancing is not possible without the reference truth; hence this is also not implemented here. In summary, the independent model's MSE is greater across all three distributions - the two distributions conditional on the fact that an object is present or not and the distribution unconditional on the objects state as seen in Table 5.1. The likelihood is always larger for the dependent model. This is explained by the fact that the independent model is equivalent to the dependent model while setting all the Dunnet-Sobel coefficients to zero. The sensors, however, are most likely statistically dependent and, therefore, the Dunnet-Sobel coefficient will not be equal to zero. Thus, by fitting the Dunnet-Sobel coefficients, the reference distributions can be better represented and the likelihood of the model is increased. It can be concluded that the independent model does not properly represent the reference distribution, even though the parameters are derived from the reference truth data. The independent model does not fit the reference distribution because it does not account for statistical dependence between the sensors. However, the sensors are most likely influenced by environment conditions such as weather and light conditions, which leads to a statistical dependence of the sensor outputs. Therefore, the results of the independent model are not further considered in subsequent validation steps while the dependent model is studied in further detail.

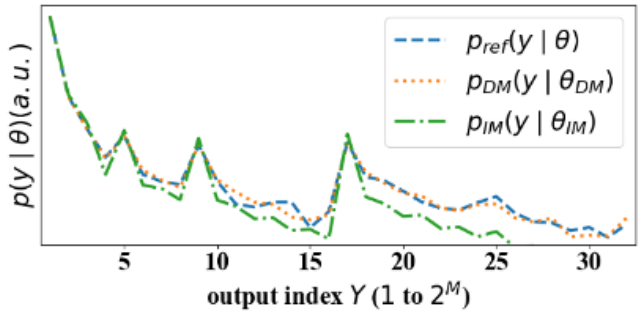


Figure 5.2: PMF $p(y \mid \theta)$ unconditional on the fact whether an object is present or not. The parameters correspond to the parameters of the PMF in Figure 5.1. The PMF is obtained using equation (2.4) (reprinted with permission from [70], ©SAE International).

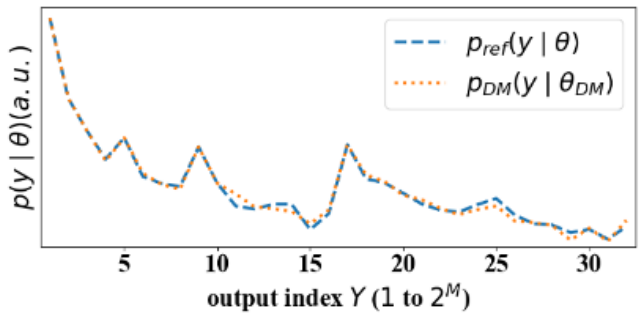


Figure 5.3: Estimated PMF $p(y \mid \theta)$. The PMF from the dependent model with fitted Dunnet-Sobel coefficients $\lambda_{POD,m}$ and $\lambda_{PFA,m}$ is compared to the reference PMF. The sensor reliabilities POD_m and PFA_m as well as the probability of an object being present are derived using the reference truth data (reprinted with permission from [70], ©SAE International).

Validation step (II)

Figure 5.3 shows the results when fitting the correlation parameters $\lambda_{POD,m}$ and $\lambda_{PFA,m}$ of the dependent model without reference truth data. The two curves correspond to the PMF $p_{ref}(y \mid \theta)$ of all observed sensor system outputs y from the sensor system and the PMF of the dependent model $p_{DM}(y \mid \theta_{DM})$. The curve of the dependent model approximates the PMF of the observed sensor system outputs closely.
Figure 5.4 (a) and (b) show the conditional probabilities $p(y \mid \theta, O = 1)$ and $p(y \mid \theta, O = 0)$, respectively, which are obtained by filling in the sensor reliabilities POD_m and PFA_m derived from the reference truth data and the fitted Dunnet-Sobel coefficients λ_{POD} and λ_{PFA} into equation (2.7) and (2.8), respectively. In comparison to the conditional probabilities from Figure 5.1 (a) and (b), where the $\lambda_{POD,m}$ and $\lambda_{PFA,m}$ were fitted based on the reference truth data, the curves deviate stronger from the reference distribution, especially, for the case when an object is present in Figure 5.4 (a) it is clearly visible.
The corresponding MSE and the log-likelihood are listed in Table 5.2. While the MSE values of the conditional PMFs are increased compared to the conditional PMFs from

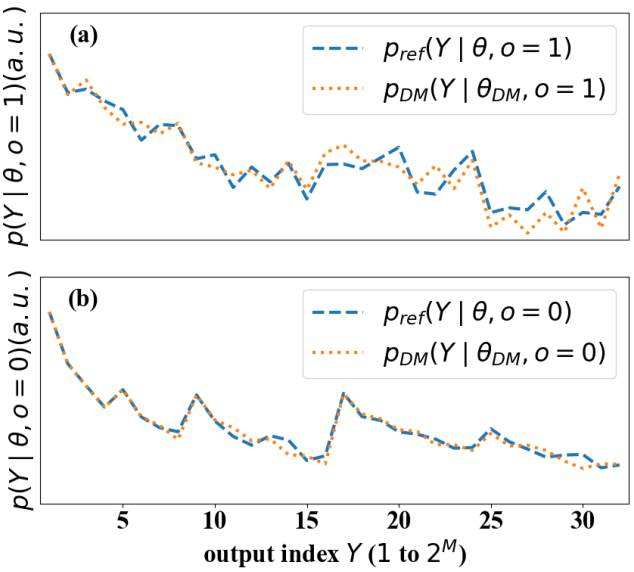


Figure 5.4: PMF conditional (a) on the case that an object is present $p\,(y \mid \boldsymbol{\theta}, O = 1)$ and (b) on the case that no object is present $p\,(y \mid \boldsymbol{\theta}, O = 0)$. The parameters are derived in analogy to the parameters of the PMF in Figure 5.3 (reprinted with permission from [70], ©SAE International).

Table 5.1, the MSE of the unconditional PMF $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM})$ is decreased. Moreover, the MSE of the unconditional distribution $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM})$ is smaller compared to the MSE of the PMF conditional on the fact that no object is present $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 0)$ in Table 5.2 while in Table 5.1 the MSE of the unconditional distribution $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM})$ is larger than the PMF conditional on the fact that no object is present $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 0)$. It can be concluded that in case all Dunnet-Sobel coefficients are fitted based on data without reference truth, the resulting unconditional distribution of the dependent model approximates the reference distribution better as seen in Figure 5.3 as the number of free parameters is increased to fit the distribution. A quantitative confirmation is given by the decrease of the MSE from Table 5.2 versus the MSE from Table 5.1 for the distribution $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM})$ unconditional on the state of the object. While the distribution unconditional on the state of the object is fitted better, however, the conditional distributions $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 0)$ and $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 1)$ are no longer that well described by the fitted parameters. The parameters do not represent the true correlation of the sensors perfectly. Thus, even though the superposition of the two conditional distributions leads to a better approximation of the distribution unconditional on the state of the object, one conditional distribution is overestimated while the other conditional distribution is underestimated for specific sensor system outputs.

Validation step (III)

Figure 5.5 shows the resulting distribution $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM})$ from the dependent model when fitting all parameters without reference truth data. Figure 5.6 shows the corresponding distributions $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 0)$ and $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 1)$ conditional on the fact

Model	MSE	Log-likelihood
Dependent (O=1)	2.6615e-03	-2251719
Dependent (O=0)	2.8772e-08	-45220179
Dependent	2.6371e-08	-49536819

Table 5.2: Mean squared error and the logarithm of the likelihood for the PMFs of Figure 5.3 and Figure 5.4 (reprinted with permission from [70], ©SAE International).

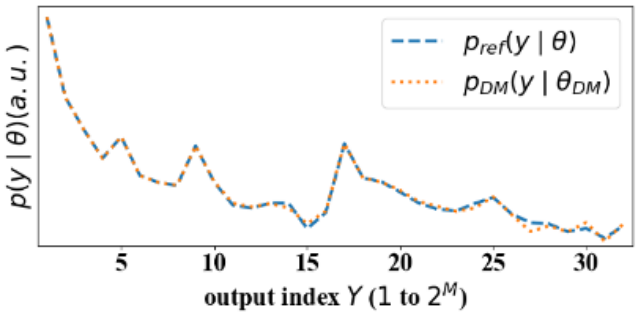


Figure 5.5: PMF $p\,(y \mid \boldsymbol{\theta})$ unconditional on the fact whether an object is present or not. The two curves represent the reference distribution and the resulting PMF $p\,(y \mid \boldsymbol{\theta})$ when fitting all parameters providing the values from validation step (II) as starting values to the MCMCs (reprinted with permission from [70], ©SAE International).

whether an object is present or not. It is visible that the fitted distributions do not represent the reference distribution perfectly. Table 5.3 shows the corresponding values for the MSE and the log-likelihood. While the MSE is decreased for the unconditional distribution $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM})$ by an order of magnitude compared to the values from validation step (II), the MSE for the distributions $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 0)$ and $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 1)$ conditional on the state of the object is much larger compared to the values in Table 5.2. It can be summarized that when fitting all parameters, the dependent model better fits the reference distribution. However, the conditional distributions from Figure 5.6 are fitted worse compared to the conditional distributions derived by using the reference truth data as seen in Figure 5.3.

Figure 5.7 shows the histogram for POD_m and PFA_m of the MC. The POD_m and PFA_m derived by using the reference truth data are shown by the vertical line. It can be seen that the POD_m and PFA_m values converge at values other than the ones derived by using the reference truth. However, the values are of the same order of magnitude. The values in Figure 5.7 are normalized with respect to the sensor reliabilities derived from the reference truth.

Model	MSE	Log-likelihood
Dependent (O=1)	2.2142e-02	-2307737
Dependent (O=0)	1.0177e-05	-45688336
Dependent	1.4296e-09	-49522532

Table 5.3: Mean squared error and the logarithm of the likelihood for the PMFs of Figure 5.5 and Figure 5.6 (reprinted with permission from [70], ©SAE International).

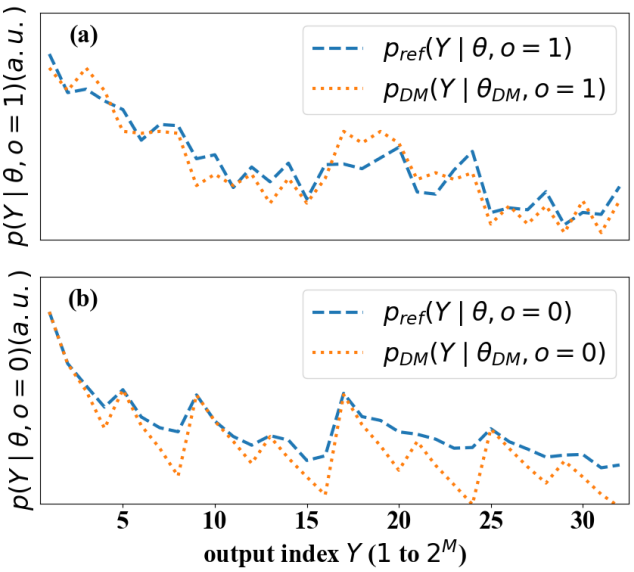


Figure 5.6: PMF conditional (a) on the case that an object is present $p\,(y \mid \boldsymbol{\theta}, O = 1)$ and (b) on the case that no object is present $p\,(y \mid \boldsymbol{\theta}, O = 0)$. All model parameters are fitted without providing the reference truth (reprinted with permission from [70], ©SAE International).

Validation step (IV)

Figure 5.8 shows the result when fitting the dependent model to the data without reference truth and neither deriving the starting values of the MCMCs from the reference truth data nor from previous validation steps. The distribution $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM})$ of the dependent model approximates the distribution of the measured sensor system outputs $p_{ref}\,(Y \mid \boldsymbol{\theta})$ rather well. Compared to Figure 5.5 there is hardly any difference visible. The conditional distributions $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 1)$ and $p_{DM}\,(y \mid \boldsymbol{\theta}_{DM}, O = 0)$ from Figure 5.9 appears to be identical to the conditional distributions from Figure 5.6.
Figure 5.10 shows a histogram of the learned sensor reliabilities when starting the MCMC with the values obtained by the EM algorithm with the independent model. The values are normalized by dividing through the value obtained with the reference truth data. The reference truth is, thus, represented by the vertical line at $(POD_m)/(POD_{(m,GT)}) = 1$ and $(PFA_m)/(PFA_{(m,GT)}) = 1$, respectively. The ranges of the x-axis and the width of the histogram bars are chosen to be the same as in Figure 5.7. It is visible that the learned values for the POD_m and PFA_m from Figure 5.10 converge to similar values as in Figure 5.7. The maximum deviation occurs in the case of the PFA_1 value of about 50 % relative to the value derived from the reference truth. Table 5.4 shows the resulting MSE and log-likelihood corresponding to the fitted values. The values for the MSE and the log-likelihood both for the conditional and the unconditional distributions confirm that the learned values are nearly the same as observed in validation step (III).

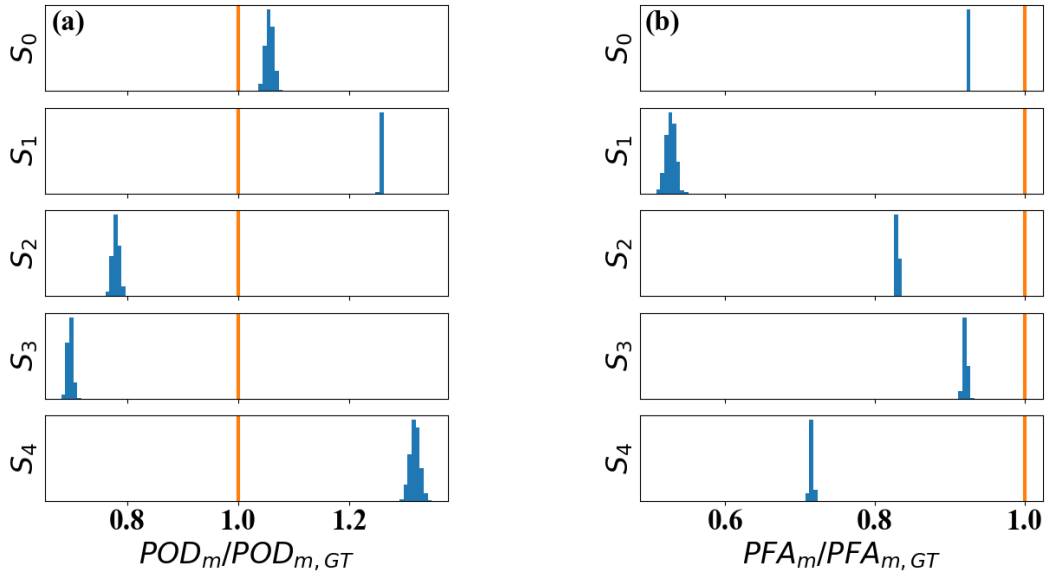


Figure 5.7: Histogram of the learned distribution for (a) the probabilities of detection POD_m and (b) the probability of false alarm PFA_m for the five sensors. Here, the starting values for the MCMCs are derived from the reference truth and the values from validation step (II). The values are normalized by dividing through the value obtained with the reference truth data (reprinted with permission from [70], ©SAE International).

Model	MSE	Log-likelihood
Dependent (O=1)	2.2148e-02	-2307774
Dependent (O=0)	1.0164e-05	-45687581
Dependent	1.4280e-09	-49522532

Table 5.4: Mean squared error and the logarithm of the likelihood for the PMFs of Figure 5.8 and Figure 5.9. The parameters of the distribution are learned starting from the result of the EM algorithm applied on the independent model (reprinted with permission from [70], ©SAE International).

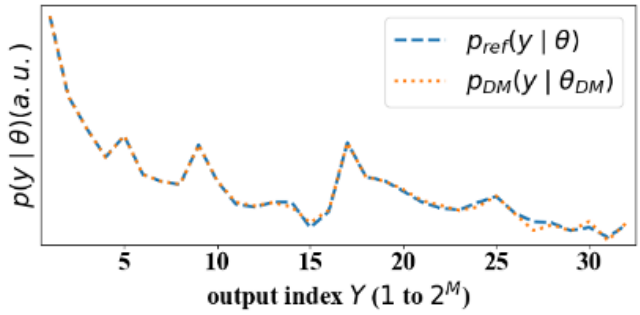


Figure 5.8: PMF $p(y \mid \boldsymbol{\theta})$ unconditional on the fact whether an object is present or not showing the resulting distribution when using the dependent model $p_{DM}(y \mid \boldsymbol{\theta}_{DM})$ relative to the reference distribution $p_{ref}(y \mid \boldsymbol{\theta})$ of the occurrences of the sensor system output with index $y = [1, \ldots, 2^M]$. The MCMC starting values for learning the parameters of the dependent model are obtained by applying the EM algorithm on the independent model (reprinted with permission from [70], ©SAE International).

5.1.3 Discussion

This section is taken from our publication [70] (©SAE International).

For validation, the statistical model from [16] to learn the sensor reliabilities without reference truth data was applied to a real-world data set. It is of interest, whether the model can properly represent real-world data. Another concern is whether the fitted parameters of the model are in agreement with parameters derived from reference truth data, which correspond to the sensor reliabilities. Plotting the distribution of the independent model as shown in Figure 5.1 and Figure 5.2 with the parameters derived from the reference truth data clearly shows that the assumption of statistical independence is not applicable for real-world data. The independent model may be useable to obtain a rough approximation for the order of magnitude of the sensor reliabilities. However, the independent model will not be a good model for a more precise analysis of the sensor reliabilities. In comparison to the independent model, the dependent model accounts for a correlation between the sensors with the integrated Dunnet-Sobel coefficients. With these additional free parameters in the dependent model, the distribution fits well with the conditional distributions which are based on the fact that an object is either present or not and which are shown in Figure 5.1. From the second validation step, however, we conclude that the Dunnet-Sobel coefficients may not represent the true correlation of the sensors perfectly when the reference truth is unknown. Therefore, when fitting all parameters of the dependent model, the probabilities of detection POD_m, the probabilities of false alarm PFA_m and the probability of an object being present p_{obj} cannot be learned exactly without reference truth.
In the work of section 5.1, we used a very basic association. This association method ignores that an object is recognized correctly when it was observed just at the border of a rectangle while the reference truth is just on the other side of the rectangle's border. This association is not suited for a practical evaluation of sensor reliabilities because it is very sensitive to small errors in position, decreasing the probability of detection POD_m and

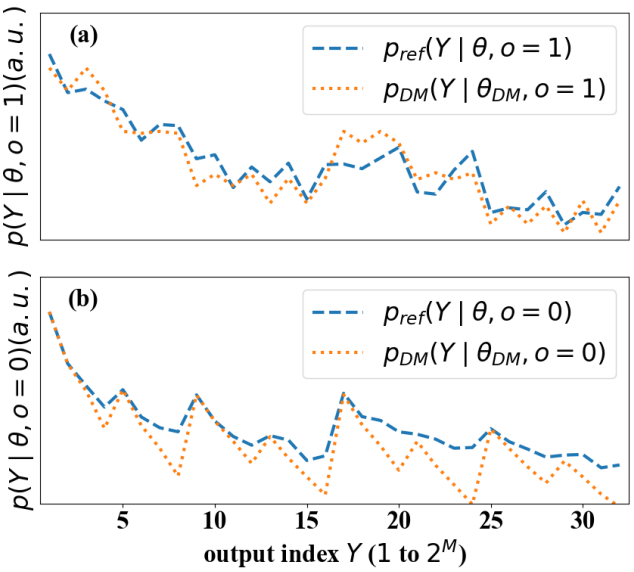


Figure 5.9: PMF conditional (a) on the case that an object is present $p\,(y \mid \boldsymbol{\theta}, O = 1)$ and (b) on the case that no object is present $p\,(y \mid \boldsymbol{\theta}, O = 0)$. The parameters are derived in analogy to the parameters of the PMF in Figure 5.8. The parameters are learned by setting the MCMC starting values to the result of the EM algorithm applied on the independent model as described in [16] (reprinted with permission from [70], ©SAE International).

increasing the probability of false alarm PFA_m of the sensors, even though in these cases the automated driving functionality would not have a problem to find a safe trajectory. However, our intention in this paper is to investigate how well the model fits real-world data, and not to make an absolute statement about sensor reliability. Other association methods will lead to an increase in the number of situations in which more than one sensor is recognizing an object, also leading to a higher correlation between the sensor data. As discussed in [16], an increase in sensor correlations can increase the credible interval of the learned parameters. To counteract the increased credible interval, more data might be necessary.

The behavior when fitting the model might be explained by the following two reasons: First of all, the available data was limited to about 1.5 million frames here. Even though the data was increased by aggregating the data from multiple locations to learn an average value of the parameters of the dependent model, subsequent time frames are statistically dependent reducing the effective sample size by about two orders of magnitude compared to the actual number of frames. Therefore, the model might show that it is more confident with the data that it found the right values than it is supposed to be. When having much more data, which is achievable with large fleets of vehicles, this effect may average out and, in case this is the reason for the deviation of the model from the conditional distributions derived with the reference truth, the model should finally learn the true values for the sensor reliabilities. The influence of the amount of data has a larger effect on the data when an object is present due to the fact that the number of frames with an object at a certain location, here specified by a two-by-two meter squared rectangle, is much smaller

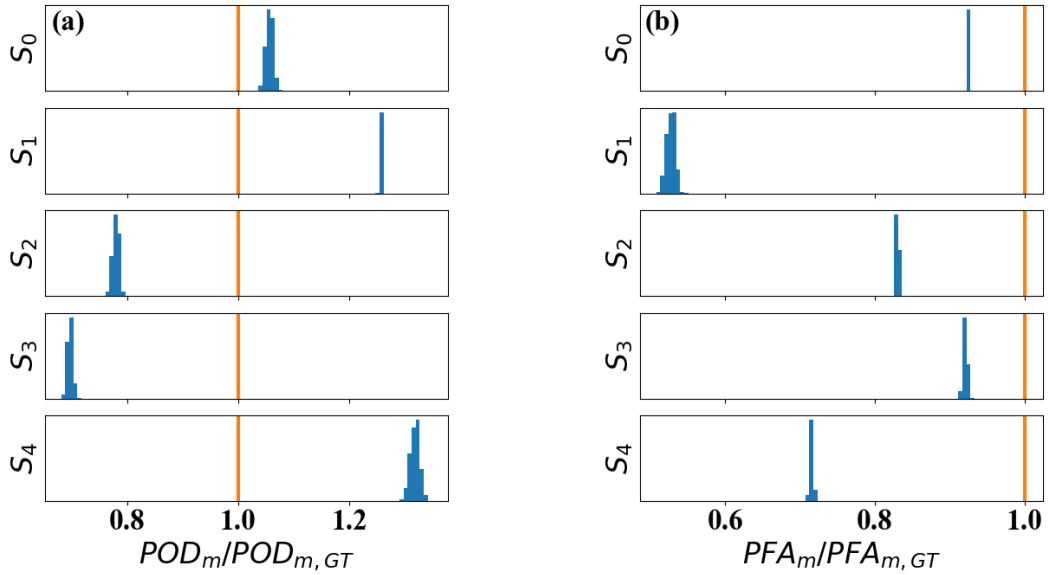


Figure 5.10: Histogram of the learned distribution for (a) the probabilities of detection POD_m and (b) the probability of false alarm PFA_m for the five sensors. Here, when deriving the starting values of the MCs by using the EM algorithm applied to the independent model. The values are normalized by dividing through the value obtained with the reference truth data (reprinted with permission from [70], ©SAE International).

than the number of frames without an object. Second, even though it is shown in this paper that the model can very well approximate the conditional distributions when just fitting the correlation parameters described by the Dunnet-Sobel coefficients with the reference truth data, when fitting all parameters, the fitted parameters do not correspond to the actual parameters derived from the reference truth data. The distributions for the sensor reliabilities derived from the MCs in validation step (III) and (IV) do not intersect with the sensor reliabilities derived from the reference truth.

When choosing the starting parameters of the MCMCs independent of the reference truth as performed in validation step (IV), the model converges to the same values fitted with providing starting values based derived from the reference truth data. Even though being different from the values for the sensor reliabilities derived from the reference truth data, these values are in the same order of magnitude with a maximum deviation of about 50 % compared to the values derived from the reference truth data. The resulting values may, therefore, be still usable as an approximation for the sensor reliabilities even if the values do not fit perfectly. However, the fitted parameters derived from the MCMCs do not always end up at the same values. Not all MCMCs of validation step (IV) end up at the values of validation step (III). Here, 50 MCs were generated to obtain this result. At least one MCMC of validation step (IV) ended with the same values as the best matching chain of validation step (III). However, it also happens that different values are found as it appears that the likelihood has many different maxima. As seen in validation step (IV) the maxima found by the MCMC may provide quite a decent estimate of the sensor reliabilities,

which yield sensor reliabilities is in the same order of magnitude as those derived from the reference truth data.

5.2 Learning the sensor reliabilities using the trajectory-clustering-based association

We repeat the evaluation from section 5.1 with the Waymo dataset [1]. The Waymo dataset does not provide data of five different sensors as required by the model in [16]. In order to obtain the data of five redundant sensors, we divided the LiDAR point cloud into equally sized subsets by using only every fifth horizontal line for one sensor. This results in strongly correlated object data sets which are obtained by applying the PointRCNN on each point cloud subset [13].
In order to associate the obtained object lists, this work utilizes the association measure introduced in 3.3. For comparison purposes with the reference truth, the reference truth object lists were included in the association.
The following sections summarize the validation procedure, the results and a discussion. The sections are taken from our contribution [57].

5.2.1 Validation

This section is taken from our publication [57] (©SAE International).

In this study, we first investigate if the dependent model can describe the binarized sensor data with five sensors derived from the Waymo dataset after the application of the object detection algorithm and the transformation into the binary format. We then evaluate if the correct values for the POD_m and the PFA_m can be learned without reference truth when utilizing the dependent model from equations (2.4), (2.7) and (2.8). We approach the validation of the model in two steps.
(I) we validate whether the sensor reliabilities derived from the reference truth can describe the measured distribution of the sensor system outputs. We derive the POD_m values, the PFA_m values and p_{obj} using the reference truth and enter these values into the model. We compare the obtained model distribution with the measured distribution of the $2^5 = 32$ possible sensor system outputs for $M = 5$ sensors. The equivalent correlation coefficients in the Gaussian space are obtained by the Nataf transformation [107]:

$$\rho_{S,ij} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(\frac{F^{-1}\left(\Phi\left(y_i\right)\right)-\mu_i}{\sigma_i}\right)\cdot\left(\frac{F^{-1}\left(\Phi\left(y_j\right)\right)-\mu_j}{\sigma_j}\right)\cdot\varphi\left(y_i,y_j,\rho_{ij}\right)\mathrm{d}y_i\mathrm{d}y_j \quad (5.1)$$

Here, $\rho_{S,ij}$ corresponds to the correlation coefficient in the binary space of the sensor detections; ρ_{ij} is the equivalent correlation coefficient in the Gaussian space; $\varphi\left(y_i,y_j,\rho_{ij}\right)$ is the bivariate normal distribution with correlation coefficient ρ_{ij}; $\Phi\left(y\right)$ is the standard normal CDF; $F^{-1}\left(u\right)$ is the inverse CDF of the Bernoulli distribution with the parameter

p being POD for the distribution conditional on the presence of an object and PFA for the distribution conditional on the fact that no object is present.

$$F^{-1}(u_m) = \begin{cases} 0, & \text{if } u_m \leq (1-p) \; with \; p = POD_m, PFA_m \\ 1, & \text{otherwise} \end{cases} \tag{5.2}$$

Equation (5.1) is solved with a root-finding algorithm and a numerical evaluation of the integral. The Dunnet-Sobel coefficients λ_m from equation (2.9) are approximated by the square root of the mean of all correlation parameters. We justify this approximation by the fact that all sensors are derived from the LiDAR-Data and, therefore, are expected to all have similar correlation coefficients.
In addition to the dependent model, we also utilize the independent model in validation step (I) to show the influence of lacking the dependence for the considered dataset.
In validation step (II) we validate the model's ability to estimate the sensor reliabilities without a reference truth. We provide random initial values to the optimization algorithm for finding the MLE. We compare the results with the reference from the previous validation step to see whether the resulting parameter estimates tend towards the values derived from the reference truth.
In comparison to [16], we used an optimizer to find the MLE instead of applying a Hamiltonian-based MCMC to find a distribution of the sensor reliabilities. The initially used Hamiltonian-based MCMC did not converge when using the Waymo dataset.
As optimizer for finding the MLE we use the SciPy [108] implementation of Powell's optimization [109]. The results are shown for grids with a chosen rectangle size of 0.5 m by 0.5 m and 2 m by 2 m.

5.2.2 Results

This section is taken from our publication [57] (©SAE International).

Figure 5.11 shows the reference distributions for the 32 possible sensor system outputs from the Waymo dataset for chosen grid cell sizes of 0.5 m x 0.5 m and 2 m x 2 m. The total number of present objects in the considered FOV was 222503 and 269924, respectively. Overall, the data of 158081 time-frames is used for the evaluation in this paper.
Table 5.5 summarizes the POD_m and the PFA_m derived from the reference truth. The probability of an object being present is $p_{obj} = 1.23 \times 10^{-3}$ when based on a grid with a cell size of 0.5 m x 0.5 m and $p_{obj} = 1.99 \times 10^{-2}$ when based on a grid with a cell size of 2 m x 2 m for the Waymo dataset when considering the FOV as described in section 3.3.

Validation step (I) - 0.5 m **x** 0.5 m

Figure 5.12 shows the resulting conditional and unconditional distributions where the parameters are derived from the reference truth and are inserted into the independent and the dependent model. The curve of the independent model is fully described by the coefficients of Table 5.5 together with the probability of an object being present p_{obj}. As

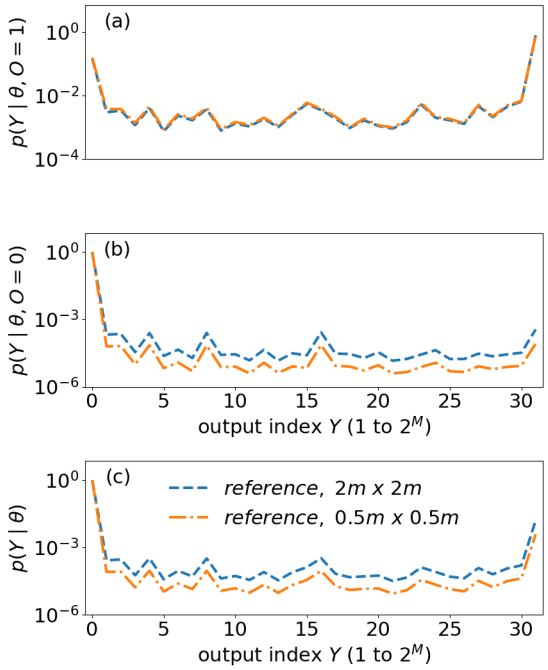


Figure 5.11: Reference distributions of the 32 possible sensor system outputs obtained by using the labeled reference data. Figure (a) shows the distribution $p\,(y \mid O = 1)$ conditional on the fact that an object is present. Figure (b) shows the distribution $p\,(y \mid O = 0)$ conditional on the fact that no object is present. Figure (c) shows the distribution $p\,(y)$ unconditional on the presence or absence of an object. The distributions are based on a discretization of the FOV with two different cell sizes: 0.5 m by 0.5 m and 2 m by 2 m (reprinted with permission from [57], ©SAE International).

Sensor	POD_m	PFA_m	Grid size
0	0.795	0.000064	
1	0.800	0.000066	
2	0.800	0.000068	0.5 m x 0.5 m
3	0.799	0.000067	
4	0.798	0.000067	
0	0.818	0.00090	
1	0.823	0.00096	
2	0.824	0.00099	2 m x 2 m
3	0.823	0.00099	
4	0.822	0.00099	

Table 5.5: The POD_m values and the PFA_m values derived from the reference truth (reprinted with permission from [57], ©SAE International).

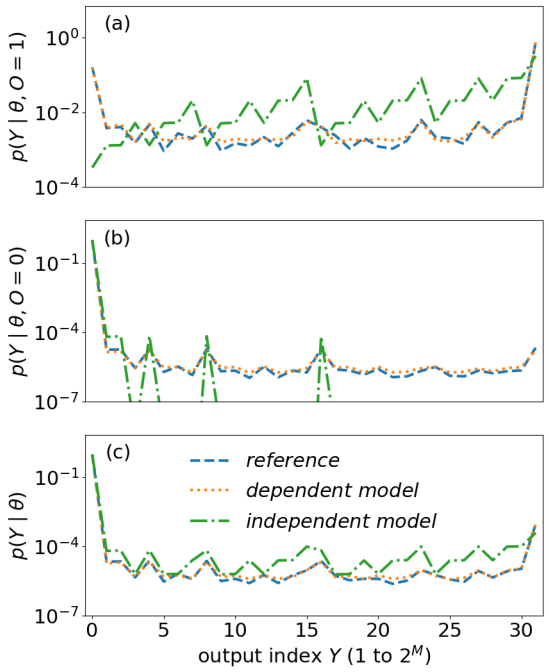


Figure 5.12: Validation step (I) - 0.5 m x 0.5 m: Resulting distributions of the system output when applying the reference values for the sensor reliabilities to the independent and the dependent model. Figure (a) shows the distribution $p\,(y \mid O = 1)$ conditional on the fact that an object is present. Figure (b) shows the distribution $p\,(y \mid O = 0)$ conditional on the fact that no object is present. Figure (c) shows the distribution $p\,(y)$ unconditional on the presence or absence of an object (reprinted with permission from [57], ©SAE International).

expected, the independent model cannot properly represent the reference distribution. In case an object is present, the independent model tends to underestimate the number of sensor system outputs where one or no sensor detects an object while overestimating the cases with more than one sensor detecting the object. In case no object is present, the opposite is observed.

In comparison to the independent model, the dependent model can approximate the distributions much better. The derived values for the Dunnet-Sobel coefficients are $\lambda_{o=1,m} = 0.99$ and $\lambda_{o=0,m} = 0.97$. The relative differences in the matrix elements of the low-rank Dunnet-Sobel class matrix defined by equation (2.9) compared to the full-rank correlation matrix obtained in equation (5.1) are smaller than 0.5 %.

The log-likelihood as well as the Kulback-Leibler-divergences derived from the models are listed in Table 5.6 for a quantitative comparison. The lower log-likelihood of the independent model compared to the dependent model is in agreement with the fit from Figure 5.12 and highlights that the dependent model is superior. The smaller KL-divergences of the dependent model also indicate a closer fit.

Distribution	Log-likelihood	KL-divergence
$p_{IM}\left(y_n \mid \boldsymbol{\theta}, O=1\right)$	-5.589e+05	1.517e+00
$p_{IM}\left(y_n \mid \boldsymbol{\theta}, O=0\right)$	-6.372e+05	1.483e-03
$p_{IM}\left(y_n \mid \boldsymbol{\theta}\right)$	-2.074e+06	8.664e-04
$p_{DM}\left(y_n \mid \boldsymbol{\theta}, O=1\right)$	-2.217e+05	2.024e-03
$p_{DM}\left(y_n \mid \boldsymbol{\theta}, O=0\right)$	-3.704e+05	4.493e-06
$p_{DM}\left(y_n \mid \boldsymbol{\theta}\right)$	-1.919e+06	4.211e-06

Table 5.6: Validation step (I) - 0.5 m x 0.5 m: Log-likelihood in case when an object is present, in case when no object is present and in any case independent whether an object is present or not using both, the independent and the dependent model. In addition, the table lists the Kulback-Leibler-divergence between the model distributions and the corresponding reference distributions (reprinted with permission from [57], ©SAE International).

Validation step (II) - 0.5 m **x** 0.5 m

Figure 5.13 shows the resulting distributions from validation step (II).
The distribution conditional on the absence of an object in Figure 5.13 (b) and the distribution unconditional on the presence or absence of an object in Figure 5.13 (c) are in agreement with the reference distribution.
Table 5.7 summarizes the fitted parameters $\boldsymbol{\theta}$ (except for p_{obj}) from validation step (II). The probability that an object is present was estimated by the dependent model to be $p_{obj} = 1.33 \times 10^{-3}$, larger than the reference value. All probabilities of missing an object $1 - POD_m$ and all probabilities of false PFA_m are in the same order of magnitude as the reference POD_m and the reference PFA_m from Table 5.5. The POD_m and most PFA_m are underestimated while the PFA_0 is overestimated. The Dunnet-Sobel coefficients deviate by up to 0.8 % in case an object is present and by up to 3.2 % in case no object is present.
The log-likelihood and the KL-divergence are listed in Table 5.8. In comparison to the previous validation step, the log-likelihood is smaller for the conditional distributions and the unconditional distribution. The opposite is observed for the KL-divergence. In the case of the unconditional distribution, the deviation in the log-likelihood is in the order of 1×10^2.

Sensor/Index	POD_m	PFA_m	$\lambda_{O=1,m}$	$\lambda_{O=0,m}$
0	0.723	0.000087	0.9991	0.9884
1	0.754	0.000053	0.9855	0.9924
2	0.756	0.000053	0.9877	0.9802
3	0.751	0.000053	0.9970	0.9420
4	0.746	0.000053	0.9961	0.9529

Table 5.7: Validation step (II) - 0.5 m x 0.5 m: fitted parameters $\boldsymbol{\theta}$: POD_m values, PFA_m values and the Dunnet-Sobel coefficients $\lambda_{O=1,m}$, $\lambda_{O=0,m}$ (reprinted with permission from [57], ©SAE International).

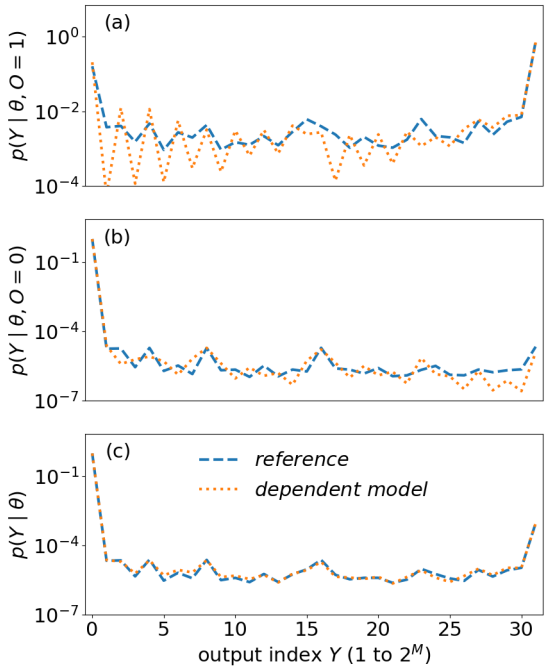


Figure 5.13: Validation step (II) - 0.5 m x 0.5 m: Resulting distributions of the system output when fitting the distributions without any prior knowledge about the sensor reliabilities. Figure (a) shows the distribution $p(y \mid O = 1)$ conditional on the fact that an object is present. Figure (b) shows the distribution $p(y \mid O = 0)$ conditional on the fact that no object is present. Figure (c) shows the distribution $p(y)$ unconditional on the presence or absence of an object (reprinted with permission from [57], ©SAE International).

Distribution	Log-likelihood	KL-divergence
$p_{DM}(y_n \mid \boldsymbol{\theta}, O = 1)$	-2.329e+05	5.144e-02
$p_{DM}(y_n \mid \boldsymbol{\theta}, O = 0)$	-3.782e+05	4.735e-05
$p_{DM}(y_n \mid \boldsymbol{\theta})$	-1.919e+06	4.695e-06

Table 5.8: Validation step (II) - 0.5 m x 0.5 m: Log-likelihood in case when an object is present, in case when no object is present and in any case independent whether an object is present or not using the dependent model. In addition, the table lists the Kulback-Leibler-divergence between the model distributions and the corresponding reference distributions (reprinted with permission from [57], ©SAE International).

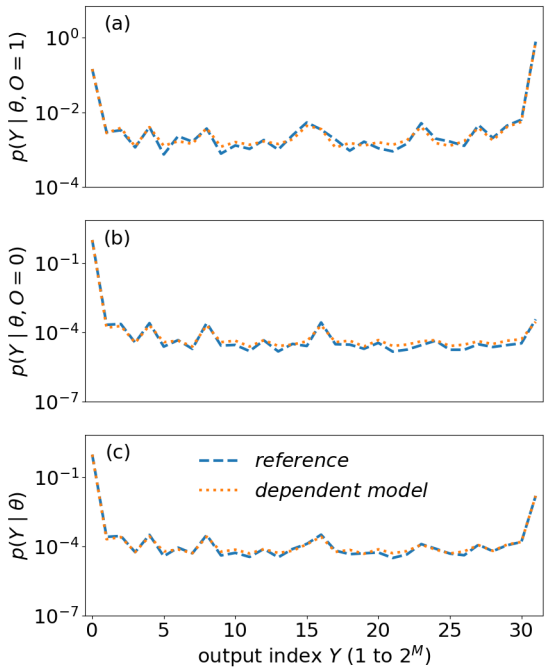


Figure 5.14: Validation step (I) - 2 m x 2 m: Resulting distributions of the system output when applying the reference values for the sensor reliabilities to the dependent model. Figure (a) shows the distribution $p(y \mid O = 1)$ conditional on the fact that an object is present. Figure (b) shows the distribution $p(y \mid O = 0)$ conditional on the fact that no object is present. Figure (c) shows the distribution $p(y)$ unconditional on the presence or absence of an object (reprinted with permission from [57], ©SAE International).

Validation step (I) - 2 m x 2 m

Figure 5.14 shows the resulting conditional and unconditional distributions for validation step (I) with a discretization in space of 2 m x 2 m.
The derived values for the Dunnet-Sobel coefficients are $\lambda_{o=1,m} = 0.99$ and $\lambda_{o=0,m} = 0.97$, the same as for the distributions based on the grid with cell size 0.5 m x 0.5 m. The relative differences in the matrix elements of the low-rank Dunnet-Sobel class matrix defined by equation (2.9) compared to the full-rank correlation matrix obtained in equation (5.1) are at most 1 %.
The log-likelihood, as well as the Kulback-Leibler-divergences, are listed in Table 5.9 for a quantitative comparison.

Validation step (II) - 2 m x 2 m

Figure 5.15 shows the resulting distributions from validation step (II) for the 2 m x 2 m cell size.
The distribution conditional on the absence of an object in Figure 5.15 (b) and the distribution unconditional on the presence or absence of an object in Figure 5.15 (c) are in agreement with the reference distribution. However, the distribution conditional on the presence of an object is underestimated by orders of magnitude for half of the sensor system outputs.

Distribution	Log-likelihood	KL-divergence
$p_{DM}\left(y_n \mid \boldsymbol{\theta}, O=1\right)$	-2.420e+05	1.72e-03
$p_{DM}\left(y_n \mid \boldsymbol{\theta}, O=0\right)$	-2.970e+05	8.64e-05
$p_{DM}\left(y_n \mid \boldsymbol{\theta}\right)$	-1.548e+06	6.40e-05

Table 5.9: Validation step (I) - 2 m x 2 m: Log-likelihood in the case when an object is present, in the case when no object is present and in any case independent whether an object is present or not using the dependent model. In addition, the table lists the Kulback-Leibler-divergence between the model distributions and the corresponding reference distributions (reprinted with permission from [57], ©SAE International).

Sensor/Index	POD_m	PFA_m	$\lambda_{O=1,m}$	$\lambda_{O=0,m}$
0	0.982	0.00110	0.902	0.953
1	0.983	0.00126	0.936	0.967
2	0.988	0.00121	0.842	0.954
3	0.996	0.00106	0.284	0.939
4	0.989	0.00119	0.822	0.945

Table 5.10: Validation step (II) - 2 m x 2 m: fitted parameters $\boldsymbol{\theta}$: POD_m values, PFA_m values and the Dunnet-Sobel coefficients $\lambda_{O=1,m}$, $\lambda_{O=0,m}$ (reprinted with permission from [57], ©SAE International).

Table 5.10 summarizes the fitted parameters $\boldsymbol{\theta}$ (except for p_{obj}) from validation step (II). The probability that an object is present was estimated by the dependent model to be $p_{obj} = 1.63 \times 10^{-2}$, smaller than the reference value. While the values for the probability of false alarm are in the same order of magnitude as the reference values, the probability of missing an object $1 - POD_m$ is underestimated by an order of magnitude in comparison to the reference values. The POD_m values are significantly overestimated in comparison to the values from Table 5.10, respectively. The Dunnet-Sobel coefficients deviate by up to 71.4 % in case an object is present and by up to 3.1 % in case no object is present.

5.2.3 Discussion

This section is taken from our publication [57] (©SAE International).

This work presents a pipeline consisting of an association approach and a statistical model

Distribution	Log-likelihood	KL-divergence
$p_{DM}\left(y_n \mid \boldsymbol{\theta}, O=1\right)$	-5.405e+05	1.11e+00
$p_{DM}\left(y_n \mid \boldsymbol{\theta}, O=0\right)$	-2.988e+05	2.24e-04
$p_{DM}\left(y_n \mid \boldsymbol{\theta}\right)$	-1.547e+06	2.31e-05

Table 5.11: Validation step (II) - 2 m x 2 m: Log-likelihood in case when an object is present, in the case when no object is present and in any case independent whether an object is present or not using both, the independent and the dependent model. In addition, the table lists the Kulback-Leibler-divergence between the model distributions and the corresponding reference distributions (reprinted with permission from [57], ©SAE International).

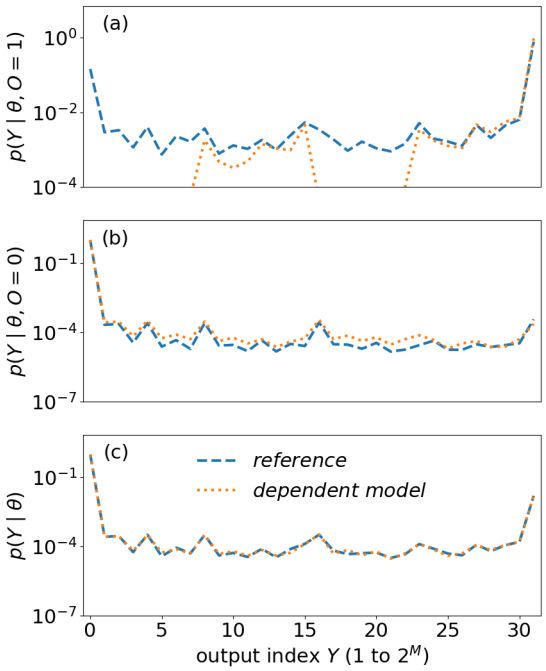


Figure 5.15: Validation step (II) - 2 m x 2 m: Resulting distributions of the system output when fitting the distributions without any prior knowledge about the sensor reliabilities. Figure (a) shows the distribution $p(y \mid O = 1)$ conditional on the fact that an object is present. Figure (b) shows the distribution $p(y \mid O = 0)$ conditional on the fact that no object is present. Figure (c) shows the distribution $p(y)$ unconditional on the presence or absence of an object (reprinted with permission from [57], ©SAE International).

to estimate the sensor reliabilities from real-world object data without the necessity of a reference truth. The pipeline includes a procedure to transform object lists into a binary format that can be used for the reliability analysis. The binary representation of the data is then used for the estimation of the sensor reliabilities. For the estimation of the sensor reliabilities without a reference truth, we incorporate the model from [16] in the pipeline. The proposed procedure to transform the object lists into a binary representation consists of a multi-object-tracking step, a clustering step and a binarization step. Multi-object-tracking is performed to avoid short-time errors for example when an object is missed one or two frames due to occlusion. The clustering is performed to relate the different objects obtained from different sensors. Finally, the binary data is obtained by subdividing the FOV into a grid. The cell size of the grid is a free parameter. There are no specific criteria for choosing the cell size except that it has to be small enough such that two neighboring objects cannot be within the same grid cell. For cars, which were the only objects that were investigated in this paper, we prescribed a maximum cell size of about 2 m x 2 m due to the fact that center points of neighboring cars hardly get closer than $2\sqrt{2}$ m ≈ 2.83 m. The difference in the number of objects is 0.5 % when using cell sizes of 0.5 m x 0.5 m and 2 m x 2 m and when considering the same FOV. This underlines the assumption that two neighboring cars get hardly closer than 2.83 m. The difference in the number of objects in the result section comes from the derivation of the FOV based on erosion as demonstrated in Figure 3.11 of section 3.3, which yields slightly different FOVs for different cell sizes. The FOV appeared to be smaller when using grid cell sizes of 0.5 m x 0.5 m rather than grid cell sizes of 2 m x 2 m. The main difference between distributions derived using different cell sizes, however, lies in the number of times when no object is present and no object is detected as can be seen in the reference distributions in Figure 5.11 (b). When choosing a cell size of 0.5 m x 0.5 m instead of 2 m x 2 m, the occurrence of no object being present and no object being detected increases by approximately a factor of 16. Thus, the chosen cell size has a great influence on the resulting distribution, which can influence the learning of the sensor reliabilities with the model from [16]. As a result, we investigate the performance of the model from [16] using two different cell sizes.

We then perform a validation of the resulting pipeline using the Waymo dataset and assess (a) whether the model is able to describe the real-world data and (b) whether the reliabilities of the sensors can be learned from the real-world data without the reference truth.

Addressing question (a): the dependent model approximates the conditional probability distributions well in validation step (I) given the parameters derived from the reference truth. This observation was achieved for any of the chosen cell sizes. The low-rank Dunnet-Sobel class matrix is sufficient for a good approximation of the distributions, which are based on the Waymo data. A full-rank correlation in the Gaussian copula is not required here. Validation step (I) also indicates that accounting for the dependence of the system is necessary for describing the real-world dataset. The independent model does describe the real-world data well. Validation step (I) proves that, independent of the chosen cell size, the dependent model is capable of describing the unconditional as well as the conditional

distributions. The different grid sizes yield different distributions of the object detections. As a result, we can assume that the model is capable of describing the distributions derived also from different association methods closely. This is a necessary condition for the model in order to fit the distribution properly and was, therefore, tested in the first step.
In order to address the question (b), we perform validation step (II). In the case of the distribution based on a cell size of 0.5 m x 0.5 m, the optimizer tends to the reference values. The fitted log-likelihood is close to the log-likelihood which is obtained when entering the reference values in the model. However, the optimization stops at values other than the reference while the log-likelihood is still smaller. Most likely, the optimizer gets stuck in a local minimum. The smaller log-likelihood of the fitted distribution indicates that the reference values provide an estimate that is closer to the global maximum of the likelihood function. Actually, a higher log-likelihood and a smaller KL-divergence is expected for the unconditional distribution in validation step (II) in comparison to validation step (I). The opposite should be the case for the conditional distribution. This is due to the fact that the optimizer approaches the unconditional distribution while the conditional distributions rely on the estimation of p_{obj}. Over- and underestimation in the PMF conditional on the presence of an object compensates with a slight under- and overestimation of the same sensor system outputs in the PMF conditional on the absence of an object, which can lead to a better approximation of the unconditional distribution.
The expected behavior in validation step (II) is observed for the data based on 2 m x 2 m. Validation step (II) with the data based on a cell size of 2 m x 2 m also demonstrates a probability of missing an object $1 - POD$ an order of magnitude lower than the reference corresponding to a much higher reliability in comparison to the reference. This shows that the pipeline does not always provide a reliable estimation of real-world sensor reliabilities. Validation step (II) was repeated multiple times with different starting values for the optimizer for finding the maximum likelihood. The estimated POD_m values within each fitting repetition turned out to be approximately the same for the different sensors $m = 0, \ldots, 4$. The same is observed for the PFA_m values. However, the values deviate between repetitions and often deviated from the reference truth. Apparently, the log-likelihood has multiple local maxima and the dependent model is not capable of representing the association between the sensor reliabilities and the observed distribution correctly. A tendency of the optimizer in validation step (II) towards the reference values with the distribution based on cell sizes of 0.5 m x 0.5 m might, therefore, be a coincidence.
Better estimations of the sensor reliabilities might be achievable in the case of less dependent sensors, as higher correlations have a negative effect while learning the sensor reliabilities [16]. Here, high correlations between the sensors were introduced due to the artificial setup that was achieved by subdividing the LiDAR data.
Current studies, which focus on sensor fusion on the raw data level, seem to outperform the fusion on the object list level, especially in the case of multiple LiDARs [110–112]. Moreover, Radar, LiDAR and camera yield complementary information. The complementary information is often used to extract additional features, e.g., the relative speed of the surrounding objects in the case of RADAR using the Doppler effect. Therefore, the sensors

may not be considered as being redundant. An application of the proposed pipeline will most likely not be possible in the future without the partial use of artificially generated sensors, as done in this paper. Hence, an advanced method should be able to derive the sensor reliabilities using less redundant sensors.
A possible future application of the proposed pipeline may be to use it for testing redundant object detection algorithms. Future vehicles may apply different object detection algorithms that rely on different sets of sensors (e.g., Radar and Camera or LiDAR and Camera), which could be considered as different sensors and which may show less statistical dependence than the pseudo-sensors used in this paper.
In addition, the definition of an error should be standardized and the distribution of the sensor system outputs should not depend on a free parameter like the cell size of the grid. A grid size of 2 m x 2 m was initially the best compromise such that the cell size gets closest to the size of the objects without more than two objects occupying the same grid cell. A future approach may consider the area of the individual objects in the evaluation. Other ways of defining a perception error for the reliability analysis of perception sensors, such that the resulting reliability is interpretable, remain a research topic.

5.3 Considering confidence values of object detection algorithms in the perception evaluation

The previous sections on estimating the sensor reliability without a reference truth utilize the model by Berk et al. [16]. This model is based on the assumption that an object is detected or no object is detected by the sensor, which is represented by a Bernoulli distribution for each individual sensor [16, 88]. The reference truth is represented likewise with a Boolean representation describing that either an object is present or no object is present. However, machine learning algorithms usually yield object data that does not conform to a binary representation. In order to obtain a binary representation of the data, an association measure is necessary in order to associate the detections with the reference objects as discussed in chapter 3. In combination with the object lists, the object detection algorithms usually provide a confidence value for the detected objects. In the previous two sections, a threshold was utilized to obtain binary values representing the presence or absence of an object. Thus, in order to obtain a binary representation, one has to introduce an additional threshold for the confidence value, which is applied after the association between the detections.
Some evaluation measures incorporate the confidence value rather than setting a certain confidence threshold and evaluating the algorithm for this specific confidence threshold. Examples of these measures are the AP and the area under the ROC curve from section 4.2. Incorporating the confidence value in the estimation of the sensor reliabilities makes the evaluation measure independent of the freely chosen confidence threshold. This circumvents the necessity of choosing a specific confidence threshold. As varying thresholds may yield advantageous outcomes for distinct tasks, not choosing a threshold introduces a degree of adaptability and flexibility for applications in a later process. E.g., in parking assistance

systems, a too low confidence threshold can lead to many FPs, which can lead to driver distraction, but a too high threshold can lead to too many FNs, which can cause the driver to hit an object when relying on the system. An appropriate confidence threshold can, thus, vary between tasks and is an additional research topic besides the task of object detection.
A model for estimating the sensor performance that does not rely on a fixed confidence threshold is more general. The following section introduces a model that extends the model by Berk et al. [16] to account for continuous confidence values.

5.3.1 Method

The model is based on a Gaussian copula in order to account for the dependence of the sensor outputs. In order to account for the continuous confidence values between 0 and 1, a Beta distribution is utilized.
The sensor system output with M sensors is represented by the vector $\vec{s} = (s_1, \ldots, s_M)$, which contains values between 0 and 1 for each individual sensor detection. The Gaussian copula density of the model for M sensors is given by c_R.

$$c_R(u_1, \ldots, u_M) = \frac{exp\left(-\frac{1}{2} \cdot \begin{pmatrix} \Phi^{-1}(u_1) \\ \vdots \\ \Phi^{-1}(u_M) \end{pmatrix}^T \left(\mathrm{R}^{-1} - \mathrm{I}\right) \begin{pmatrix} \Phi^{-1}(u_1) \\ \vdots \\ \Phi^{-1}(u_M) \end{pmatrix}\right)}{\sqrt{\det(R)}} \tag{5.3}$$

$R \in [-1, 1]^{M \times M}$ represents the correlation matrix. Equation (5.3) ignores the normalization constants as the constants have no influence on the MCMC sampling.
The uniformly distributed variables $u_1, \ldots, u_M$ are obtained by applying the cumulative distribution function of the beta distribution, which is the regularized incomplete beta function, on the sensor outputs $s_1, \ldots, s_M$ under the assumption that the sensor outputs $s_1, \ldots, s_M$ are beta distributed.

$$u_m = I_{s_m}(a_m, b_m) = \frac{B(s_m; a_m, b_m)}{B(a_m, b_m)} \tag{5.4}$$

Where $B(s_m; a_m, b_m) = \int_0^{s_m} x^{a_m - 1} (1 - x)^{b_m} \, \mathrm{d}x$ is the incomplete beta function and $B(a_m, b_m) = \int_0^1 x^{a_m - 1} (1 - x)^{b_m} \, \mathrm{d}x$ is the beta function.
With these equations, the conditional probabilities of an object being present can be reformulated in terms of the Gaussian copula density.

$$p\left(\vec{s} \mid \vec{a}_o, \vec{b}_o, \rho_o, o = 1\right) = c_{R_o}\left(I_{s_1}(a_{1,o}, b_{1,o}), \ldots, I_{s_M}(a_{M,o}, b_{M,o})\right) \tag{5.5}$$

$$p\left(\vec{s} \mid \vec{a}_{\bar{o}}, \vec{b}_{\bar{o}}, \rho_{\bar{o}}, o = 0\right) = c_{R_{\bar{o}}}\left(I_{s_1}(a_{1,\bar{o}}, b_{1,\bar{o}}), \ldots, I_{s_M}(a_{M,\bar{o}}, b_{M,\bar{o}})\right) \tag{5.6}$$

The probability for a specific sensor output $\vec{s}$, independent of the case of whether an object is present or not, is the superposition of the two conditional probabilities associated with the probability of the presence/absence of an object.

$$p\left(\vec{s} \mid \theta\right) = p\left(\vec{s} \mid \vec{a}_o, \vec{b}_o, \rho_o, o = 1\right) \cdot p_o + p\left(\vec{s} \mid \vec{a}_{\bar{o}}, \vec{b}_{\bar{o}}, \rho_{\bar{o}}, o = 0\right) \cdot (1 - p_o) \tag{5.7}$$

Here, $\theta = \{\vec{a}_o, \vec{b}_o, \rho_o, \vec{a}_{\bar{o}}, \vec{b}_{\bar{o}}, \rho_{\bar{o}}, p_o\}$ represents the model parameters, which consist of the sensor parameters, and the environmental parameter p_o, which describes the probability that an object is present. The probability that no object is present is $p_{\bar{o}} = (1 - p_o)$. However, as p_o and $p_{\bar{o}}$ are directly dependent on each other, we will always express the terms in dependence of the probability that an object is present p_o.
The likelihood $\mathcal{L}\left(\theta\right)$ corresponds to the product of the probabilities $p\left(\vec{s}_n \mid \theta\right)$ for every measurement of the $n = 1, \ldots, N$ time frames evaluated with the given parameter set θ, which contains the sensor parameters and the probability of an object being present. From the likelihood, the posterior distribution for the model parameters θ can be obtained using Bayes' theorem.

$$\begin{aligned} f\left(\theta \mid \{\vec{s}_1, \ldots, \vec{s}_N\}\right) &\propto f\left(\theta\right) \cdot \mathcal{L}\left(\theta\right) \\ \mathcal{L}\left(\theta\right) &\propto \prod_{n=1}^{N} p\left(\vec{s}_n \mid \theta\right) \end{aligned} \tag{5.8}$$

The posterior can be estimated using MCMC. The marginal distributions for the model parameters θ are then approximated by the histograms obtained from the MCMC.
The sensor reliabilities can be derived from the estimated model parameters θ by applying the threshold which is chosen to specify a detection or no detection. A visualization is shown in Figure 5.16. From the continuous distribution of the confidence values, the intention is to derive a statistical estimate for the number of cases where an object is present and where no object is present. The learned distributions are independent of the chosen threshold. The threshold can still be varied and the corresponding FP and FN rates can be derived individually after fitting the distributions. Figure 5.16 illustrates an example threshold. The threshold defines the FP rate and the FN rate.

5.3.2 Validation

We test the model in two subsequent steps. First, in order to check the model for convergence, the model was tested using simulated data. In a second step, we perform an analysis with real-world data.

Simulation

In order to validate the model before applying it to real-world data, data for the sensor system outputs $\vec{s}$ were randomly simulated with $N = 5 \times 10^4$ time frames and $M = 2$ sen-

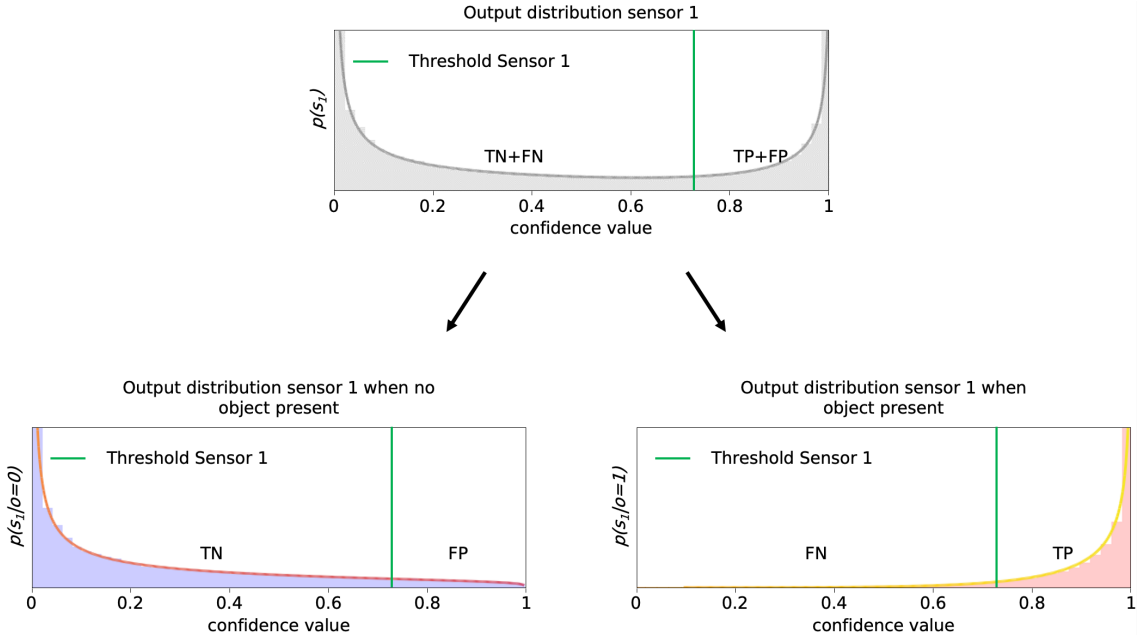


Figure 5.16: The measured distribution from one individual sensor may look like the grey distribution (simulated data here). By exploring the redundancies of two sensors, the distribution can be separated into cases where no object was present and cases where an object is present. By setting a threshold, here shown by the green lines, the number of TN, FP, FN and TP cases can be estimated from the learned distribution shown by the red and yellowish curve. Here, in addition to the curve the histograms derived by using the reference truth are shown in addition while the fitted curves in red and yellow approximate the histograms well.

sors. Random model parameters for the simulation were chosen, resulting in the following parameters: $\{\ a_{1,o}, a_{2,o},\ b_{1,o}, b_{2,o},\ \rho_o,\ a_{1,\bar{o}}, a_{2,\bar{o}},\ b_{1,\bar{o}}, b_{2,\bar{o}},\ \rho_{\bar{o}},\ p_o\ \} = \{\ 13.0, 16.0,\ 0.4, 0.7,\ 0.73, 0.4,\ 0.5, 47.0,\ 23.1, 0.64\ , 0.3\ \}$. The initial setup included the following conditions $a_{m,o} > 0$, $b_{m,o} < 0$ and $a_{m,\bar{o}} < 0$, $b_{m,\bar{o}} > 0$ so that the singularity of the functions are at 1 if an object is present and at 0 if no object is present as the sensors are assumed to are more often right than wrong.
Afterwards, the model fitted the simulated data. The prior distributions were chosen to be $U(1, 20)$ for the values of $\vec{a}_o$, $U(0.01, 1)$ for the values of $\vec{b}_o$, $U(-1, 1)$ for the value of ρ_o, $U(0.01, 1)$ for the values of $\vec{a}_{\bar{o}}$, $U(1, 50)$ for the values of $\vec{b}_{\bar{o}}$, $U(-1, 1)$ for the value of $\rho_{\bar{o}}$ and $U(0, 0.5)$ for the value of p_o.
The simulated data is fitted using MCMC. First, the reference values that are used to simulate the data are utilized as starting values of the MC. This allows one to validate that the model does not diverge. Second, random starting values are used in order to find out whether the model converges towards the reference values.

Application on real-world data

In the second step, this work analyzes the output of real-world data. The analysis is based on the Waymo dataset. In order to achieve multiple sensors from the LiDAR data, this study subdivides the LiDAR data into subsets and applied the PointRCNN algorithm on each subset of the LiDAR point cloud individually as in section 3.3. The performed analysis utilizes two sensors. The association between the sensor data is performed using the association measure from section 3.3. The data is normalized to the range $\left[1 \times 10^{-9}, 1 - 1 \times 10^{-9}\right]$ as the Beta distribution can have singularities at 0 and 1.

5.3.3 Results

Figures 5.17 and 5.19 show the simulated and the real-world, respectively. Figure 5.17 of the simulated data also contains the fitted distributions. The model does not represent the real-world data well and the fitting procedure does not converge. The following two subsections analyze the simulated and the real-world data in more detail.

Simulation

Figure 5.17 demonstrates the simulated data. The marginal distributions are shown along the axis of the plot. Red illustrates the cases where an object is present while blue illustrates the cases where no object is present. One expects more object detections with higher confidence when objects are present. More red data points are observed in the upper right corner. Fewer object detections, or object detections with less confidence, are expected when no object is present. A higher number of blue data points are shown in the bottom left corner, which corresponds to detections even though no object is present.
Figure 5.18 shows the resulting MCs for the 11 parameters $\boldsymbol{\theta}$ of the model when using two sensors. The MC demonstrates that the model can converge to the reference values when setting the starting values of the MC to a random set of values.

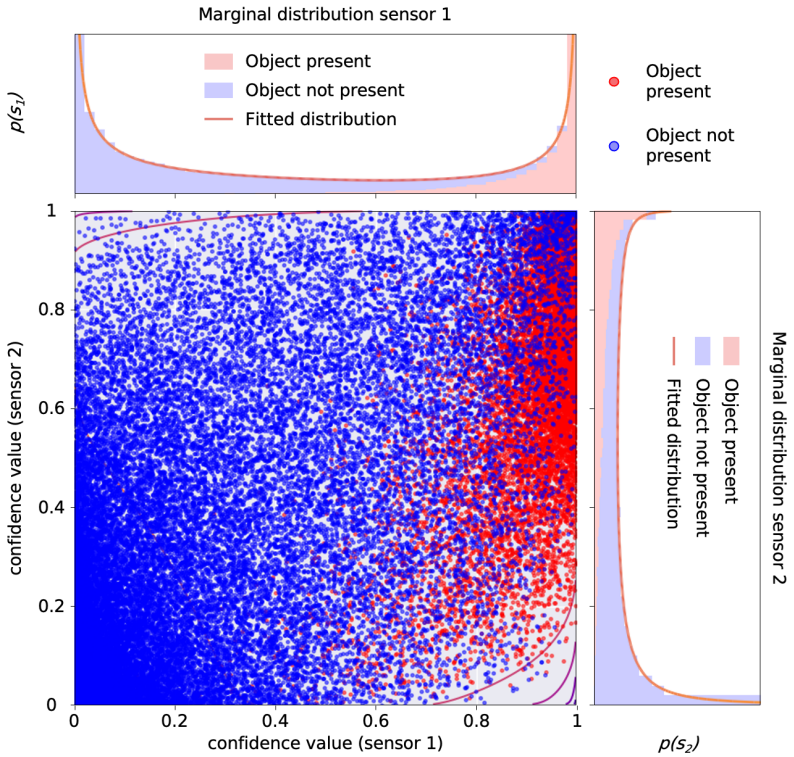


Figure 5.17: Visualization of the simulated data and the contour plot of the joint probability density function. The marginal distributions indicating the distributions of the individual (simulated) sensors are plotted along the axis of the plot. Red and blue data points differentiate between the presence and absence of an object.

For more sensors, the evaluation of the likelihood becomes computationally more demanding. A convergence with more than two sensors could not be observed when fitting all parameters simultaneously. One alternative procedure to fit all parameters is to fit the distribution with only a single sensor or with two sensors. The fitted parameters can then be utilized to subsequently find the correlation parameters by using the model for three parameters.

Application on real-world data

Figure 5.19 visualizes the data derived from the Waymo dataset after applying the PointRCNN to derive the object data and using the association measure from section 3.3 to associate the obtained object data. Figure 5.19 (a) shows the confidence values obtained from the PointRCNN which were transformed to the range $[0, 1]$. One can see in Figure 5.19 (a) that the algorithm assigns a confidence value of at least 0.2 to object detections and does not provide values in the entire range $[0, 1]$. The occurrence of no objects is obtained by introducing the grid after the association and accounting grid cells without any object as the occurrence of no object. The number of occurrences with no object and no detections are, therefore, dominant in the dataset and lead to a data imbalance between occurrences with and without objects.
Figure 5.19 (b) shows the data after removing all data points where both sensors did not detect an object. This reduces the observed data imbalance between cases with and without

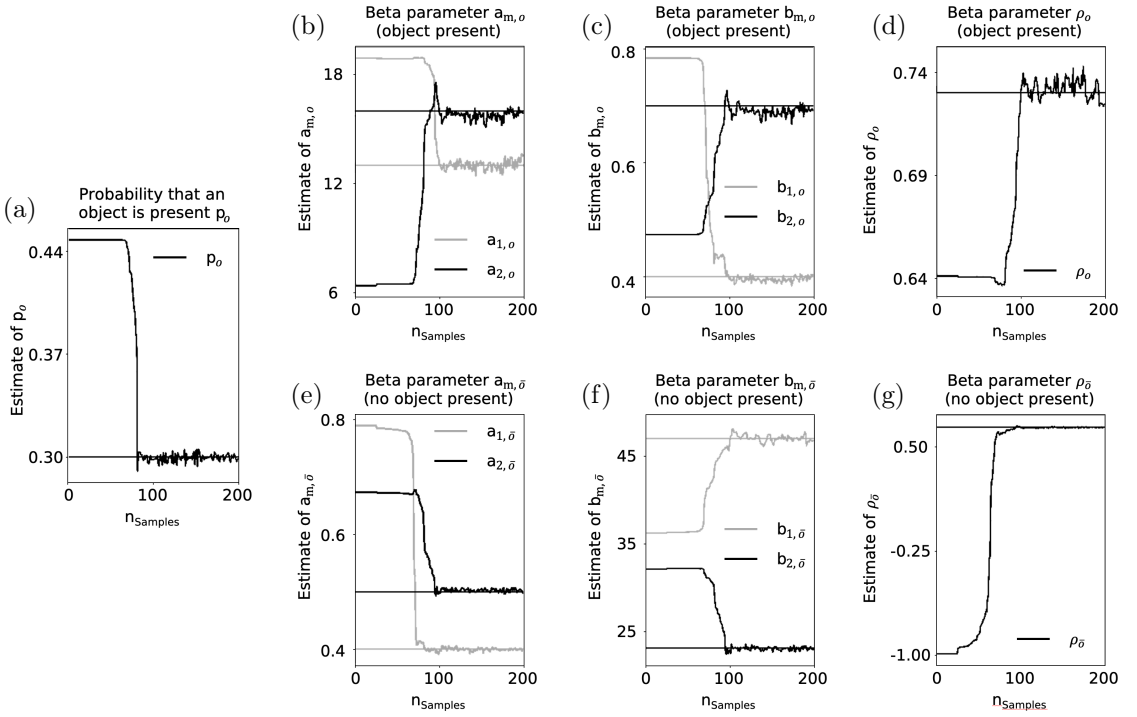


Figure 5.18: The resulting two MCs from fitting the simulated data for all 11 parameters in case of two considered sensors which are $\boldsymbol{\theta} = \{\ a_{1,o}, a_{2,o}, b_{1,o}, b_{2,o}, \rho_o, a_{1,\bar{o}}, a_{2,\bar{o}}, b_{1,\bar{o}}, b_{2,\bar{o}}, \rho_{\bar{o}}, p_o\ \}$. The fitting was once performed by starting with the reference values and once by starting with random values. In both cases the MCMC algorithm converges towards the reference values which are indicated in the plots for all parameters by horizontal lines.

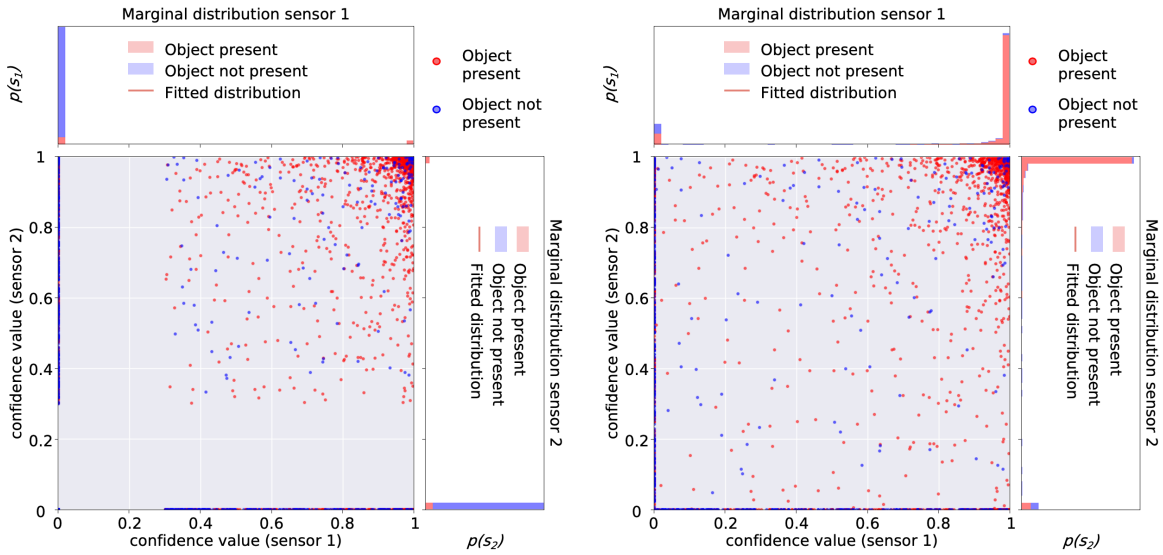


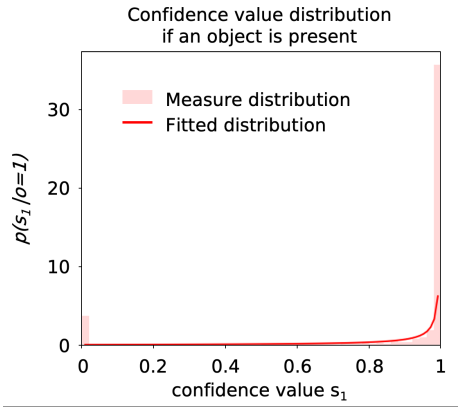
Figure 5.19: Visualization of data obtained from the Waymo dataset [1]. Object data is generated from two subsets of the LiDAR point cloud. The object data is associated using the procedure described in section 3.3. Figure (a) shows the unprocessed data. The utilized object detection algorithm assigned confidence values ≤ 0.2 to detections. As the Beta distribution covers all values between $[0, 1]$, the confidence values were transformed to the range $[0, 1]$. Moreover, due to the data imbalance between occurrences with and without an object, the occurrences without any detections were removed for the visualization in Figure (b).

present objects. However, it also excludes the FN cases in which both sensors have a FN at the same time. In addition, the values are transferred to the range $\left[1 \times 10^{-9}, 1 - 1 \times 10^{-9}\right]$ as, depending on the parametrization, the Beta distribution demonstrates singularities at 0 and 1.

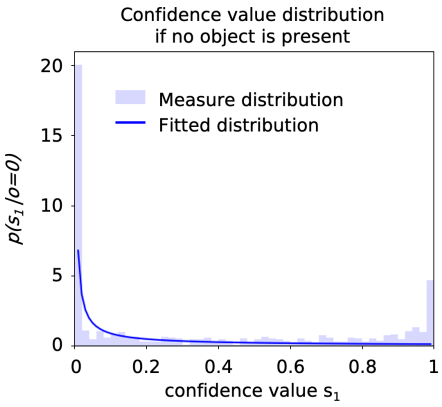
Figure 5.20 shows the distributions of the confidence values for sensor 0 conditional on the fact whether (a) an object is present in red or (b) no object is present in blue. The unconditional, marginal distribution of sensor 0 from Figure 5.19 is a superposition of these two conditional distributions. In addition to the distributions conditional on whether an object is present or not, Figures 5.19 (a) and (b) also show the fit of the Beta distribution for the conditional cases. Due to a higher number of observations at values close to 0 and values close to 1 both conditional distributions of the confidence values have a bathtub shape. The conditional distributions are not well approximated by the two Beta distributions.

Consequently, fitting the parameters of the beta functions turned out difficult. First, the confidence value distribution of a single sensor, sensor 0, was fitted. It turned out that fitting the unconditional distribution for sensor 0 did not result in the parameter values obtained by fitting the conditional distributions individually.

As the confidence values do not closely resemble a beta distribution, a model that takes the correlation as an additional parameter into account may perform better. However, fitting the joint distributions from Figure 5.19 (a) and (b) did not converge.



(a)



(b)

Figure 5.20: The fitted marginal distributions of Figure 5.19 (b). Figure (a) shows the distribution for the cases where an object is present while Figure (b) shows the distribution for the cases where no object is present. For the generation of the distributions, all occurrences where no object was present and where no object was detected by either of the sensors were excluded from the data.

5.3.4 Discussion

The presented model accounts for continuous confidence values of object data and, therefore, can be seen as an extension to the model described by Berk et al. [16].

One advantage of the idea behind the model is that instead of defining a threshold in advance, as necessary for the model from [16], no confidence threshold is required for this model. This allows for an analysis of a good choice for the confidence threshold even after determining the performance of the sensors with the Beta distribution. Moreover, instead of at least 5 different sensors as for the method in [16], in theory, only a single sensor is necessary for the application of the method in the case that the confidence values are Beta distributed.

However, the model based on continuous confidence values comes with numeric challenges in the evaluation of the likelihood.

The model by Berk et al. [16] requires a reduction of the confidence value provided by the object detection algorithm to a binary number saying either that the sensor recognized an object or it did not recognize an object. This is obtained by setting a threshold: objects with a confidence value exceeding the threshold are considered as detected objects while detected objects with a confidence value below the threshold are considered as no detected objects. In this binary representation, the amount of sensor system outputs is restricted to the number of 2^M while M is the number of redundant sensors. Thus, the number of evaluations of the probability $p(y \mid \boldsymbol{\theta})$ for a certain sensor system output y is limited to the 2^M for each iteration in the optimization or MCMC process and does not scale with the number of data points like in the presented model which becomes computationally demanding.

In comparison with the proposed model, the number of the possible sensor system outputs $\vec{s}$ is unlimited as every sensor output s_m, which corresponds to the confidence value provided

by the object detection algorithm, can be any value in the interval of 0 to 1. Therefore, the probability $p(\vec{s} \mid \theta)$ has to be evaluated for all N time frames in order to evaluate the likelihood. For a large amount N of recorded frames, which is necessary to also include driving scenes including relevant corner cases, it is, therefore, computationally very demanding to evaluate the likelihood. However, the many evaluations of the probability $p(\vec{s} \mid \theta)$ of the sensor system output $\vec{s}$ occurring are of the type single instruction multiple data (SIMD). The calculations can, therefore, be parallelized on graphical processing units (GPUs). In order to accelerate the fitting process we parallelized the evaluation of the probability $p(\vec{s} \mid \theta)$ on the GPU using the cupy package for numpy. However, despite the parallelization on the GPU, the fitting process was in the order of hours. For more than two sensors the fitting did not converge.

Furthermore, the chosen Beta distribution does not properly describe the distribution of the real-world data. The confidence values obtained from the PointRCNN applied on subsets of Waymo's LiDAR data are far from being Beta distributed as shown in Figure 5.20 [1]. Both conditional distributions of the data where either an object is present or not present are already bathtub shaped. Beta distributions with limited parameter intervals so that the beta distributions only demonstrate one pole at either 0 or 1 are, thus not suitable. Not limiting the parameter range, however, may not offer a robust solution. In this case, both conditional distributions can be bathtub shaped. A superposition of two bathtub-shaped functions, however, does not allow separation between cases when an object is present and when no object is present.

We want to point out that the shape of the distribution can vary for different object detection algorithms. Hence, multiple sensors with multiple object detection algorithms will result in multiple different distributions of confidence values. The Beta distribution is, thus, a strong assumption that may seldom be satisfied.

Including multiple sensors may be beneficial due to the additional fitting of the correlation parameter which is independent of the utilized marginal distributions. For the derived object data from two subsets of the Waymo LiDAR data, however, the model did not converge.

Summarizing, the model offers a mathematical description for Beta distributed confidence values from multiple sensors that are correlated. The model is computationally very demanding. The model converges in the case of Beta distributed values for two sensors. However, real-world confidence values are not expected to be exactly Beta distributed. Thus, there are probably only very limited applications for the utilization of the model.

5.4 Conclusion

The chapter focuses on the reliability analysis without a reference truth by exploiting sensor redundancies. The first two sections utilize the model from [16] using two different datasets. For the first dataset, the grid-based association is utilized. For the second dataset, the association based on trajectory clustering is used. The third section extends the model from [16] to account for continuous confidence values instead of binary outputs.

The model from [16] is capable of describing the data. However, when fitting the sensor reliabilities, the model can converge to values different from the reference.
The advantage of the model from section 5.3 is that in theory, a single sensor could be sufficient for deriving sensor reliabilities if the data adheres strictly to a Beta distribution. If the data does not perfectly conform to a Beta distribution, which is expected for real-world data, we anticipated the need for more than one sensor to also depend on the correlation parameter as an additional model parameter that is independent of the marginal distribution. Even with additional sensors, we assumed to use fewer sensors for the reliability estimation with the proposed model in comparison to the five redundant sensors required by the model in [16].
The model performs well in simulations with up to two sensors. In the case of more sensors, however, the fitting did not converge. Thus, for three sensors we first fitted the marginal distribution and utilized the derived parameter values of the marginal Beta distribution to obtain the correlation coefficients. Alternatively, one could do the same with the joint distribution of just two sensors and use the derived parameter values to determine the parameters of the third sensor. This could be scaled up to any number of sensors. However, as discussed in section 5.2, the use of many different sensors that provide redundant object data is not expected in future applications.
For the real-world data, we found out that it is not well described by the utilized Beta distribution of the model. A fit of the real-world data turned out to be not possible with the proposed setup.
In upcoming studies, one may consider employing different marginal distributions to better accommodate real-world data. In this case, since we selected subsets of LiDAR data to represent multiple sensors, the data distributions were approximately equal for all sensors. If different sensors were to be utilized, one could also tailor individual marginal distributions to align more closely with the respective sensor distribution as the correlation parameters are utilized in the copula which is independent of the marginal distributions.

6 Discussion and conclusion

A reliable environment perception is required for safe automated driving. The following sections discuss the contributions of this work to the assurance of a reliable environment perception and future research needs. The section starts with an overall discussion of the achieved results. The following section concludes the work and presents an outlook on possible future work.

6.1 Discussion

This work emphasizes the requirement of a reliable environment perception for safe automated driving. The evaluation of the environment and the assurance of its reliability is far from trivial due to a lacking interpretation of most methods utilized for environment perception. Methods for perceiving the environment commonly focus only on specific functions like object detection, traffic sign and traffic light detection, or lane and street detection [86]. Moreover, the functions often rely on neural networks which are difficult to interpret.

All functions are necessary for the safe maneuvering of the vehicle on public roads, although the focus of the environment perception evaluation often lies in the evaluation of object detection. One advantage of object detection lies in the interpretability of the resulting objects. The safety issue caused by an object that is not detected by the perception of an automated vehicle is easy to understand. One can imagine that not seeing the vehicle right in front of the ego vehicle can immediately result in an accident. Not detecting a traffic sign or the lanes may not cause an accident given no other traffic participants are present. And, in case other traffic participants are present, one could reduce the environment perception to object perception again.

Beyond common examples of detection failures like not detecting an object at all, an interpretable definition of an error of object detection algorithms is not trivial as shown and discussed in chapter 3. All introduced approaches rely on thresholds that define when a detection and a reference object are no longer the same. Defining crisp thresholds is non-intuitive as no crisp threshold in human perception and judgment exists. Whether a detection is shifted by 0.5 m or by 0.55 m from the reference object may not be relevant while a shift of 2 m might make a crucial difference. The question arises where to set the crisp threshold between 0.5 m and 2 m and why not move it some centimeters in any direction? The question arises for all parameters that are considered in the evaluation. Parameters that might be considered in the evaluation might be the position defined by

x-, y-, z-coordinates, size defined by height, width, length, and rotation around an axis perpendicular to the driving plane relative to the ego vehicle. Area and volume-based measures like the IoU, also known as the Jaccard index, implicitly incorporate these parameters, however, the question of the interpretability of the chosen threshold remains. As an example, KITTI utilizes a threshold of 0.7 for cars [49]. However, there is no clear reason why one should not use instead a threshold of, e.g., 0.8 or 0.75.

A defined perception error and the corresponding association measure allows to evaluate the object-based environment perception and estimate the reliability of the perception system. Chapter 4 provides a comparison of approaches to evaluate object detection algorithms once detections and reference objects are associated. And chapter 5 utilizes the model from [16] to learn the sensor reliabilities that are described by the probability of detection, also known as recall, and the probability of false alarm, also known as false positive rate.

The interpretation of the resulting perception performance values depends on the utilized association measure. No association measure from the comparison in section 3.1 is superior as all measures focus on different parameters that all have a legitimate influence on the perception performance. All of these association measures are parameter specific and might be sufficient for individual driver assistance systems, however, for highly automated driving these measures might be insufficient.

An accepted perception error and a corresponding association measure is essential in order to determine the perception reliability. The definition of a superior perception error might have to take additional components of the environment perception into account besides object detection.

Without a more general error definition, ever new scenarios can be generated that provide examples that violate current error definitions. Studies on automated driving still alternate between scenario-based testing and a stochastic evaluation. Scenario-based testing is very limited as the vehicle is not exposed to an infinite number of situations on public roads as the number of defined scenarios will always stay limited. In addition, as described in section 4.1, scenario-based testing interchanges the consequences, also described as criticality, of a malfunctioning perception with the reliability of the perception.

Section 3.5 defines an error based on the drivable area. The measure only focuses on the nearest obstacle that cannot be passed. It does not incorporate second row objects, so objects that are occluded by other objects in the 2D bird-eye perspective. An explicit evaluation of RADAR perception which may provide detections of occluded objects due to the nature of RADAR is, thus, not possible. Moreover, the measure does account for lane markings as they are drivable. However, the introduced measure is not limited to a finite set of object classes and, therefore, includes all obstacles that are not drivable by the vehicle. In addition, implicit perception evaluations even of second row objects can be obtained when predicting the drivable area in the future, as the trajectories of second row objects also have an influence on the first-row objects. In addition, the measure allows accounting for the distance-dependent performance of sensors and accounts for the fact that a perfect detection may not be required, especially for faraway objects.

The idea behind the distance-dependent error is the fact that humans, despite being able

to navigate safely on public roads, are not good at estimating distances precisely, especially for objects at far distances. Thus, for automated driving, precise detection of the position of surrounding vehicles may also not be necessary in order to navigate safely. Association measures do not account for distance-dependent detection performance. As a result, the evaluation of the perception is difficult or impossible to interpret and yields bad results, which are not sufficient for automated driving. Furthermore, faraway objects should only have little influence on path planning and, thus, allowing a larger deviation in the detection of faraway objects may increase the interpretability of reliability estimations for the perception.

Without an interpretable definition of the perception error, the evaluation only allows relative comparisons between different object detection algorithms as described in section 3.1. For cases where a reference truth is present, the reference truth corresponds to an ideal sensor. Thus, any tendency towards the reference can be considered as improvement, independent of the types of errors that are decreased, such that they lead to a better score in the evaluation.

Based on this argumentation, the performed reliability analysis of the environment perception in chapter 5 may seem superfluous as it does not include the latest error definition which incorporates the distance dependence. However, considering that the initial idea of this work lies on the reliability estimation of automated vehicles where it is non-feasible to gain enough reference data, chapter 5 provides a basis for an initial setup for estimating the sensor reliability from object data without an existing reference truth. In section 5.2.2, the resulting sensor reliabilities described by the POD and the PFA, are far from values that are sufficient for automated driving. The testing of the method by [16] itself, however, should already be testable with sensor data that is not yet sufficient for automated driving. Moreover, any error definition can be incorporated as the model is based on binary sensor output saying either an object is present or not which can only be achieved by defining the respective error first. Hence, chapter 5 provides two case studies with different distributions from two sensor datasets. The study shows that the model can represent the distribution with the reference values for the POD and PFA values of every sensor, however, the model is not robust in finding the POD and PFA of every sensor without reference. As a result, the case studies demonstrate that the model is not robust in fitting the two distributions that are derived from real-world data. The same applies to the model that is based on the Beta distribution to account for continuous confidence values rather than the binary output of an object being present or not. In conclusion, the two models presented in this work do not provide an approach that can solve the approval trap.

In order to make utilize these models for an estimate of the sensor reliabilities it would be essential to explore potential avenues for improving the robustness. One plausible approach could be to consider alternative distributions, apart from the Beta distribution, which might better capture the complexities of the continuous confidence values. For the discrete model, one could reevaluate alternatives for the underlying Gaussian copula. By conducting a thorough analysis, it may be possible to identify more suitable distributions. While being very vague as the distributions strongly depend on the chosen set of sensors,

such an exploration could hold promise for advancing the models presented in this study.

6.2 Outlook

The study explores various approaches to estimate the reliability with and without reference truth. The performed reliability analysis is based on data obtained from object detection. In object detection errors are defined by the utilized association either with reference truth objects or, in case that no reference truth is utilized, with other sensor's detections. The study extends the error definition beyond the scope of object detection by introducing a comparison based on the drivable area.

Besides a reliability analysis using object data, one could include the drivable area-based error definition in the reliability analysis. With a high-resolution sensor, the environment can be precisely perceived, which may allow the generation of the drivable area without human labeling. For the calculation of the ego trajectory future predictions of surrounding vehicles are required. The reference for such future predictions corresponds to the same dataset only with a time shift that defines how far the model is supposed to predict into the future.

Corner cases could be caught if the detections, like objects or the free space, differ significantly between different sensors as defined by the error definition. However, this can only be a rough indicator, as different sensors have complementary properties and a detection by a camera can be different from the detection of a RADAR due to the different physical principles used for detections.

Above all is the requirement for a generalized error definition that does not only account for moving and isolated objects like cars but also accounts for stationary objects in the environment like trees. The drivable area allows for an extended error definition in comparison to the error definition based on objects, as it incorporates the street to a certain extent and is not limited to a predefined set of objects. It, therefore, expands the scope of the research beyond object detection, considering the surrounding environment and potential obstacles or hazards in the absence of detectable objects.

However, lane markings, traffic signs and traffic lights are not included in the drivable area. Moreover, in some directions, even if there is an object, it might be of little relevance. For example, a plastic bag flying over the street. Future error definition may, therefore, include a weighting for the borders of the drivable area saying in which direction it might be worse to hit the object in comparison to other directions. Thus, these error definitions should be further investigated and integrated into the analysis to provide a more comprehensive understanding of sensor reliability, especially in scenarios where object-based detection is not the primary focus.

Besides the investigation of association measures, further research is required to find a widely accepted evaluation measure that introduces a reference for a sufficient environment perception for automated vehicles. The research on reliability analysis of automated vehicle sensors in this work has made progress by investigating various evaluation measures for obtaining a reliability estimate, including precision, recall, and mean average precision. Evaluation measures such as precision and recall can be classified as absolute measures,

as they can be translated into error rates, while evaluation measures like the mAP are considered a relative evaluation measure, allowing for comparisons between object detection algorithms but not providing an error rate (errors per unit time). In future research, one should take into account that an evaluation measure that can classify a perception sensor system as sufficient for automated driving can only be based on an absolute evaluation measure.

Furthermore, integrating distance-dependent error definitions in the reliability evaluation and finding an appropriate evaluation measure, should also be pursued. The research has introduced distance-dependent error definitions that have not yet been incorporated into the reliability estimation. Expanding the investigation to the reliability of sensors detecting and analyzing the drivable area is crucial as it expands the scope of the research beyond object detection, considering the surrounding environment and potential obstacles or hazards in the absence of detectable objects.

The investigated approaches for estimating the reliability without the need for a reference truth do not provide satisfying results. In the case of the model from [16], the obtained reliability estimates that were different from the reference truth indicate the need for further investigation and improvement. One potential avenue for enhancing the robustness of the model is to explore alternatives for the underlying Gaussian copula. Other copulas may better capture the complexities inherent in the sensor data.

Moreover, it is worth considering the practicality of utilizing object data from five different sensors in the research. The focus of future research appears to be on low-level sensor fusion rather than fusion at the object level as fusion on the raw data level seems to outperform the fusion on object list level [110–112]. Thus, obtaining object data from at least five redundant sensors for the same FOV may be challenging in future applications. Especially, as the considered sensor types for automated driving are only threefold, namely, camera, RADAR and LiDAR.

This motivates the development of other approaches for accessing the sensor reliability like the presented continuous model from section 5.3. However, in the case of the continuous model that accounts for continuous confidence values, it is crucial to recognize that the initially deployed Beta distribution does not fully capture the characteristics of the real-world data with its high peaks at 0 and 1. By thoroughly analyzing the data and considering alternative distributions, researchers may uncover a more suitable distribution that effectively models the underlying patterns, thus improving the reliability estimation.

6.3 Conclusion

A reliable environment perception is necessary for safe automated driving. This work focuses on concepts for obtaining reliability estimates of the environment perception for automated vehicles.

The evaluation of the environment perception is often based on object detection. Not detecting surrounding objects can be a direct cause of an accident. The object detection can be divided into two parts. The first part is an evaluation of each individual frame where object detections of the vehicle's environment perception are associated with manually

labeled reference objects. The second part consists of the accumulation of the associated objects over the entire dataset.

Bounding boxes identified through object detection algorithms provide the benefit of their results - the object boxes - are intuitively understood at a quick look. That being said, the detailed interpretation faces constraints due to the need for clear-cut definitions of distance and size as real-world objects seldom correspond to a rectangular box. Moreover, this is in contrast with humans, who often make decisions based on flexible rules including the surrounding context. Furthermore, object detection does not represent the full environment due to finite sets of object classes. Hence, a different error definition based on the drivable area is introduced.

The resulting value from the second part of the evaluation strongly depends on the chosen association. Moreover, accumulating different types of errors can further reduce the validity of the obtained reliability. Different ways to obtain an evaluation in object detection used in studies for automated driving were investigated.

An additional challenge with these approaches is the missing reference truth for billions of kilometers that are required for statistical significance to ensure reliability. This work includes two case studies using a model that was developed to solve the problem with the missing reference by exploiting sensor redundancies instead of requiring a reference truth. Two case studies based on the model are performed in this work, with highly and weakly correlated sensor data. The datasets used for the case studies included a reference truth for the observed object data. In both cases, the approximated reliability values obtained by the model are hardly in agreement with the reference values. The same applies to an additional model introduced in this work that additionally accounts for the continuous confidence scores of object detection algorithms.

How to run validation procedures that do not rely on reference labeling for infeasible amounts of kilometers remains an open research topic. In case a human-labeled reference truth is not required for the validation, one can validate the environment perception by running the algorithms in the background of human-driven vehicles and deploying the validation procedure in large vehicle fleets in order to achieve a large number of kilometers.

References

[1] Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., J. Guo, Y. Z., Y. Chai, B. C., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhao, S., Cheng, S., Zhang, Y., Shlens, J., Chen, Z. & Anguelov, D. *Scalability in Perception for Autonomous Driving: Waymo Open Dataset* arXiv 1912.04838, `https://waymo.com/open/`, last checked 2020-12-16. 2020.

[2] Liu, P., Yang, R. & Xu, Z. How Safe Is Safe Enough for Self-Driving Vehicles? *Risk Analysis* **0** (2018).

[3] Morando, M. M., Tian, Q., Truong, L. T. & Vu, H. L. Studying teh Safety Impact of Autonomous Vehicles Using Simulation-Based Surrogate Safety Measures. *Hindawi Journal of Advanced Transportation* (2018).

[4] Kalra, N. & Paddock, S. M. Driving to Safety: How Many Miles of Driving Would It Take to Demonstrate Autonomous Vehicle Reliability? *RAND Corporation* (2016).

[5] Schöner, H.-P. *Challenges and Approaches for Testing of Highly Automated Vehicles* in *Energy Consumption and Autonomous Driving* (ed Langheim, J.) (Springer International Publishing, Cham, 2016), 101–109.

[6] Wachenfeld, W. & Winner, H. in *Autonomous Driving - Technical, Legal and Social Aspects* (eds Maurer, M., Gerdes, J., Lenz, B. & Winner, H.) 425–449 (Springer, Berlin, Heidelberg, 2016).

[7] Scholz, M. & Nestlinger, G. Fast generic sensor models for testinghighly automated vehicles in simulation. *Elektrotechnik & Informationstechnik* **135,** 365–369 (July 2018).

[8] Berk, M., Schubert, O., Kroll, H.-M., Buschardt, B. & Straub, D. Assessing the Safety of Environment Perception in Automated Driving V ehicles. *SAE Int. J. Trans. Safety* **8,** 49–74 (Apr. 2020).

[9] Koopman, P. & Wagner, M. in *WCX World Congress Experience* (SAE International, 2018).

[10] Mullins, G. E., Stankiewicz, P. G. & Gupta, S. K. *Automated generation of diverse and challenging scenarios for test and evaluation of autonomous vehicles* in *2017 IEEE International Conference on Robotics and Automation (ICRA)* (2017), 1443–1450.

[11] Corso, A. L. *Safety Validation of Black-Box Autonomous Systems* `http://ai.stanford.edu/blog/black-box-safety-validation/` (Oct. 19, 2020).

[12] Ding, M., Huo, Y., Yi, H., Wang, Z., Shi, J., Lu, Z. & Luo, P. Learning Depth-Guided Convolutions for Monocular 3D Object Detection. *arXiv* (Dec. 2019).

[13] Shi, S., Wang, X. & Li, H. *PointRCNN: 3D Object Proposal Generation and Detection from Point Cloud* arXiv 1812.04244. 2019.

[14] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C. & Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *arXiv* (Jan. 2015).

[15] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J. & Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision* **88,** 303–338 (June 2010).

[16] Berk, M., Schubert, O., Kroll, H.-M., Buschardt, B. & Straub, D. Exploiting Redundancy for Reliability Analysis of Sensor Perception in Automated Driving Vehicles. *IEEE Transactions on Intelligent Transportation Systems* **21,** 5073–5085 (2020).

[17] Koopman, P. & Wagner, M. Challenges in Autonomous Vehicle Testing and Validation. *SAE Int. J. Trans. Safety* **4,** 15–24 (Apr. 2016).

[18] Hoss, M., Scholtes, M. & Eckstein, L. A Review of Testing Object-Based Environment Perception for Safe Automated Driving. *arXiv:2102.08460v1* (Feb. 2021).

[19] Peng, L., Wang, H. & Li, J. Uncertainty Evaluation of Object Detection Algorithms for Autonomous Vehicles. *Automotive Innovation* **4,** 241–252 (Jan. 2021).

[20] Wachenfeld, W. & Winner, H. in *Automated Driving - Safer and More Efficient Future Driving* (eds Watzenig, D. & Horn, M.) 419–435 (Springer International Publishing AG Switzerland, Sept. 2016).

[21] Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J. & Beijbom, O. PointPillars: Fast Encoders for Object Detection from Point Clouds. *arXiv:1812.05784v2.* arXiv: `1812.05784 [cs.LG]` (2019).

[22] Philion, J., Kar, A. & Fidler, S. Learning to Evaluate Perception Models Using Planner-Centric Metrics. *arXiv:2004.08745v1* (Apr. 2020).

[23] Rosique, F., Navarro, P. J., Fernández, C. & Padilla, A. A Systematic Review of Perception System and Simulators for Autonomous Vehicles Research. *Sensors* **19.** `https://www.mdpi.com/1424-8220/19/3/648` (2019).

[24] Steyer, S., Tanzmeister, G. & Wollherr, D. Grid-Based Environment Estimation Using Evidential Mapping and Particle Tracking. *IEEE Transactions on Intelligent Vehicles* **3** (2018).

[25] Danescu, R., Oniga, F. & Nedevschi, S. Modeling and Tracking the Driving Environment With a Particle-Based Occupancy Grid. *IEEE Transactions on Intelligent Transportation Systems* **12,** 1331–3142 (2011).

[26] *Why the Best Way to Detect Objects is not to Detect Them* https://www.baselabs.de/sensor-fusion/why-the-best-way-to-detect-objects-is-not-to-detect-them/.

[27] Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G. & Beijbom, O. nuScenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027* (2019).

[28] https://www.youtube.com/watch?v=Ucp0TTmvqOE.

[29] Yang, N., von Stumberg, L., Wang, R. & Cremers, D. *D3VO: Deep Depth, Deep Pose and Deep Uncertainty for Monocular Visual Odometry* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2020).

[30] Rao, R. & Chen, J.-H. *Stereo and 3D Vision* https://courses.cs.washington.edu/courses/cse455/09wi/Lects/lect16.pdf.

[31] *Puck LITE* Velodyne (5521 Hellyer Avenue, San Jose, CA 95138 USA, 2019).

[32] Chipengo, U., Krenz, P. M. & Carpenter, S. From Antenna Design to High Fidelity, Full Physics Automotive Radar Sensor Corner Case Simulation. *Modelling and Simulation in Engineering* (2018).

[33] Zhang, X., Kwon, K., Henriksson, J., Luo, J. & Wu, M. C. A large-scale microelectromechanical-systems-based silicon photonics LiDAR. *Nature* **603,** 253–258 (Mar. 2022).

[34] Berk, M., Dura, M., Rivero, J. V., Schubert, O., Kroll, H.-M., Buschardt, B. & Straub, D. *A Stochastic Physical Simulation Framework to Quantify the Effect of Rainfall on Automotive Lidar* in *SAE International Journal of Advances and Current Practices in Mobility* **1** (SAE International, 2019), 531–538.

[35] Yang, S.-W. & Wang, C.-C. On Solving Mirror Reflection in LIDAR Sensing. *IEEE/ASME Transactions on Mechatronics* **16,** 255–265 (Apr. 2011).

[36] Mai, T. *What are passive and active sensors?* https://www.nasa.gov/directorates/heo/scan/communications/outreach/funfacts/txt_passive_active.html (Oct. 14, 2012).

[37] Gentile, R. *Algorithms to Antenna: Increasing Angular Resolution Using MIMO Radar* https://www.mwrf.com/technologies/systems/article/21849496/algorithms-to-antenna-increasing-angular-resolution-using-mimo-radar (Dec. 20, 2018).

[38] Giannini, V., Hegde, M. & Davis, C. *Digital Code Modulation MIMO Radar Improves Automotive Safety* https://www.microwavejournal.com/articles/print/32668-digital-code-modulation-mimo-radar-improves-automotive-safety (Oct. 8, 2019).

[39] Rao, S. *Application Report - MIMO Radar* tech. rep. SWRA554A (Texas Instruments, 2017).

[40] Geiger, A., Lenz, P., Stiller, C. & Urtasun, R. Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research (IJRR)* (2013).

[41] Palffy, A., Dong, J., Kooij, J. & Gavrila, D. CNN based Road User Detection using the 3D Radar Cube. **5,** 1263–1270 (Jan. 2020).

[42] Nuss, D., Reuter, S., Thom, M., Yuan, T., Krehl, G., Maile, M., Gern, A. & Dietmayer, K. A Random Finite Set Approach for Dynamic Occupancy Grid Maps with Real-Time Application. *arXiv:1605.02406v2* (Sept. 2016).

[43] Badino, H., Franke, U. & Mester, R. *Free Space Computation Using Stochastic Occupancy Grids and Dynamic Programming* in (2007).

[44] Hu, P., Huang, A., Dolan, J., Held, D. & Ramanan, D. Safe Local Motion Planning with Self-Supervised Freespace Forecasting. *CVPR 2021* (2021).

[45] Bhattacharya, S. & Raj, R. A. Performance evaluation of multi-sensor data fusion technique for test range application. *Sadhana* **29,** 237–247. `https://doi.org/10.1007/BF02703734` (2004).

[46] Gelder, E., Manders, J., Grappiolo, C., Paardekooper, J.-P., Camp, O. O. & Schutter, B. *Real-World Scenario Mining for the Assessment of Automated Vehicles* in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)* (2020), 1–8.

[47] Geiger, A., Lenz, P. & Urtasun, R. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. *CVPR 2012* (2012).

[48] Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G. & Beijbom, O. nuScenes: A multimodal dataset for autonomous driving. *arXiv:1903.11027v5* (May 2020).

[49] Geiger, A., Lenz, P., Stiller, C. & Urtasun, R. *3D Object Detection Evaluation 2017* `http://www.cvlibs.net/datasets/kitti/eval_object.php?obj_benchmark=3d`.

[50] Weng, X., Wang, J., Held, D. & Kitani, K. AB3DMOT: A Baseline for 3D Multi-Object Tracking and New Evaluation Metrics. *ECCVW* (2020).

[51] Padilla, R., Passos, W. L., Dias, T. L. B., Netto, S. L. & da Silva, E. A. B. A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit. *Electronics* **10.** `https://www.mdpi.com/2079-9292/10/3/279` (2021).

[52] *ImageNet Large Scale Visual Recognition Challenge 2016 (ILSVRC2016)* `https://www.image-net.org/challenges/LSVRC/2016/index.php#scene`.

[53] Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R. & Ren, D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. *arXiv* (Nov. 2019).

[54] Čech, E. & Katětov, M. eng. in. Chap. 2 (Academia, Publishing House of the Czechoslovak Academy of Sciences, Praha, 1969). `http://eudml.org/doc/276998`.

[55] Rezatofighi, H., Tsoi, N., Gwak, J. Y., Sadeghian, A., Reid, I. & Savarese, S. Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression. *CVPR 2019* (2019).

[56] Powers, D. Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. *Mech. Learn. Technol.* **2** (Jan. 2008).

[57] Kryda, M., Berk, M., Qiu, M., Buschardt, B. & Straub, D. Assessing Perception Sensor Reliability from Field Tests Without Reference Truth. *SAE Technical Paper 2023-01-5078.*

[58] Cramér, H. *Mathematical Methods of Statistics* 7th ed. (Princeton University Press, 1957).

[59] Corso, A. & Kochenderfer, M. J. Interpretable Safety Validation for Autonomous Vehicles. *arXiv:2004.06805v2.* eprint: `2004.06805v2` (2020).

[60] Davis, J. & Goadrich, M. The relationship between Precision-Recall and ROC curves. *Proceedings of the 23rd international conference on Machine learning* (June 2006).

[61] Bernardin, K. & Stiefelhagen, R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. *EURASIP Journal on Image and Video Processing* (2008).

[62] Taha, A. A. & Hanbury, A. Metrics for evaluating 3D medical imagesegmentation: analysis, selection, and tool. *BMC Medical Imaging* **15** (2015).

[63] *nuScenes Detection Task* `https://www.nuscenes.org/object-detection?externalData=all&mapData=all&modalities=Any`.

[64] Deng, B., Qi, C. R., Najibi, M., Funkhouser, T., Zhou, Y. & Anguelov, D. *Revisiting 3D Object Detection From an Egocentric Perspective* in *35th Conference on Neural Information Processing System* (Dec. 2021).

[65] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C. & Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *Int J Comput Vis* **115,** 211–252 (2015).

[66] Simonelli, A., Bulò, S. R., Porzi, L., López-Antequera, M. & Kontschieder, P. Disentangling Monocular 3D Object Detection. *CoRR* **abs/1905.12365.** `http://arxiv.org/abs/1905.12365` (2019).

[67] Weng, X., Wang, J., Held, D. & Kitani, K. 3D Multi-Object Tracking: A Baseline and New Evaluation Metrics. *IROS* (2020).

[68] Jadon, S. *A survey of loss functions for semantic segmentation* arXiv. Sept. 2020.

[69] Fernandez-Moral, E., Martins, R., Wolf, D. & Rives, P. *A new metric for evaluating semantic segmentation:leveraging global and contour accuracy* in *2018 IEEE Intelligent Vehicles Symposium (IV)* (2018), 1051–1056.

[70] Kryda, M., Berk, M., Buschardt, B. & Straub, D. *Validating an approach to assess sensor perception reliabilities without ground truth* in *SAE WCX World Congress Experience Digital Summit* (SAE International, Apr. 2021).

[71] Kryda, M., Qiu, M., Berk, M., Buschardt, B., Antesberger, T., German, R. & Straub, D. *Associating sensor data and reference truth labels: A step towards SOTIF validation of perception sensors* in *Sixth IEEE International Workshop on Automotive Reliability, Test and Safety (ARTS)* (Oct. 2021).

[72] Kuhn, H. W. The Hungarian method for the assignment problem. *Naval Research Logistics* **2** (Mar. 1955).

[73] Lee, J.-G., Han, J. & Whang, K.-Y. Trajectory Clustering: A Partition-and-Group Framework. *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data* **7,** 593–604 (2007).

[74] Ester, M., Kriegel, H.-P., Sander, J. & Xu, X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)* (1996).

[75] Koopman, P. & Osyk, B. Safety Argument Considerations for Public Road Testing of Autonomous Vehicles. *SAE Int. J. Adv. & Curr. Prac. in Mobility* **1,** 512–523 (Apr. 2019).

[76] Wachenfeld, W. & Winner, H. Virtual Assessment of Automation in Field Operation - A New Runtime Validation Method. *10. Workshop Fahrerassistenzsysteme, Walting im Altmühltal* **10,** 161–170 (Sept. 2015).

[77] Oboril, F., Buerkle, C., Sussmann, A., Bitton, S. & Fabris, S. *MTBF Model for AVs - From Perception Errors to Vehicle-Level Failures* in (IEEE Intelligent Vehicle Symposium (IV), June 2022).

[78] Homm, F., Kaempchen, N., Ota, J. & Burschka, D. *Efficient Occupancy Grid Computation on the GPU with Lidar and Radar for Road Boundary Detection* in *IEEE Intelligent Vehicles Symposium* (June 2010).

[79] Wolff, J., Grundhoff, S., Bast, H., Gomoll, W., Oliveira, J., Sommer, M. & Kilimann, S. *Das Auto-Magazin im Internet* `https://www.alle-autos-in.de/porsche/porsche_911_carrera_s_ktcm6421.shtml`.

[80] Soille, P. *Morphological Image Analysis: Principles and Applications* (Springer, Berlin, Heidelberg, 1999).

[81] Gonzalez, R. C. & Woods, R. E. *Digital Image Processing* 4th ed. (Pearson, 2018).

[82] Bradski, G. The OpenCV Library. *Dr. Dobb's Journal of Software Tools* (2000).

[83] Romanova, M. A. & Mamchenko, M. V. *Method and Algorithm for Estimating the Maximum Total Error of an Automotive LiDAR* in *International Conference on Automatics and Energy (ICAE 2021)* (2021).

[84] *Depth perception* `https://en.wikipedia.org/wiki/Depth_perception`.

[85] Kirchheim, B. *Das menschliche Auge – wie wir Bilder sehen* `https://www.digitalkamera.de/Fototipp/Das_menschliche_Auge_wie_wir_Bilder_sehen/5619.aspx`.

[86] Li, L., Huang, W., Liu, Y., Zheng, N. & Wang, F. Intelligence Testing for Autonomous Vehicles: A New Approach. *IEEE Transactions on Intelligent Vehicles* **1,** 158–166 (2016).

[87] Armand, A., Ibanez-Guzman, J. & Zinoune, C. in *Automated Driving - Safer and More Efficient Future Driving* (eds Watzenig, D. & Horn, M.) 201–244 (Springer International Publishing AG Switzerland, Sept. 2016).

[88] Straub, D. *Lecture Notes in Engineering Risk Analysis* 2021.

[89] Henze, F., Faßbender, D. & Stiller, C. Identifying Admissible Uncertainty Bounds for the Input of Planning Algorithms. *IEEE Transactions on Intelligent Vehicles* (2021).

[90] Sackmann, M., Bey, H., Hofmann, U. & Thielecke, J. *Classification of Driver Intentions at Roundabouts* in *6th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS)* (2020), 301–311.

[91] Fagerlind, H., Heinig, I., Viström, M., Wisch, M., Sulzberger, L., McCarthy, M., Hulshof, W., Roynard, M. & Schaub, S. Analysis of accident data for test scenario definition in the ASSESS project. `https://bast.opus.hbz-nrw.de/opus45-bast/frontdoor/deliver/index/docId/559/file/Analysis_of_accident_data_for_test_scenario_definition_in_the_ASSESS_project.pdf`.

[92] Krajewski, R., Bock, J., Kloeker, L. & Eckstein, L. *The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems* in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)* (2018), 2118–2125.

[93] de Gelder, E., Op den Camp, O. & de Boer, N. Scenario Categories for the Assessment of Automated Vehicles. *Centre of Excellence for Testing and Research of Autonomous Vehicles, Nanyang Technological University, Singapore and the Land Transport Authority of Singapore* (2020).

[94] *PEGASUS Symposium 2019* `https://www.pegasusprojekt.de/en/about-PEGASUS` (May 14, 2019).

[95] Winner, H., Lemmer, K., Form, T. & Mazzega, J. in (eds Meyer, G. & Beiker, S.) 185–195 (Springer Nature Switzerland AG, 2019).

[96] Pütz, A., Zlocki, A. & Eckstein, L. Absicherung hochautomatisierter Fahrfunktionen mithilfe einer Datenbank relevanter Szenarien. *11. Workshop Fahrerassistenzsysteme und automatisiertes Fahren* (Mar. 2017).

[97] Dave, A., Dollar, P., Ramanan, D., Kirillov, A. & Girshick, R. Evaluating Large-Vocabulary Object Detectors: The Devil is in the Details. *arXiv:2102.01066v1* (Feb. 2021).

[98] *COCO Evaluate* `https://cocodataset.org/#detection-eval`.

[99] Gupta, A., Dollár, P. & Girshick, R. LVIS: A Dataset for Large Vocabulary Instance Segmentation. *arXiv:1908.03195v2* (Sept. 2019).

[100] Luiten, J., Osĕp, A., Dendorfer, P., Torr, P., A. Geiger, Leal-Taixé, L. & Leibe, B. HOTA: A Higher Order Metric for Evaluating Multi-object Tracking. *International Journal of Computer Vision,* 548–578 (Oct. 2020).

[101] Hosang, J., Benenson, R., Dollár, P. & Schiele, B. What makes for effective detection proposals? *arXiv:1502.05082v3* (Aug. 2015).

[102] Stiefelhagen, R., Bernardin, K., Bowers, R., Rose, R. T., Michel, M. & Garofolo, J. The CLEAR 2007 Evaluation. **4122,** 1–44 (Apr. 2006).

[103] Wen, L., Du, D., Cai, Z., Lei, Z., Chang, M.-C., Qi, H., Lim, J., Yang, M.-H. & Lyu, S. UA-DETRAC: A New Benchmark and Protocol for Multi-Object Detection and Tracking. *arXiv:1511.04136v4* (Jan. 2020).

[104] Solera, F., Calderara, S. & Cucchiara, R. Towards the evaluation of reproducible robustness in tracking-by-detection. *IEEE Conference on Advanced Video and Signal-based Surveillance* (Aug. 2015).

[105] Salvatier, J., Wiecki, T. V. & Fonnesbeck, C. Probabilistic programming in Python using PyMC3. *PeerJ Computer Science* (2015).

[106] Hoffmann, M. D. & Gelman, A. The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research* **15,** 1593–1623 (2014).

[107] Li, H.-S. & Au, S.-K. *Implementing Nataf Transformation in A Spreadsheet Environment and Application in Reliability Analysis* in *International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (QR2MSE)* (2013).

[108] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., 5orov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P. & SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* **17,** 261–272 (2020).

[109] Powell, M. J. D. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal* **7,** 155–162. eprint: `https://academic.oup.com/comjnl/article-pdf/7/2/155/959784/070155.pdf`. `https://doi.org/10.1093/comjnl/7.2.155` (Jan. 1964).

[110] Li, P., Chen, X. & Shen, S. Stereo R-CNN based 3D Object Detection for Autonomous Driving. *arXiv* (Apr. 2019).

[111] Wen, L.-H. & Jo, K.-H. Fast and Accurate 3D Object Detection for Lidar-Camera-Based Autonomous Vehicles Using One Shared Voxel-Based Backbone. *IEEE Access* **9,** 22080–22089 (Jan. 2021).

[112] Arnold, E., Dianati, M., de Temple, R. & Fallah, S. Cooperative Perception for 3D Object Detection in Driving Scenarios using Infrastructure Sensors. *IEEE Transactions on Intelligent Transportation Systems,* 1–13. http://dx.doi.org/10.1109/TITS.2020.3028424 (2020).

A Acknowledgements

First of all, I want to thank Daniel Straub for being my doctoral supervisor and who provided me the opportunity to write my PhD thesis at the engineering risk analysis group. He has always been reachable when I required input, also on the phone during home office hours. Besides being my mentor at work, he was also a good companion for having a barbecue after work.
I want to thank Mario Berk who acted as my company's supervisor. I used many concepts from his PhD thesis to apply them to real-world data. He helped me in understanding the basics of his work and gave me access to resources that I required throughout the project.
I want to thank Boris Buschardt who also gave me insight from the company's perspective and who made the funding of the project possible. Thanks are also given to Erich Bruns who provided me with information during our company meetings.
Special thanks are given to my supervisor in the last 6 months, Tobias. As a long-time employee and the same background as a physicist, he introduced me to the differences between engineers and physicists. He always had an open ear for discussing equations and the interpretation of different metrics and measures.
I enjoyed working with Minhao who allowed me to see problems from a computer science perspective. Due to our overlapping fields of research, we helped us find and discuss relevant papers...without you, I would probably have missed many of them. Moreover, we motivated ourselves to write conference papers and I was thankful that he always helped me with proofreading. Furthermore, we helped generate results by sharing our code base.
I am particularly thankful to my two PhD colleagues Jan and Franzi for helping me a lot with the presentation of essential parts of this work. Both iterated with me many times over my figures to present the ideas for the definition of errors to the board member as intuitively as possible. I am sure that without the two my presentation of the work would have been way too complex.
I would also like to thank Valeria for our virtual meetings to write on the thesis from time to time. These meetings helped me a lot in gaining motivation and made the long home-office days much more joyful.
I want to thank the entire engineering risk analysis group for providing a very welcoming atmosphere whenever I was at the uni office. At the latest when we met at lunchtime any cloudy mood was gone. In particular, I want to thank Sabine for all her help regarding paperwork at TUM. I would also like to thank my two office roommates Antonis and Hugo, who made the long office hours pass much quicker and who occasionally helped me with their risk analysis background.

I would also like to express my gratitude towards my Hiwi Chenguang who helped me a lot in obtaining the object data from the Waymo dataset for my further analysis.
At this point, I also want to thank the PhD coordinator Helena for motivating me to present my work in front of the board members. Moreover, I am really thankful for the help in all organizational topics related to the PhD project at the company. In the latter matter, I also want to thank Sandra and Anna.
I thank my climbing partners Chris and Jana for being always present for physical exercise on the climbing wall which helped me to keep me going.
Finally, I want to thank my family, in particular my mum who kept motivating me throughout the time of my PhD. I also want to express my gratitude towards my godfather and his wife in this regard.

B Symbols

α	Threshold that specifies the minimum requirement for association using an association measure. The threshold can either be a constant or a function of the size of or the distance to the object
α_{dd}	Relative distance dependent error that defines the allowed distance dependent deviation ΔR in radial direction, also described as error constant.
γ	Solid angle of an object.
$\Delta\varphi$	Difference in the azimuth angle in polar coordinates (for example between two LiDAR measurement)
Δh_s	Height difference that defines maximum slope. See Figure 3.22 for further details.
Δh_t	Height difference that defines maximum threshold. See Figure 3.23 for further details.
ΔR	Distance dependent deviation.
Δr_s	Length difference that defines maximum slope. See Figure 3.22 for further details.
Δt	Time difference between two subsequent time frames.
λ	Wavelength of an electromagnetic signal.
λ_{act}	Failure rate of the actuators of the vehicle like motors.
λ_{plan}	Failure rate of the trajectory planning.
λ_{sense}	Failure rate of the environment perception.
λ_{sys}	Failure rate of the system, which is the automated vehicle.
$\hat{\mu}_{d,\tau,t}$	Expected average number of detections.
$\hat{\mu}_{o,\tau,t}$	Expected value of the average number of objects per time frame.
ρ	Euclidean distance between two points in the 2D bird's-eye-view perspective e.g. between a detected object and a reference object or the distance between the ego vehicle and another object.
τ	Threshold that specifies the minimum requirement for association using an association measure. The threshold can either be a constant or a function of the size of or the distance to the object
φ	Azimuth angle (for example of a LiDAR measurement)
φ_{dev}	Constant angular deviation.
φ_{img}	Angular resolution utilized for generating plots.
A	Representation of a set of objects according to set theory.

B	Representation of a set of objects according to set theory.
b	Baseline distance between two eyes or between two cameras of a stereo system.
B_{2D}	2D bounding box that encircles an object from the bird eye perspective. This bounding box is described by four parameters, its position x_{2D}, y_{2D}, its size w_{2D} and l_{2D} and an angle of rotation relative to the ego vehicle γ_{2D}.
$B_{2D,ref}$	Reference 2D bounding box that encircles an object from the bird eye perspective. This bounding box is described by four parameters, its position x_{2D}, y_{2D}, its size w_{2D} and l_{2D} and an angle of rotation relative to the ego vehicle γ_{2D}.
B_{3D}	3D bounding box. This bounding box is described by four parameters, its position x_{3D}, y_{3D}, z_{3D}, its size w_{3D}, l_{3D}, h_{3D}, an angle of rotation relative to the ego vehicle and a confidence value.
$B_{3D,ref}$	Reference 3D bounding box. This bounding box is described by four parameters, its position x_{3D}, y_{3D}, z_{3D}, its size w_{3D}, l_{3D}, h_{3D}, an angle of rotation relative to the ego vehicle and a confidence value.
B_C	Contour of a bounding box from the bird eye perspective.
$B_{C,ref}$	Contour of bounding box $B_{2D,ref}$.
B_{img}	2D bounding box that encircles an object in an image while the sides of the sides of the bounding box are parallel to the image borders. This bounding box is described by four parameters, its position x_{img}, y_{img} and its size w_{img} and h_{img}
$B_{img,ref}$	Reference 2D bounding box that encircles an object in an image while the sides of the sides of the bounding box are parallel to the image borders. This bounding box is described by four parameters, its position $x_{img,ref}$, $y_{img,ref}$ and its size $w_{img,ref}$ and $h_{img,ref}$
C	Representation of a set of objects according to set theory.
C	Consequences.
c	Diagonal of the smallest possible bounding box that includes the detection and the reference box.
f	Focal length.
h_{3D}	Height of a 3D bounding box.
$h_{3D,ref}$	Height of a reference bounding box.
h_{img}	Height of a 2D bounding box that encircles an object in an image measured in pixels.
$h_{img,ref}$	Height of a reference bounding box that encircles an object in an image measured in pixels.
l_{2D}	Height of a 2D bounding box that encircles an object from the bird eye perspective.
$l_{2D,ref}$	Height of a reference bounding box that encircles an object from the bird eye perspective.
n_t	Number time frames in a dataset.
p	Precision.

p_D	Distance weighted precision.
r	Recall.
$\mathbf{r}$	Center point position of a detected object (vector).
r_{2D}	Distance to a point from ego position to a detected obstacle in the 2D bird's-eye-view perspective (scalar).
$r_{2D,ref}$	Reference distance to a point from ego position to an obstacle in the 2D bird's-eye-view perspective (scalar).
r_D	Distance weighted recall.
$\mathbf{r}_{ref}$	Center point position of a reference object (vector).
v	Function that accounts for the aspect ratio of detection and reference object according to [53].
w_{2D}	Width of a 2D bounding box that encircles an object from the bird eye perspective.
$w_{2D,ref}$	Width of a reference bounding box that encircles an object from the bird eye perspective.
w_{3D}	Width of a 3D bounding box.
$w_{3D,ref}$	Width of a reference bounding box.
w_{img}	Width of a 2D bounding box that encircles an object in an image measured in pixels.
$w_{img,ref}$	Width of a reference bounding box that encircles an object in an image measured in pixels.

C Acronyms

AAE average attribute error . . . 82
AMOTA average multi object tracking accuracy . . . 85
AMOTP average multi object tracking precision . . . 83
AOE average orientation error . . . 82
AOS average orientation similarity . . . 82
AP average precision . . . 76
APD distance weighted average precision . . . 79
AR average recall . . . 80
ASE average scale error . . . 82
ATE average translation error . . . 82
AUC area under curve . . . 81
AVE average velocity error . . . 82
CIoU Complete Intersection over Union . . . 26
CVC Convex Visible Contour . . . 29
DBSCAN Density-Based Spatial Clustering of Application with Noise . . . 35
DIoU Distance Intersection over Union . . . 25
EM expectation maximization . . . 93
FM frequency modulated . . . 8
FN False Negative . . . 12
FOV field of view . . . 6
FP False Positive . . . 7
GIoU generalized Intersection over Union . . . 12
GPU graphical processing unit . . . 124
HOTA higher order tracking accuracy . . . 87
IoU Intersection over Union . . . 12
LiDAR Light Detection And Ranging . . . 5
mAOE mean average orientation error . . . 83

mAP mean average precision . . . 78
mAR mean average recall . . . 81
mATE mean average translation error . . . 83
mAVE mean average velocity error . . . 83
MC Markov chain . . . 93
MCMC Markov Chain Monte Carlo . . . 93
MLE maximum likelihood estimate . . . 17
MOT Multi-Object-Tracking . . . 34
MOTA multi object tracking accuracy . . . 84
MOTP multi object tracking precision . . . 82
MSE mean squared error . . . 94
MTBF mean time between failures . . . 72
NDS nuScenes detection score . . . 84
NN neural network . . . 9
PFA Probability of False Alarm . . . 13
PMF probability mass function . . . 17
POD Probability Of Detection . . . 13
RADAR RAdio Detection And Ranging . . . 5
ROC receiver operator characteristic . . . 81
sAMOTA scaled average multi object tracking accuracy . . . 85
SD support distance . . . 28
SDE Support Distance Error . . . 28
SIMD single instruction multiple data . . . 124
SIMO single-input, multiple-output . . . 8
sMOTA scaled multi object tracking accuracy . . . 85
SOTIF safety of intended functionality . . . 4
TN True Negative . . . 12
TP True Positive . . . 12

D Bounding box contour

Equations (3.7) and (3.8) utilize the bounding box contour. Figure D.1 illustrates the derivation a bounding box contour from the 2D bird-eye few perspective. The bounding box area in the object coordinate system is described by the set $B_{O,A} = [-l_{2D}/2, l_{2D}/2] \times [-w_{2D}/2, w_{2D}/2]$. The transformation from the coordinate frame of the object bounding box to the coordinate frame of the ego vehicle is provided by

$$
\begin{aligned}
T :&\mathbb{R}^2 \to \mathbb{R}^2 \\
&\begin{pmatrix} x_O \\ y_O \end{pmatrix} \mapsto \begin{pmatrix} \cos(\gamma_{2D}) & -\sin(\gamma_{2D}) \\ \sin(\gamma_{2D}) & \cos(\gamma_{2D}) \end{pmatrix} \begin{pmatrix} x_O \\ y_O \end{pmatrix}
\end{aligned}
$$

The area of the bounding box in the coordinate frame of the ego vehicle is then provided by $B_A = T(B_{O,A})$. The contour of the bounding box in the ego position coordinate frame corresponds to $B_C = \partial B_A$. Actually, instead of using B_C in equation (3.7) one could also enter B_A due to the fact that the minimum of the set B_A lies on its boundary $B_C = \partial B_A$.

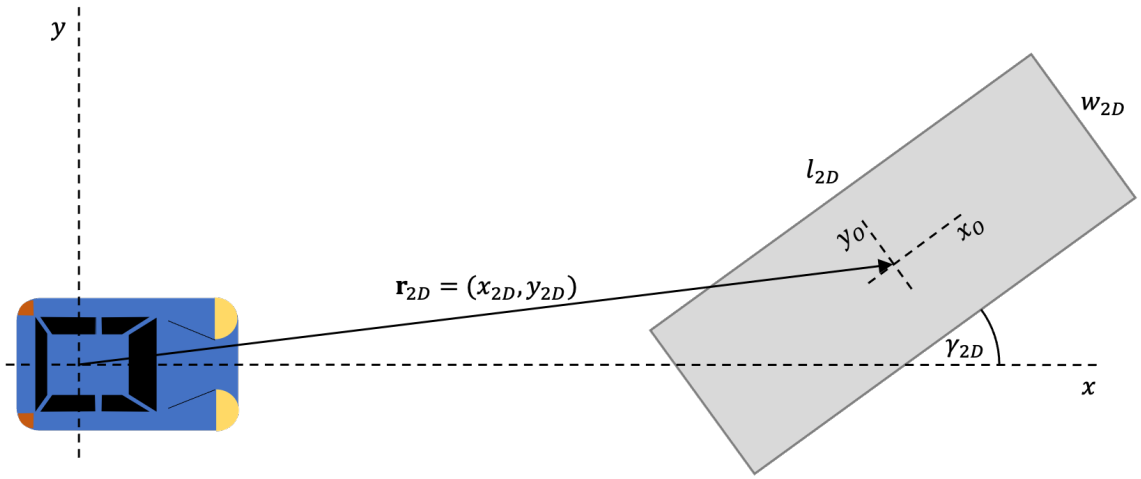


Figure D.1: Evaluation of the bounding box contour which is utilized in some association measures.