MSc thesis

Ultrasonic perception for autonomous vehicles

J.L.P. Spruit

2023

MSc thesis

Ultrasonic perception for autonomous vehicles

by

Johannes Leonardus Petrus Spruit

to obtain the degree of

Master of Science
in Mechanical Engineering

at the Delft University of Technology,
to be defended on Tuesday August 29, 2023 at 15:00

Student number:	1526286	
Thesis committee:	Prof. dr. D.M. Gavrila	TU Delft, supervisor
	Dr. H. Caesar	TU Delft

Mechanical, Maritime and Materials Engineering
Delft University of Technology
The Netherlands

Acknowledgements

Useful advise in life often comes from people close to you: parents, grandparents, teachers. Sometimes however, it is handed to you by a complete stranger:

"Never look back until you've reached the finish line, just keep pushing as hard as you can."

This advise was given to me years ago, in a completely different context. However, I feel it still applies to the last months working on this thesis. Pushing it to the finish line.

Now that the finish line is reached, we can look back. While this thesis took longer than anyone would prefer, I think it changed both the end result as well as myself for the better. There were the usual ups, and the usual downs, then more ups, more downs, and even more. But all the while it was being pushed closer to the finish line.

Reaching this finish line would not have been possible without the help of numerous people. First and foremost, I would like to thank Prof. Gavrila for his academic guidance. Thanks to Karin van Tongeren and Dr. Kooij for highlighting the importance of finishing this thesis.

I would like to thank everyone at *2getthere* for the warm welcome I got at the company. The support you provided during this thesis was amazing. Dimitrios and Sjoerd, thank you for highlighting the importance of this project within the company and continuously showing your faith in me. Remco and Tom, without your work on the hardware this thesis would not have been possible. Thanks to Bob, who, on his first week, was tasked with helping me out setting up the ultrasonics. Fortunately, his job got better after this. And a very special thanks to Athanasios and Joris, thanks for your continuous support and guidance, always asking the right questions.

Pap and mam, finally I can announce it is truly "almost done". Without your unconditional love and support this would not have been possible. Thanks for teaching me the important things in life. A special thanks to my niece Do, unfortunately not all contributions you provided with "Boris de Bulldozer" and "Petra het Paard" made it into the final version, but you definitely showed that typing a thesis can also be fun.

Finally, I would like to thank Krista, my girlfriend and soon to be wife. Your love, patience, and support guided me through this. Thanks for always knowing what I need. Finally, no more typing during the weekends, let's fully enjoy them together again.

And thanks to the kind stranger for his advise. Since then, it has been applied successfully on multiple occasions. Although he probably will never know.

Abstract

With recent advancements in autonomous driving, the demand for precise and accurate perception systems has increased. Perception of the vehicle's environment is a key element in ensuring safe operation. Due to their wide aperture angle and low cost, ultrasonic sensors are a viable option for achieving close-range 360° perception around the vehicle. This thesis investigates and evaluates the use of ultrasonic sensors for detection and tracking of objects and vulnerable road users in an SAE-level 4 autonomous vehicle. The thesis focuses on optimizing ultrasonic sensors for their designated measurement task and optimizing the positions of ultrasonic sensors on the vehicle. Furthermore, it explores different detection and tracking methods to investigate whether an ultrasonic detection system is indeed a viable candidate for close-range detection and tracking.

Contents

Chapter 1

Introduction

With recent advancements in autonomous driving, the demand for precise and accurate perception systems has increased, as vehicles take over more and more tasks from the driver. Accurate and precise perception of the vehicle's environment is a key element in ensuring safe operation of the vehicle. The drive for autonomous vehicles has lead to an increase in the performance of state-of-the-art perception sensors, like radar and lidar. This thesis aims to contribute to the field of perception for autonomous vehicles by exploring the use of ultrasonic sensors for detection and tracking of objects and road users in close proximity of the vehicle.

Compared to radar and lidar sensors, ultrasonic sensors are simple and inexpensive. Nowadays, these small sensors can be found on almost any new vehicle, as they are widely used in parking assistant systems and automated parking systems. However, the use of ultrasonic sensors in the driving function is generally limited to these parking related tasks. For general driving radar, lidar, and camera systems seem to be the preferred option.

A drawback of lidar and radar sensors is their minimum detection range. These sensors typically have a minimum detection range of 1 m. However, due to the low propagation speed of sound in air (compared to the propagation speed of EM waves of radar and lidar), the minimum detection range of ultrasonic sensors is typically in the range of 0 cm to 30 cm, with a maximum detection range of 2.5 m to 8 m.

In SAE level 4 and 5 autonomous driving, full awareness of the environment is a key element for safe operation. Achieving full awareness of the environment with radar and lidar sensors can be challenging and expensive. Due to their low minimum detection range and wide aperture angle, achieving 360° close range sensor coverage around the vehicle with ultrasonic sensors could be a feasible option.

In this thesis the use of ultrasonic sensors as a complement to lidar and radar sensing in autonomous vehicles is investigated. More precisely, the deployment of an ultrasonic detection system for an SAE level 4 vehicle is evaluated. This vehicle functions as a public transportation means in both segregated and mixed-traffic environments. The vehicle is equipped with various state-of-the-art sensors for autonomous driving, such as radar and lidar. Since these sensors are mainly intended for the driving function, they are mounted in the direction of travel of the vehicle, as show in Figure 1.1. The intention of adding ultrasonic

sensors to the sensor set is to achieve 360° sensor coverage in the close range around the vehicle, as shown in Figure 1.2.

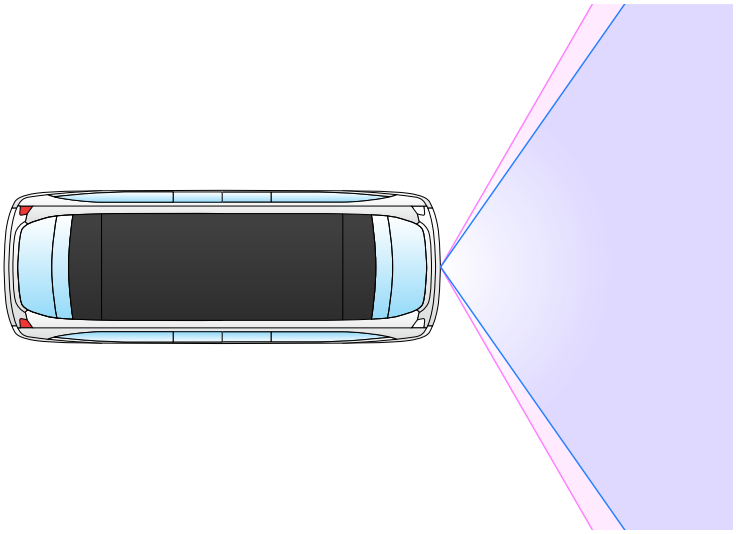


Figure 1.1: Top-down illustration of the vehicle with the fields-of-view of the different sensors of the sensor set intended for the driving functions. The field-of-view of the radar (pink) and lidar (blue) mainly cover the area in the direction of travel of the vehicle.

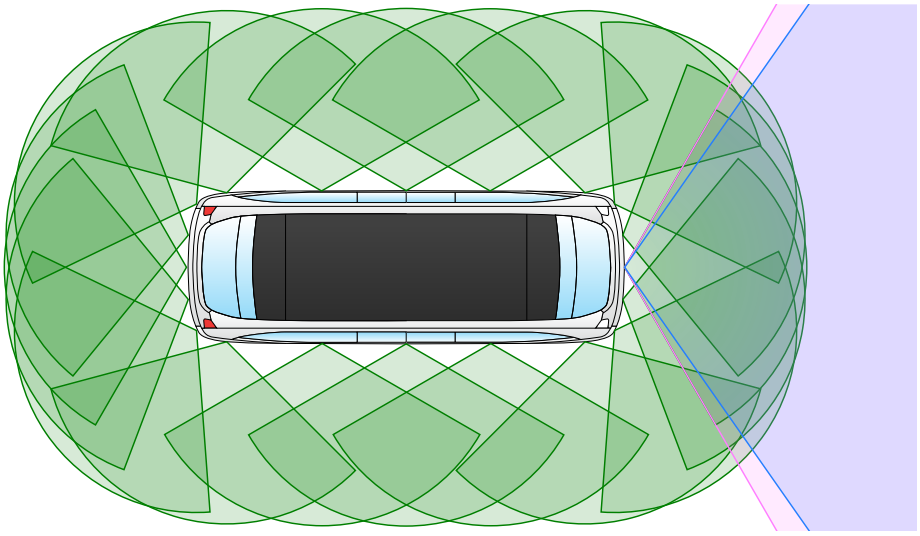


Figure 1.2: Top-down illustration of the vehicle with the fields-of-view of the different sensors of the sensor set of the vehicle. With the addition of ultrasonic distance sensors, a 360° close range sensor coverage can be achieved. Due to the wide aperture angle of automotive ultrasonic distance sensors, the entire area around the vehicle can be covered with a limited amount of sensors.

The two operational design domains of the vehicle: segregated operation

and mixed-traffic operation, each have their own characteristics. In segregated operation the vehicle operates on a designated track, where access for other road users is prohibited. However, at intersections other traffic can cross the lanes of the dedicated track. Other road users, especially pedestrians, could also unintentionally enter the track. In these situations the vehicle should be able to detect any other road users and act accordingly. Whenever the vehicle stops at a station, passengers can enter and exit the vehicle. In these stop-start scenarios, the vehicle has to be able to detect and track passengers and pedestrians moving around the vehicle, in order to ensure that it is safe to accelerate after a stop.

In mixed-traffic operation the vehicle operates in normal city traffic, and thus encounters all types of road users. Here the vehicle should be able to detect and track all road users in close proximity around the vehicle. In addition to normal stop-start scenarios of mixed-traffic operation the vehicle also stops at dedicated stations to let passengers exit and enter the vehicle. In all these scenarios, the vehicle should be able to detect and track all, especially vulnerable, road users.

This thesis investigates and evaluates the use of multilateration for an ultrasonic sensor system for the detection and tracking of (vulnerable) road users. In the introduction of this thesis, an introduction of automotive ultrasonic sensors is presented and the challenges of automotive ultrasonic distance sensors are discussed. In Chapter 2 the research questions of this thesis are presented. Chapter 3 gives an overview of state-of-the-art methods in sensor calibration, sensor positioning, object detection, and object tracking. Chapter 4 proposes a method to calibrate ultrasonic sensors for use in autonomous driving. Furthermore, a method for the positioning of sensors around the vehicle is proposed. A method for determining detailed object positions from the measurements is proposed. And finally, clustering and tracking algorithms are proposed for use with an ultrasonic detection system. Chapter 5 presents the results of the four different methods proposed in Chapter 4. Finally, Chapter 6 presents the conclusion of this thesis and gives an outlook for future work.

1.1 Ultrasound

Sound waves with frequencies above the upper audible limit of human hearing are called ultrasound or ultrasonic sound. The frequency spectrum of ultrasound extends from 20 kHz [1] and has no clear upper limit, as the upper limit depends on the medium in which the ultrasonic waves propagate. Between 1910 to 1925 Paul Langevin developed ultrasonic underwater detection for the French Navy, which was the first technological application of ultrasound [2]. Since Langevin's invention, ultrasound has spread to applications in the field of automotive [3], defense, industrial, medical [4] and security [5]. Ultrasound can be used for object detection, distance measurement, imaging in medicine, nondestructive testing, cleaning, and accelerating chemical processes. The position of ultrasound within the overall sound spectrum is shown in Figure 1.3.

1.2 Ultrasonic sensors

Ultrasonic sensors "use wave-propagation phenomena in air to measure physical or chemical variables" [7]. Ultrasonic sensors can generally be divided into

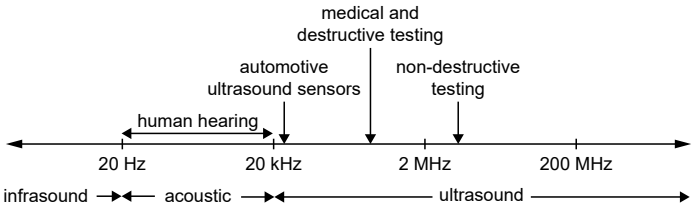


Figure 1.3: Frequency range of the sound spectrum, with an indication of several ultrasound applications within the spectrum. Figure based on [6].

two classes: contact and non-contact sensors. Contact sensors are in direct contact with the measurement subject. They utilize the propagation of ultrasonic waves through the subject to measure certain properties of the subject, like temperature, pressure, or density.

Non-contact ultrasonic sensors evaluate the propagation time of ultrasonic waves to determine the distance to an ultrasonic source or an object reflecting an ultrasonic wave. Non-contact ultrasonic sensors can be further divided into two categories: active and passive sensors. Passive ultrasonic sensors can only receive ultrasonic signals. Active ultrasonic distance sensors can emit ultrasonic signals and receive their echoes, which are reflected by objects within their measurement range.

1.3 Ultrasonic sensors in automotive applications

In the automotive field, ultrasonic distance sensors are typically found in parking assistance systems or automated parking systems of vehicles. These sensors measure distances to obstacles around the ego-vehicle. When an object appears in the vicinity of the ego-vehicle or the time-to-collision (TTC) to the object reaches a predefined threshold, an indication or warning is given to the driver in case of a parking assist system or the vehicle is stopped in case of an automated parking system. Warnings from a parking assist system are often acoustic, as defined in [8], sometimes accompanied by a visual representation of the situation around the vehicle on an in-vehicle display.

Due to the limited range of ultrasonic distance sensors, they are typically used in low speed scenarios [9]. However, [10]–[12] show that ultrasonic distance sensors can also be used at high velocities, for example in blind spot surveillance in motorway scenarios.

In order to cover a large observation area with a limited amount of sensors, automotive ultrasonic distance sensors have a wide field-of-view (FOV). For example, the sensors used in this thesis have an approximate FOV of $\pm$ 60° in the horizontal plane and $\pm$ 30° in the vertical plane [9]. Figure 1.4 shows the effective field of view of the sensor.

1.4 Automotive ultrasonic sensor function

Figure 1.5(a) shows a picture of an automotive ultrasonic distance sensor, with Figure 1.5(b) showing the different internal components of the sensor in a cross section. Figure 1.6 shows the different signals of the sensor when performing a

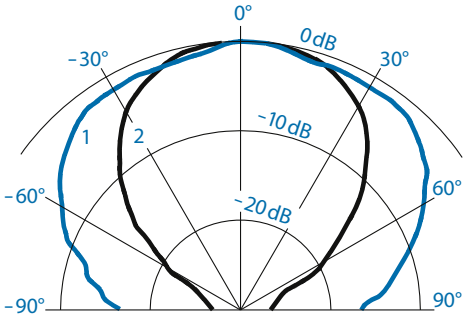
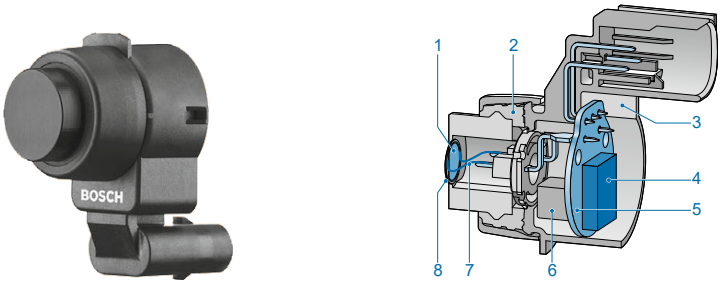


Figure 1.4: Horizontal (blue) and vertical (black) radiation pattern of an automotive ultrasonic distance sensor. Image from [3].

measurement. When the sensor receives a command to perform a measurement from the electric control unit (ECU) of the vehicle, the sensor generates a pulse with its piezo-ceramic element (labelled 1 in Figure 1.5(b)). This pulse stimulates the oscillation of the aluminium membrane (8) at its resonant frequency. A series of ultrasonic pulses is sent out by the alumunium membrane. Once the ultrasonic pulses are emitted and the occilation of aluminium membrane has decayed, the sensor turns into an ultrasonic microphone. The time it takes for the membrane to stop oscillating defines the minimum detection range of the sensor. When an ultrasonic pulse is reflected by an object, the aluminium membrane is stimulated. The oscillation of the membrane is converted back into an electric signal and compared to a predefined threshold value. If the electric signal exceeds the threshold value, a signal is sent to the ECU. From this signal, the ECU can calculate the time-of-flight (TOF) of the ultrasonic pulse, by determining the duration between the measurement command and the reception of the signal from the sensor.



(a) **(b)**

Figure 1.5: (a) Automotive ultrasonic sensor. (b) Cross section of an automotive ultrasonic sensor. 1: piezoceramics, 2: decoupling washer, 3: plastic housing with connector, 4: integrated circuit chip, 5: printed circuit board, 6: transducer, 7: bonding wire, 8: aluminium membrane. Images from [3].

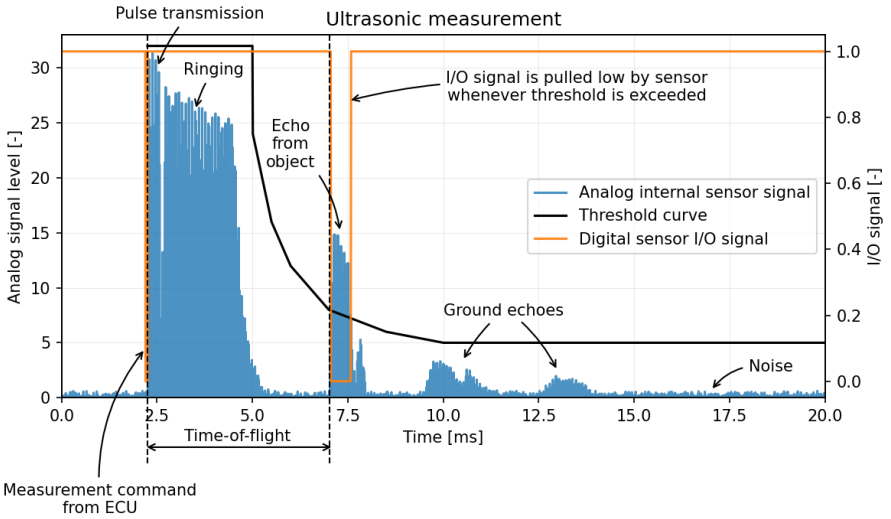


Figure 1.6: Example measurement cycle of an ultrasonic distance sensor. At $t = 2.2\,\text{ms}$ the sensor receives a command from the electronic control unit (ECU) to start a measurement cycle. After reception of this command, the sensor sends out a short burst of ultrasonic pulses. After this, it takes some time for the membrane to stop oscillating, during this time from $t = 2.5\,\text{ms}$ to $t = 4.5\,\text{ms}$ the sensor cannot perform any measurements. At $t = 7\,\text{ms}$ ultrasonic pulses reflected by an object hit the membrane. The vibration of the sensor membrane is translated into an analog electric signal (blue) by the piezoceramic element. Whenever the analog electric signal exceeds the time dependent threshold (black) the sensor switches the digital input/output signal to zero. From the time between the measurement command and switching of the I/O signal, the ECU can compute the distance to the object causing the echo. In this example multipath propagation ($t = 7.7$), ground echoes ($t = 10$, $t = 13$) and background noise do not cause a sensor output, as their amplitude is below the threshold.

The method to find the voltage of a received ultrasonic pulse is described in [13]. This method is summarized below, with equations (1.1) to (1.4) cited from [13]. The sound pressure level (SPL) of the ultrasonic pulses generated by the sensor are given by

$$\text{SPL} = 20\log_{10}\left(\frac{p_{\text{rms}}}{p_{\text{ref}}}\right) \tag{1.1}$$

where p_{rms} and p_{ref} are the RMS sound pressure and the reference sound pressure, respectively. The SPL depends on the distance between sensor and target, as the pressure of the sound level p is inversely proportional to the distance d

$$p \propto \frac{1}{d} \tag{1.2}$$

Besides this loss in pressure, additional losses occur due to absorption of the energy of the ultrasonic pulse in air and absorption by the object. The SPL of the echo received at the sensor can be approximated by

$$\text{SPL}_{\text{echo}} = \text{SPL}_{\text{transmitted}} - 20\log_{10}\left(\frac{2\cdot d}{30}\right) - 2\cdot\alpha\cdot d - \text{SPL}_{\text{absorbed by object}} \tag{1.3}$$

where α is the absorption coefficient of air. Given a known receiver sensitivity, the voltage produced by the receiver can be found

$$\gamma_{\text{echo rms}} = K \cdot p_{\text{ref}} \cdot \frac{30}{2 \cdot d \cdot 10^{0.1 \cdot \alpha \cdot d}} \tag{1.4}$$

with K as constant gain. Figure 1.7 shows the relation between receiver voltage and distance to the object. The voltage of the received signal strongly depends on the distance between the sensor and object. Therefore, the threshold value of automotive ultrasonic distance sensors is usually varied over time, following the shape of the curve in Figure 1.7. This ensures an even detection probability over the entire range of the sensor for a given object.

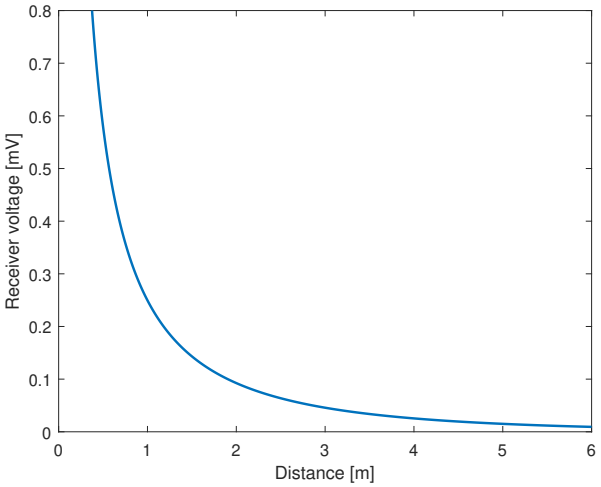


Figure 1.7: Example receiver voltage of an automotive ultrasonic sensor as function of distance. Given an ultrasonic pulse with a transmission SPL of 106 dB at 30 cm, air absorption of 1.3 dB m^{-1} and receiver sensitivity of −85 dB. Figure based on [13]

Given TOF t of the ultrasonic pulse, the distance to the object causing the echo can be found by

$$r = \frac{c \cdot t}{2} \tag{1.5}$$

with c the speed of sound propagating through air. The speed of sound in air depends on temperature T, humidity $\%RH$ and air pressure p_{ref} and is appriximated by [9]

$$c = 331.5 + 0.6 \cdot T \tag{1.6}$$

A more accurate approximation of the speed of sound in air is given by [14], here the speed of sound is described as a function of temperature, pressure, humidity, and CO_2 concentration. Python code for this approximation is given in Appendix A. Unless stated otherwise, this work uses the latter method to determine the speed of sound in air.

1.5 Problem description

While the wide aperture angle of automotive ultrasonic distance sensors is useful to cover a large observation area with one sensor, it also poses a challenge in the

detection of an object. Figure 1.8 shows a situation where an ultrasonic sensor is mounted on a vehicle. Area T (red) represents the trajectory of the base hull of the vehicle. This is the area the vehicle will occupy when it travels along its trajectory. If an object is present within T, it is assumed to be detected by the sensor in order to stop the vehicle or take evasive action. Object O is positioned outside area T. Since O is positioned within the FOV of the sensor the sensor measures distance r_1 to the object, producing hypothesis H_1, as shown in Figure 1.8. Hypothesis H_1 states that an object could be present within F at distance r_1 from the sensor. From H_1 it follows that O could possibly be in the path of the vehicle, since H_1 intersects trajectory T, although the object is positioned outside T. This situation causes a false positive detection and leads to an unnecessary stop of the vehicle, possibly causing passenger discomfort.

A conventional solution to prevent these false positive detections, is reducing the effective range of the sensor to the boundary of the trajectory of the base hull of the vehicle. This solution is shown in Figure 1.9, and prevents the false positive detection of object O.

Figure 1.10 also shows this solution, but now object O' is placed inside the trajectory of the base hull of the vehicle. Due to the reduction of the effective detection range of the sensor, O' is not detected. This results in a false negative detection of object O', which could lead to a collision of the vehicle with object O'. This is an event that should be avoided at all costs, as it may cause damage or harm.

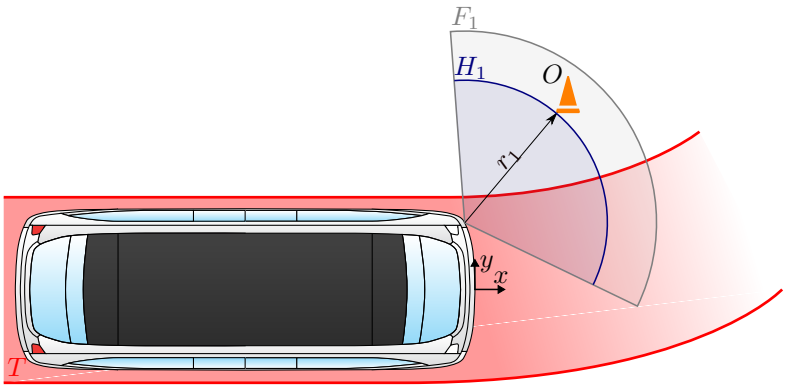


Figure 1.8: Object O is positioned outside trajectory T of the base hull of the vehicle and inside the field-of-view of the sensor F_1. The sensor measures distance r_1 to the object. Since only distance r_1 is known, the object could be present at any point within F_1 with at distance r_1 from the sensor, producing hypothesis H_1. From hypothesis H_1 it follows that object O could also be present in the trajectory of the base hull, as H_1 intersects with trajectory T.

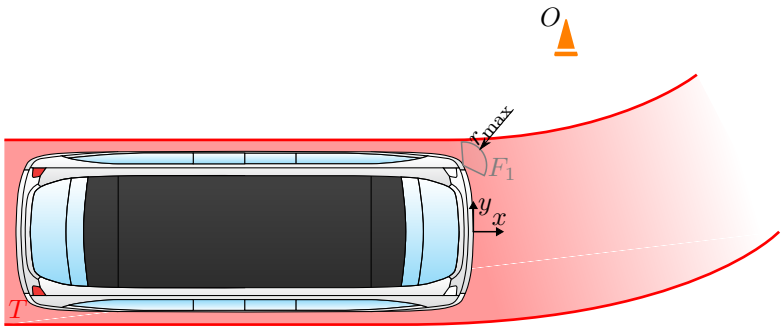


Figure 1.9: In order to prevent false positive detections of object O by the sensor as shown in Figure 1.8, the effective range $r_{\max}$ of the sensor can be reduced as not to extend trajectory T of the base hull of the vehicle.

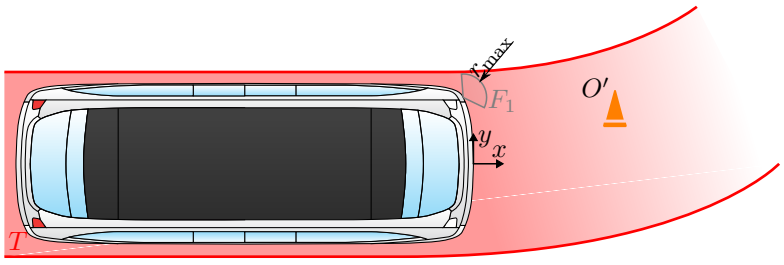


Figure 1.10: Object O' is placed inside the trajectory T of the base hull of the vehicle. With the effective sensor range of Figure 1.9, object O' cannot be detected, resulting in a false negative detection of O'.

A possible solution to this problem is the use of multilateration [15], with

multilateration the position of an object can be found using distance measurements of multiple sensors to the object. This means that instead of only having a known range r from sensor to object, the position of the object can be determined in the (x, y)-plane.

In Figure 1.11 a second sensor is introduced. This sensor measures distance r_2 to object O, from which hypothesis H_2 is constructed. Hypothesis H_2 states that there could be an object present within F_2 at distance r_2. With hypotheses H_1 and H_2, the position of object O' can be found by determining the intersection of H_1 and H_2. This solution allows the sensors to operate at their maximum detection range, while still being able to detect if the object is positioned inside or outside the trajectory of the base hull of the vehicle.

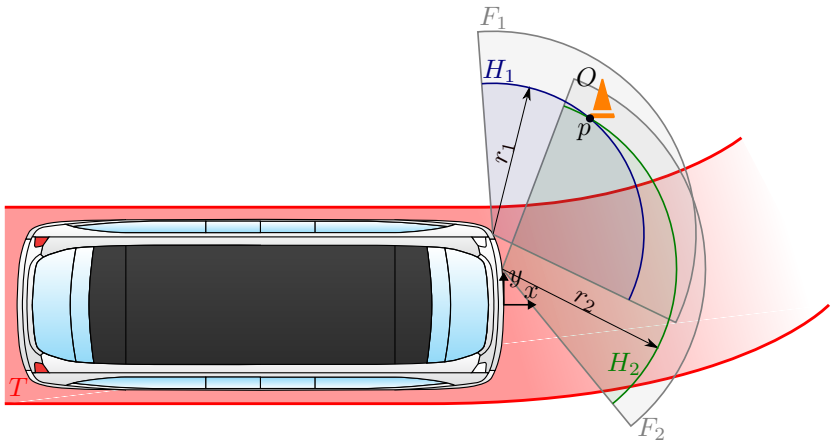


Figure 1.11: Adding a second sensor, similar to Figure 1.8, we measure the distance to the object with both sensors. This results in ranges r_1 and r_2, producing hypotheses $H1$ and H_2 respectively. The position of the object can be found at point p, the intersection of H_1 and H_2.

Figure 1.12 shows the effective range of an ultrasonic detection system using the conventional method of using only single distance measurements. Figure 1.13 shows the effective range of the same ultrasonic detection system, now using multiple sensor measurements, as required for the multilateration method. The effective range in direction of driving x, is nearly doubled for the system using the multilateration compared to using the conventional method.

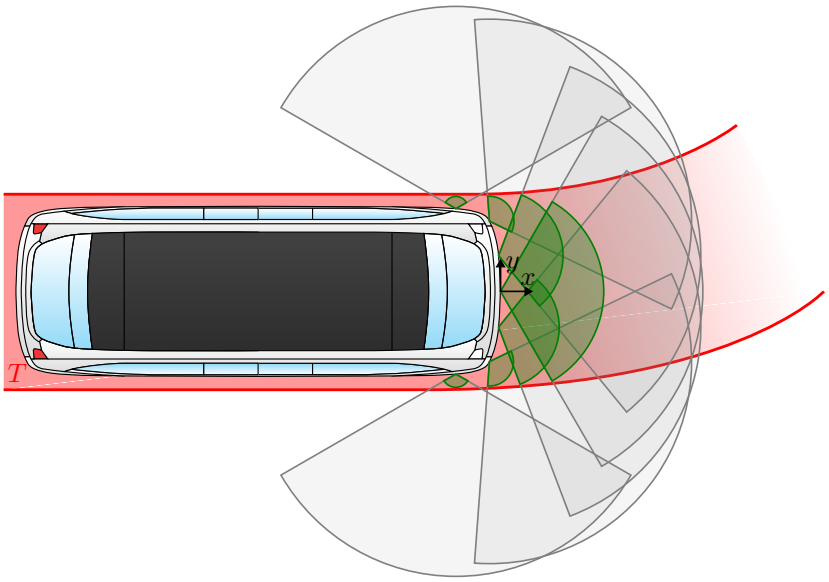


Figure 1.12: Effective range (green) of the ultrasonic system using single distance measurements. The effective range of the sensors has to be reduced to the boundary of the trajectory of the base hull of the vehicle, in order to prevent false positive detections.

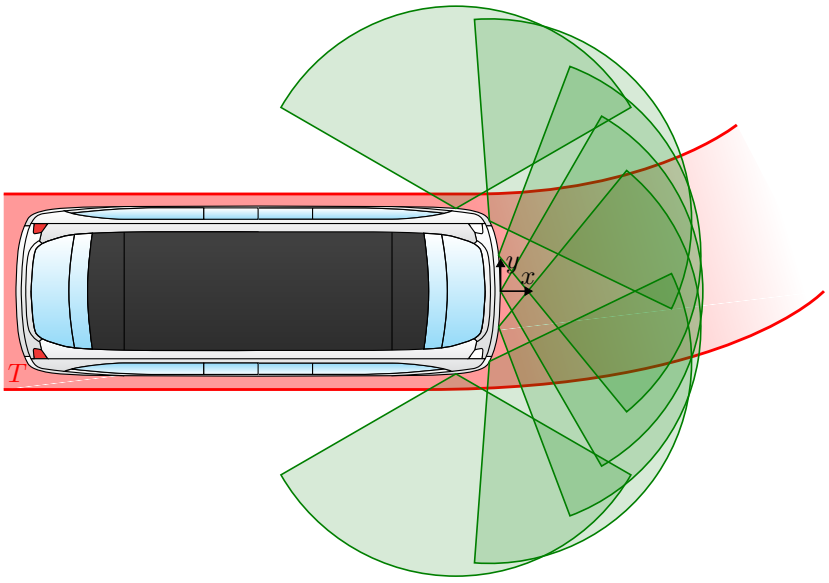


Figure 1.13: Effective range (green) of the ultrasonic system using distance measurements from multiple sensors simultaneously. The effective range of the sensors can be the maximum detection range of the sensor, as the object position can be determined in (x, y) coordinates.

Chapter 2

Research questions

In this chapter the research questions of this thesis are presented. The main question this thesis is intended to answer is:

Can ultrasonic sensors be used as a complement to conventional sensors for close-range detection and tracking in automated and autonomous vehicles?

To answer this question, additional sub questions are formulated. These research questions are divided into four parts: sensor calibration, sensor positioning, object detection, and object tracking, and will be presented in the following sections.

2.1 Sensor calibration

This thesis uses automotive grade ultrasonic range sensors. These sensors are typically calibrated to function in park assist systems or automated parking systems. As the requirements for autonomous driving in a public transport domain might differ from the requirements of a parking system, the following questions were formulated:

1. Can the detection performance of automotive ultrasonic sensors be improved for the operational design domain of the vehicle, by calibration with reference objects as ground truth, when compared to the conventional calibration method of only eliminating the ground echoes from the sensor detection?
   (a) Which parameters of the sensor can/should be calibrated to optimize the detection performance of the sensor?
   (b) What calibration methods can be used?

2.2 Sensor positioning

Similar to the previous question, the current sensor positions of the vehicle are evaluated in a case study. A method to improve the detection and tracking performance of the system, specifically the performance of multilateration, by repositioning of the sensors is evaluated.

1. Can the detection performance of the system be improved by positioning the sensors using an algorithm that searches for a global optimum, when compared to the conventional method of an engineer positioning the sensors based on their field-of-view?
   (a) Which algorithms are available for positioning sensors to improve the detection and tracking performance of the system?
   (b) What metrics are available to evaluate the detection and tracking performance of the system?

2.3 Object detection

As presented in the introduction, the use of multilateration could improve the detection range and precision of the ultrasonic detection system.

1. Which methods of multilateration could be used for an ultrasonic detection and tracking system?
2. What is the performance of these methods in terms of detection accuracy and runtime efficiency?
3. We propose a method intended to decrease runtime of the multilateration, how does this method influence performance in terms of detection accuracy and runtime efficiency?
4. Which methods can be used to remove noise from the ultrasonic detections?
5. Which methods can be used for the clustering of ultrasonic detections?

2.4 Object tracking

1. Which object tracking method has the best tracking performance of the detections obtained by multilateration of the ultrasonic measurements?

Chapter 3

Related work

This chapter provides an overview of the work related to this thesis. The related work is divided into four sections, each related to one of the four topics that this thesis covers.

3.1 Sensor calibration

As presented in the introduction, when an ultrasonic pulse is received by an automotive ultrasonic distance sensor, the amplitude of the resulting electric signal is compared to a threshold. This threshold is typically varied over time to allow for an even detection performance, as the energy of the reflected ultrasonic pulse from an object varies in time, as shown in Figure 1.7.

Instead of a threshold curve that is variable in time, [13] presents a method to obtain a fixed threshold value for an automotive ultrasonic distance sensor. The calibration of the threshold values is done by the manufacturer of the sensor, typically automotive ultrasonic range sensors do not require further calibration [16] or require limited calibration to ensure the functioning of the sensor after repair or replacement [17].

As automotive ultrasonic distance sensors are typically used in tasks related to parking, such as parking assist systems or automated parking systems, the sensors are calibrated for these tasks. Requirements for parking assist systems and automated parking systems are given in [8].

In [18] the performance of several OEM and after-market systems for "back-over prevention" is evaluated. These systems consisted of eight "sensor-based" systems and three "visual" systems. Six out of eight "sensor-based" systems contained ultrasonic sensors, of which three solely use ultrasonic sensors and three use a combination of either radar or video sensors in combination with ultrasonic sensors. Several static and dynamic tests were performed with vehicles equipped with these systems. The test objects consisted of several generic objects, plus dummies representing a one year old child, a three year old child, and an adult male. According to the study *"sensor-based systems generally exhibited poor ability to detect pedestrians, particularly children, located behind the vehicle. Systems' detection performance for children was inconsistent, unreliable, and in nearly all cases quite limited in range"*. Further finding include that the *"response times of sensor-based systems ranged from 0.18 to 1.01 seconds"*. Four

out of seven systems were found to have a response time above the ISO limit of 0.35 s [8]. This lead to the conclusion that *"detection ranges exhibited by the systems were not sufficient to prevent many collisions with pedestrians or other objects"*.

3.2 Sensor positioning

The effect of sensor position on detection and tracking performance is a topic studied in various fields. The detection and tracking performance of sensor systems can be quantified with several metrics.

A straightforward approach to sensor positioning, is covering the entire observation area of the system with a predetermined amount of sensors. In [19] an algorithm is described to determine position and orientation of directional sensors, such that an observation area or set of target points is fully covered by at least k sensors, this metric is introduced as k-coverage by the authors.

In [20] the effect of radar sensor position on the covariance of the position found by multilateration of measurements is evaluated. The resulting location dependent covariances are used in a target tracker.

The Cramer-Rao lower bound (CRLB) can be used to quantify the tracking performance limits of sensors, and combinations of sensors, in autonomous driving [21]. The CRLB is a theoretical lower limit on the variance of an unbiased estimator, and can be used as a benchmark to compare the mean square error of positioning algorithms [15]. In [22] the CRLB is used to optimize the positions of acoustic underwater range sensors in a system for target localization.

An optimization method wherein a cost function is minimized is proposed in [23]. The cost function consists of several elements: coverage of the region-of-interest, k-cover as described in [19], dilution of position, and point spread function. If needed, a penalty can be added to the cost function to limit the area wherein the sensors can be positioned.

3.3 Object detection

To obtain qualitative detections from the range measurements of ultrasonic sensors, several steps need to be taken into consideration. Multilateration, as described in the introduction, is a method of obtaining a detection in $n-1$ dimensions from a sensor set of n sensors. Several general solutions to obtain a target position from time or distance based sensor measurements are summarized in [15]. In [24] an exact solution for obtaining the target position from ultrasonic range sensors is implemented in a perception algorithm for a mobile robot.

Van Kleef, et al. [20] describe the data association problem that arises when two radar sensors detect multiple objects simultaneously. The proposed solution to solve this data association problem, is to increase the number of sensors from two to three, and extend the used multiple hypothesis tracking (MHT) algorithm with hypotheses for false detections.

In order to simultaneously reduce the number of detections and separate noise from detections, clustering algorithms can be used. DBSCAN [25], HDBSCAN [26], and OPTICS [27] are three popular density-based clustering algorithms, which can differentiate between detections and noise. These algorithms

all identify dense regions of data points based on their proximity to one another. However, there are several differences between these algorithms that make them suitable for different types of datasets.

DBSCAN identifies dense regions of data points based on their proximity to one another. DBSCAN uses two parameters, ϵ and minimum number of samples. These parameters control the size of the neighborhood around a point and the minimum number of points required to form a dense region.

HDBSCAN is an improvement on DBSCAN, and also takes the density variation among different dimensions of the dataset into account. This means that it can handle datasets with varying levels of density across different dimensions [28]. HDBSCAN requires at least one parameter, the minimum number of samples.

Similar to DBSCAN and HDBSCAN, OPTICS also identifies dense regions of data points based on their proximity to one another. However, instead of using a fixed size neighborhood, it uses a variable-size neighborhood that adapts to the local density of the dataset. The ξ parameter in OPTICS determines the cluster boundaries based on the minimum steepness of the reachability plot. The minimum number of samples determines the required amount of samples in a neighborhood to be considered as a core point. [27], [29]

3.4 Object tracking

Assuming the state of the tracked object is a random variable with a Gaussian distribution we can use several object tracking methods. Several object tracking methods are described in [30]. Although this study primarily focuses on tracking of features in computer vision, a subset of tracking methods described are methods for tracking of point features. The point tracking methods are divided into two categories: deterministic and probabilistic. Probabilistic tracking methods for point features are the Kalman filter, joint probabilistic data association filter (JPDAF), and probabilistic multi hypothesis tracking (PMHT).

An important subset of objects to track are vulnerable road users, like pedestrians and cyclists. Since especially pedestrians are likely to come in close proximity of the ego-vehicle, it is important to track them precisely and accurately. The nonlinear motion of pedestrians is described in [31], [32]. As the mounting height of automotive ultrasonic distance sensors is generally between 40 cm and 65 cm [33], primarily the legs of a pedestrian will reflect the ultrasonic pulses of the sensor and cause detections. The nonlinear motion of the different body parts of a pedestrian is described in [34]. This infers that a tracking method with a nonlinear transition might be required.

The extended Kalman filter (EKF) [35] and unscented Kalman Filters (UKF) [36] are Kalman filters that can track nonlinear systems. The EKF linearizes the system about a nominal state estimate, while the UKF uses the unscented transformation to approximate the probability density function of the state vector [37] without linearizing the system. In [38] the performance of an extended Kalman filter (EKF) and unscented Kalman filter (UKF) for tracking of moving objects with data from a linear array of eight ultrasonic sensors is evaluated. The input data for the tracking algorithms is the range measurements of the sensors.

The GM-PHD filter is a probabilistic data association (PDA) filter that uses

a Gaussian mixture model (GMM) to represent the probability density function of the target states [39]. The GM-PHD filter can handle situations where the number of targets is unknown and can adapt to changes in the number of targets over time [40], [41].

In [42] a survey of data association methods in multiple object tracking of vehicles and pedestrians is presented. Here probabilistic data association methods are divided into two global categories: Kalman filters and Particle Filters. The joint probablistic data association filter (JPDAF) [43] is often used in Kalman filters for data association. Being a multi-target extension of the probablistic data association filter (PDAF), the JPDAF uses a weighted average of all measurements within a tracks validation region to update the tracks state [42].

The global nearest neighbor (GNN) data association method is a straightforward data association method. The GNN does not use a probabilistic method evaluate different hypotheses, but instead assigns a single most likely hypothesis on a frame-by-frame basis [42].

Chapter 4

Methodology

4.1 Sensor calibration

In order to measure the distance between the sensor and an object, automotive ultrasonic distance sensors measure time-of-flightof an ultrasonic pulse. When the ultrasonic pulse is received by the sensor, it is converted into an electric signal. The voltage of the signal is measured and compared to a time dependent threshold voltage. When the measured voltage exceeds the time dependent threshold, the sensor gives an output, indicating a valid measurement. In this section, a method is proposed to optimize this time dependent threshold voltage curve based on measurement data.

Conventionally, the calibration of the time dependent threshold curve tries to suppress false positive detections caused by ground echoes, while keeping a high detection probability of the reference object. As automotive ultrasonic distance sensors are mainly used in parking systems, they are calibrated according to the requirements defined in [8]. These requirements state that the system must detect the reference object as defined by [8] for 90 % of the area from 0.2 m up to 0.6 m behind the vehicle, and 87 % for the area from 0.6 m up to 1 m behind the vehicle.

As stated in [18], the default calibration of automotive ultrasonic sensors might not suffice the requirements of a SAE level 4 or 5 system. In the proposed method, the goal of the sensor calibration is to maximize the detection probability of a ground truth object up to the maximum detection range of the sensor, which is 2.5 m. In order to achieve this, the threshold curve is adjusted such, that over the entire detection range the maximum detection probability is achieved. The proposed method aims to maximize the trade-off between true positive and false positive detections of the ground truth object. The elimination of ground echoes of the sensor signal are thus merely a byproduct of this optimization, and not a goal.

In order to optimize the time dependent threshold curve, the performance for each threshold level is determined. This allows for the selection of the most suitable threshold level for each range. The detection probability is measured by the ability of the sensor to detect a ground truth object. The ground truth object is a known object, and has a known distance to the sensor.

4.1.1 Datasets

In order to evaluate the detection probability of an object type, measurements of the object are collected and compared to the ground truth position of the object.

Object types

The ground truth objects are of the type *ISO pole* and *2gt object*. The *ISO pole* object is a cylinder of diameter 75 mm and 1000 mm length and is used as standard reference object for ultrasonic parking systems [8]. The *2gt object* is introduced as a more realistic representation of the leg of a child. The *2gt object* is a pole of diameter 75 mm and 160 mm length, covered in cloth material. While maintaining the same shape as the *ISO pole* object, due to its smaller size the *2gt object* will reflect less energy to the sensor. The cloth material of the *2gt object* adds a layer of material absorbing the energy of ultrasonic pulses, similar to the clothing of a pedestrian. Figure 3.2 and 3.3 show the *ISO pole* and *2gt object* respectively.

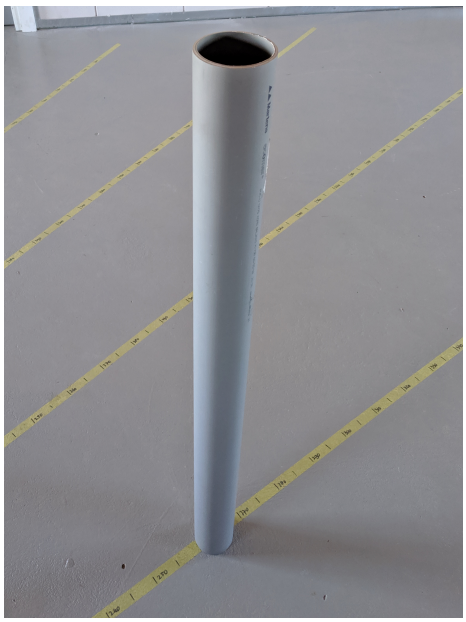


Figure 4.1: *ISO pole* ground truth object

Data collection

Since the exact field-of-viewof the sensor is unknown, area A_{FOV} is defined in polar coordinates as

$$A_{\text{FOV}} = \begin{Bmatrix} r = [0.1, 3.0] & \text{m} \\ \phi = [10, 170] & ^\circ \end{Bmatrix} \tag{4.1}$$

with radial distance r and azimuth ϕ. The sensor positioned at the origin of the coordinate frame, as shown in Figure 4.4. A set of positions X_{FOV} within A_{FOV}

Figure 4.2: *2gt* ground truth object

is defined at which ground truth object O is positioned.

$$X_{\mathrm{FOV}} = \begin{Bmatrix} r = \{0.1, 0.2, \ldots, 3.0\} & \mathrm{m} \\ \phi = \{10, 20, \ldots, 170\} & ^{\circ} \end{Bmatrix} \tag{4.2}$$

In order to collect measurement data, the ground truth object is positioned at each position $x(r, \phi)$ in X_{FOV} and for each sensor threshold level L, $N = 100$ measurement cycles are performed by the sensor. The resulting time-of-flight values are stored with the associated ground truth position. For each threshold level L, $N = 100$ measurement cycles without a ground truth object present are also performed by the sensor.

Ambient air temperature T, relative humidity $\%RH$, and pressure p are measured by environmental sensor E and stored for calculation of the distance from the time-of-flight measurements. Figure 4.3 shows a top-down schematic of the measurement setup.

4.1.2 Data classification

The collected time-of-flight measurements are converted into distance measurements r [14], [44], [45]. The distances measured by the ultrasonic sensor are assumed to be normal distributed around ground truth object distance r_{gt}, due to local measurement noise. The measurement value lies in the interval $[0, r_{\mathrm{max}}]$, with r_{max} the maximum detection range of the sensor [46].

In order to measure sensor performance, distance measurements r are compared to the reference distance to the ground truth object r_{gt}. For the detection of the ground truth, three different situations can occur: true positive detection, false positive detection, false negative detection. For a situation without

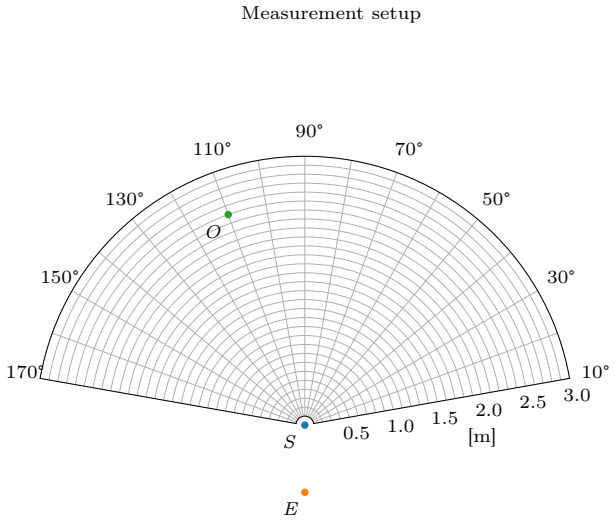


Figure 4.3: Top-down view of the measurement setup. Ultrasonic distance sensor S is positioned at the origin. Object O is positioned at $(r = 2.5\,\mathrm{m}, \phi = 110°)$. The set of possible object positions X, is represented by the grid, as O is positioned at each intersection point of the grid. At each position $N = 100$ time-of-flight measurements are taken by the sensor. The ambient air temperature T, relative humidity $\%RH$, and pressure p are measured by environmental sensor E and stored for compensation of the time-of-flight measurements.

ground truth object either a false positive detection or true negative detection can occur. Table 4.1 shows the confusion matrix for a detection.

Table 4.1: Confusion matrix for the detection of the ground-truth object

		True condition	
		True condition	*False condition*
Predicted condition	*Positive*	True positive	False positive (Type I error)
	Negative	False negative (Type II error)	True negative

Since the detection performance of the sensor is not uniformly distributed over its entire field-of-view, a subset of the field-of-view is defined in which the sensor is expected to have reliable detection of objects. This subset of the field-of-view is the region-of-interest. Figure 4.4 shows the area A_{ROI} of the region-of-interest.

Since A_{ROI} is a subset of A_{FOV} it follows that the ground truth object

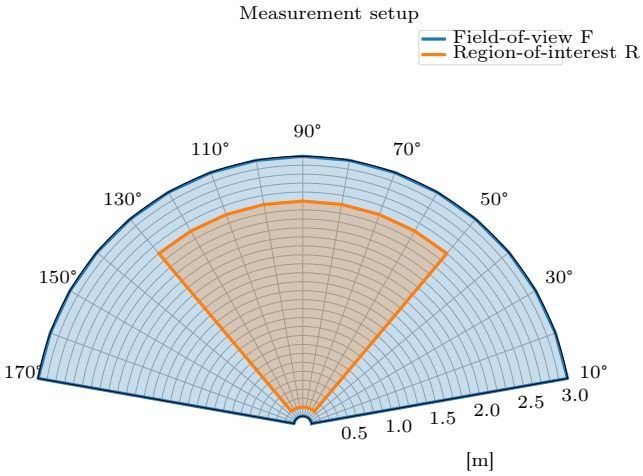


Figure 4.4: Area of the field-of-view A_{FOV} (blue) and region-of-interest A_{ROI} (orange).

positions inside the region-of-interest, X_{ROI} are a subset of X_{FOV}

$$X_{\text{ROI}} = \begin{Bmatrix} r = \{0.2, 0.3, \ldots, 2.5\} & \text{m} \\ \phi = \{-50, -40, \ldots, 50\} & ^\circ \end{Bmatrix} \tag{4.3}$$

True positive detection

A true positive detection is when the ground truth object is positioned at r_{gt} within the region-of-interest R of the sensor, and the sensor measures range r which falls within a predefined interval around r_{gt}.

$$TP = \begin{cases} 1 & \text{if } |r_{\text{gt}} - r| \leq d_{\max} : r_{\text{gt}}(r, \phi) \in A_{\text{ROI}} \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

False positive detection

A false positive detection is when there is a detection r by the sensor, but no ground truth object is present within the interval around r:

$$FP = \begin{cases} 1 & \text{if } r_{\text{gt}} \notin [r - d_{\max}, r + d_{\max}] : x \in A_{\text{FOV}} \\ 0 & \text{otherwise} \end{cases} \tag{4.5}$$

False negation detection

A false negative detection is when a ground truth object located at r_{gt} is within the region-of-interest R of the sensor, but is not detected by the sensor in the interval around r_{gt}.

$$FN = \begin{cases} 1 & \text{if } r \notin [r_{\text{gt}} - d_{\max}, r_{\text{gt}} + d_{\max}] : r_{\text{gt}} \in R \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

True negative detection

A true negative detection is when there is no detection by the sensor, and there is no object present in the region-of-interest R of the sensor.

$$TN = \begin{cases} 1 & \text{if } r = \{\} \text{ and } r_{\text{gt}} \notin R \\ 0 & \text{otherwise} \end{cases} \tag{4.7}$$

The validity interval around r_{gt} is defined as a function of the distance between sensor and ground truth.

$$d_{\max} = 0.05 r_{\text{gt}} \tag{4.8}$$

4.1.3 Threshold curve optimization

Given the collected datasets from Section 4.1.1 and definitions from Section 4.1.2, each range measurement r can be classified as either a true positive, false positive, false negative, or true negative detection. Each series of measurements $r(r, \phi, L)$ thus gives the true positive rate (TPR), false positive rate (FPR), and false negative rate (FNR), for position (r, ϕ) and threshold level L.

Since only the range to the object (and not the azimuth) is known to the sensor, the threshold curve is range dependent. Thus, the data for each range r has to be combined. A false positive detection is relevant for the entire field-of-view F of the sensor. On the other hand, true positive and false negative detections are only relevant to the region-of-interest R of the sensor.

$$TP(r, L) = \sum_{\phi \in R} \sum_{n=0}^{N} TP(r, \phi, n, L) \tag{4.9}$$

$$FP(r, L) = \sum_{\phi \in F} \sum_{n=0}^{N} FP(r, \phi, n, L) \tag{4.10}$$

$$FN(r, L) = \sum_{\phi \in R} \sum_{n=0}^{N} FN(r, \phi, n, L) \tag{4.11}$$

$$TN(L) = \sum_{n=0}^{N} TN(n, L) \tag{4.12}$$

Equation (4.9)–(4.11) gives the total amount of true positive, false positive, false negative, and true negative detections for each range r and threshold level

L, respectively. From this the true positive rate (TPR) and false positive rate (FPR) can be found for each threshold level.

$$\mathrm{TPR}(r, L) = \frac{TP(r, L)}{TP(r, L) + FN(r, L)} \tag{4.13}$$

$$\mathrm{FPR}(r, L) = \frac{FP(r, L)}{FP(r, L) + TN(L)} \tag{4.14}$$

The TPR and FPR are performance metrics for classifiers. For an ideal classifier TPR = 1 and FPR = 0. With the TPR and FPR metrics a receiver operating characteristic ROC curve can be plotted. The ROC curve is a graphical representation of the classifier performance for different threshold levels. For each threshold level, the ROC curve plots the TPR against the FPR.

By evaluating the TPR and FPR of each threshold level, the optimized threshold level L_{opt} can be found. The optimized threshold level corresponds to the threshold level where TPR $\to 1$, FPR $\to 0$, this corresponds to the point on the ROC curve closest to the top-left corner $(1, 0)$.

The optimized threshold curve is found, by finding the optimized threshold level L_{opt} for each range r.

4.2 Sensor positioning

Given a sensor with known parameters, as found in Section 4.1, it is possible to optimize the overall detection performance of a system consisting of multiple of these sensors. This section describes an algorithmic strategy for determining the optimal positioning of these sensors in order to maximize the detection performance of the system.

4.2.1 Positioning algorithm

The objective of the sensor placement algorithm is to maximize overall detection performance, precision, and accuracy of the system. Since the detection process of the system depends on the multilateration method, ideally each point in the ROI of the system should be covered by at least two sensors ($N > 2$). Increasing the coverage with even more sensors can improve detection accuracy.

The mean absolute error of the object position found by multilateration gives an indication on the accuracy, while the variance of the object position found by multilateration gives an indication on the precision. The Cramér-Rao lower bound (CRLB) determines the theoretical limit on the variance of any unbiased estimator.

Combining these metrics gives function f, which should be maximized.

$$f = \begin{bmatrix} N_{\mathrm{ROI}} > 2 & \mathrm{MAE}^{-1} & \mathrm{CRLB}^{-1} \end{bmatrix} \tag{4.15}$$

As the CRLB is the inverse of the Fisher Information Matrix (FIM) I, the cost functino can also be written as

$$f = \begin{bmatrix} N_{\mathrm{ROI}} > 2 & \mathrm{MAE}^{-1} & I \end{bmatrix} \tag{4.16}$$

The region-of-interest of the system is represented as a grid with $1\,\text{cm} \times 1\,\text{cm}$ cells. N_{ROI}, MAE, and I are computed for each cell in the region-of-interest of the system.

N_{ROI} represents the k-cover of a cell in the ROI, where

$$N_{\text{ROI}} = \begin{cases} n & \text{if covered by } n \text{ sensors with each at least 95 \% detection rate} \\ 0 & \text{otherwise} \end{cases} \tag{4.17}$$

The mean absolute error is the absolute error of the estimated position for the cell. The estimated position for the cell is found by the nonlinear least estimate of the position of the cell

$$\hat{\mathbf{x}} = \arg\min_{\hat{\mathbf{x}}} \; J(\mathbf{x}) \tag{4.18}$$

with

$$J(\mathbf{x}) = \sum_{l=1}^{L} \left(r_l - \sqrt{(x - x_l)^2 + (y - y_l)^2} \right)^2 \tag{4.19}$$

where L is the total amount of sensors having a valid measurement to the position of the cell.

The FIM is found by [15]

$$I(\mathbf{x}) = \left[\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right]^{\mathrm{T}} C^{-1} \left[\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right] \tag{4.20}$$

with

$$\left[\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right] = \begin{bmatrix} \frac{x-x_1^t}{d_2^t} - \frac{x-x_1}{d_1} & \frac{y-y_1^t}{d_1^t} - \frac{y-y_1}{d_1} \\ \frac{x-x_2^t}{d_2^t} - \frac{x-x_1}{d_1} & \frac{y-y_2^t}{d_2^t} - \frac{y-y_1}{d_1} \\ \vdots & \vdots \\ \frac{x-x_M^t}{d_M^t} - \frac{x-x_1}{d_1} & \frac{y-y_M^t}{d_M^t} - \frac{y-y_1}{d_1} \\ \frac{x-x_1^t}{d_1^t} - \frac{x-x_1}{d_2} & \frac{y-y_1^t}{d_1^t} - \frac{y-y_2}{d_2} \\ \vdots & \vdots \\ \frac{x-x_M^t}{d_M^t} - \frac{x-x_L}{d_L} & \frac{y-y_M^t}{d_M^t} - \frac{y-y_L}{d_L} \end{bmatrix} \tag{4.21}$$

where $\mathbf{x}^t$ are the transmitting sensors with distance d^t to the cell, and $\mathbf{x}_L$ are the receiving sensors with distance d to the cell. C is the covariance matrix. The CRLB for x and y are given by

$$\text{CRLB}_x = [I^{-1}]_{(1,1)} \quad \text{CRLB}_y = [I^{-1}]_{(2,2)} \tag{4.22}$$

Input data

Given a threshold curve $L(r)$ and the dataset found in Section 4.1, the input data for the sensor can be determined. This input data consists of mean range μ_r, standard deviation of the range σ_r, TPR, FPR, and FNR.

A model of the vehicle is used in order to determine valid sensor positions. Since the sensors in the system are typically positioned at approximately the same height, a 2 dimensional outline of the vehicle hull at this height is used. Optionally, areas where it is physically not possible to mount a sensor can be identified as such.

The region-of-interest of the system is defined as an area around the vehicle. The region-of-interest is represented as a grid with $1\,\text{cm} \times 1\,\text{cm}$ cells.

Parameters

The adjustable parameters are the sensor (x, y)-position and orientation around the z-axis.

Constraints

The sensors can be positioned anywhere along the outline of the vehicle hull, with exception of aforementioned areas identified as "keep-out" area. The orientation of the sensors can be set to fixed around their z-axis, i.e. perpendicular to the vehicle hull, variable within bounds, or fully variable. A perpendicular orientation of the sensor to the vehicle hull, or orientation of the sensor with limited angle to the normal of the vehicle hull, can be required for aesthetic reasons.

Algorithm description

Initially a set of N sensors is evenly distributed along the outline of the vehicle hull. The initial orientation angle of the sensors is perpendicular to the vehicle outline. The optimization variable f is computed for this configuration as an initial value.

Sensor positions and, if allowed, orientations are adjusted for the next iteration. Using gradient descent the algorithm iterates to find a maximum for f. If the stop conditions are met on reaching the maximum for f, results are stored and the algorithm is terminated. If the conditions to stop the algorithm are not met or a maximum for f is not found, an extra sensor is added to the configuration and the optimization process is continued.

Stop conditions

The condition to stop the algorithm is if a minimum coverage of two sensors for at least 95 % of the region-of-interest is reached.

4.3 Detection algorithm

For rudimentary obstacle detection, the range measurements of the ultrasonic sensors can be used without further postprocessing. However, if more detailed precision is required additional postprocessing steps will be required.

4.3.1 Multilateration

As explained in the introduction, multilateration is a method of estimating a position based on range measurements from multiple transmitters or receivers. Given n measurements, multilateration allows to find a position in $n-1$ dimensions (or n dimensions if ambiguities are allowed).

Multilateration using direct echo

Given two ultrasonic sensors, positioned at P_1 and P_2, as shown in Figure 4.5. Each sensor performs a measurement to object O, resulting in ranges r_1 and r_2, respectively. From r_1 and r_2 hypotheses C_1 and C_2 can be formulated, each describing a circle on which object O could be present. By finding the point(s) of intersection of the hypotheses, P_4 and P_4', an estimation of the position of object O in the (x, y)-plane is found.

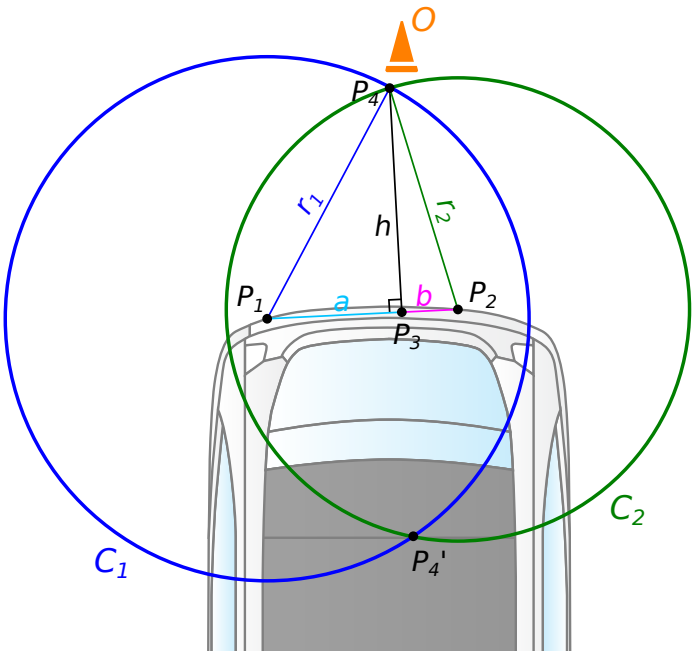


Figure 4.5: Multilateration using range measurements r_1 and r_2 from two sensors positioned at P_1 and P_2, respectively. The position of the object at P_4 can be estimated by finding the intersection of circles C_1 and C_2.

Nonlinear least squares solution A method of finding the estimated object position $\hat{\mathbf{x}}$, based on a set of independent range measurements $\mathbf{r}$ from a set of L sensors with known position, is solving the nonlinear least squares problem [15]

$$\hat{\mathbf{x}} = \arg\min_{\hat{\mathbf{x}}} \; J(\mathbf{x}) \tag{4.23}$$

with

$$J(\mathbf{x}) = \sum_{l=1}^{L} \left(r_l - \sqrt{(x - x_l)^2 + (y - y_l)^2} \right)^2 \tag{4.24}$$

where r_l is the range measurement from the sensor positioned at (x_l, y_l), and Given an initial estimated position, the nonlinear least squares problem should converge to an estimated object position.

Exact solution The estimated object position can also be found using an exact solution [24], [47]. Given circles C_1 and C_2 from Figure 4.5, the aim is to find intersections P_4 and P_4'. Given distance d between the circle centers

$$d = \|P_2 - P_1\| \tag{4.25}$$

The number of intersections is found by

$$N_{\text{intersection}} = \begin{cases} 0 & \text{if } d > r_1 + r_2 & \text{circles are separate} \\ 0 & \text{if } d < |r_1 + r_2| & \text{one circle contained in other} \\ \infty & \text{if } d = 0 \text{ and } r_0 = r_1 & \text{coincident circles} \\ 1 & \text{if } d = r_1 + r_2 & \text{circles touch} \\ 2 & \text{otherwise} & \end{cases} \tag{4.26}$$

Using the triangles formed by $P_1P_3P_4$ and $P_2P_3P_4$

$$a = \|P3 - P1\|, \qquad b = \|P3 - P2\|, \qquad h = \|P4 - P3\|$$

$$a^2 + h^2 = r_1^2 \text{ and } b^2 + h^2 = r_2^2 \tag{4.27}$$

solving for h gives $h = \sqrt{r_1^2 - a^2}$. Using $d = a + b$

$$a = \frac{r_1^2 - r_2^2 + d^2}{2d} \tag{4.28}$$

Now the coordinates of P_3 can be found by

$$\begin{aligned} x_3 &= x_1 + \frac{a}{d}(x_2 - x_1) \\ y_3 &= y_1 + \frac{a}{d}(y_2 - y_1) \end{aligned} \tag{4.29}$$

if $N_{\text{intersections}} = 1$, P_3 is the point of the intersection of circles C_1 and C_2. If $N_{\text{intersections}} = 2$ the coordinates of intersection points P_4 and P_4' are found by

$$\begin{aligned} x_4 &= x_3 \pm \frac{h}{d}(x_2 - x_1) \\ y_4 &= y_3 \pm \frac{h}{d}(y_2 - y_1) \end{aligned} \tag{4.30}$$

This method finds two points of intersection P_4 and P_4'. This ambiguity is easily solved since the sensors are directional. Only the point present in the intersection of the area of both sensors regions-of-interest is kept

$$P_{\text{intersection}} = P \in R \qquad \text{where} \qquad \begin{aligned} P &= \{P_4, P_4'\} \\ R &= R_1 \cup R_2 \end{aligned} \tag{4.31}$$

where R_1 and R_2 are the regions-of-interest of sensor 1 and sensor 2, respectively.

Multilateration using cross echo

Multilateration can also be performed on dependent measurements, i.e. cross echo measurements. As shown in Figure 4.6, the sensor positioned at P_1 emits an ultrasonic pulse. The ultrasonic pulse is reflected by object O and received by sensors positioned at P_2 and P_3. Measurement r_{12} creates a hypothesis which represents ellipse E_1 with foci P_1 and P_2, and the relation $r_{12} = 2a_1 = \|P_O - P_1\| + \|P_O - P_2\|$ where a_1 is semi-major axis of E_1. Measurement r_{13} creates a hypothesis which represents ellipse E_2 with foci P_1 and P_3, and the relation $r_{23} = 2a_2 = \|P_O - P_1\| + \|P_O - P_3\|$ where a_2 is the semi-major axis of E_2.

The use of cross echo eliminates the need for multiple consecutive measurements from different sensors. This means that the object position can be estimated in one measurement cycle.

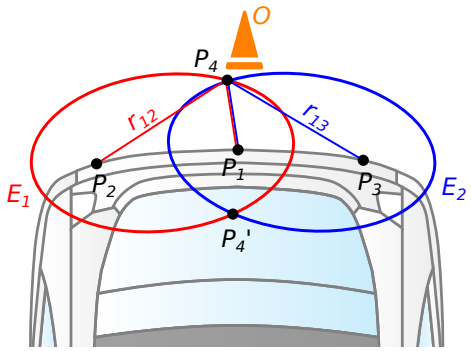


Figure 4.6: Multilateration using cross echo with an ultrasonic system consisting of three sensors positioned at P_1, P_2, and P_3 respectively. The sensor at P_1 emits an ultrasonic pulse, which is reflected by object O. The echo received by the sensor at P_2 creates a hypothesis consisting of ellipse E_1 with foci P_1 and P_2, and the relation $r_{12} = 2a_1 = \|P_O - P_1\| + \|P_O - P_2\|$ where a_1 is semi-major axis of E_1. The echo received by the sensor at P_3 creates a hypothesis consisting of ellipse E_2 with foci P_1 and P_3, and the relation $r_{23} = 2a_2 = \|P_O - P_1\| + \|P_O - P_3\|$ where a_2 is semi-major axis of E_2. The position of object O can be estimated by determining the intersection of E_1 and E_2.

Nonlinear least squares solution The estimated object position $\hat{\mathbf{x}}$ can be found by solving the nonlinear least squares problem [15]

$$\hat{\mathbf{x}} = \arg\min_{\hat{\mathbf{x}}} \; J(\mathbf{x}) \tag{4.32}$$

where $\mathbf{x}$ is the set of known sensor positions and $\mathbf{r}$ the set of range measurements, and

$$J(\mathbf{x}) = \sum_{m=1}^{M} \sum_{l=1}^{L} \left(r_{m,l} - \sqrt{(x - x_m)^2 + (y - y_m)^2} - \sqrt{(x - x_l)^2 + (y - y_l)^2} \right)^2 \tag{4.33}$$

where m is the set of transmitting sensors, and l is the set of corresponding receiving sensors. Given an initial estimate of the object position, the nonlinear least squares problem should converge to an estimate of the object position.

Exact solution An exact solution for the intersection of two ellipses also exists. As ellipses are conics, the general solution of finding the intersection points of two conics, as shown in [48], can be used. The intersection points can be found by combining the quadratic equations of both ellipses into a quartic equation (polynomial equation of fourth degree) and finding its roots.

This solution also works for circles (and combinations of ellipses and circles), as a circle is a type of ellipse with focal points $F_1 = F_2$. The full solution for finding the intersection points of two ellipses is provided in Appendix B.

Simplified exact solution The simplified exact method is based on the exact solution, but assumes $r << d$ where $r = \|P_1 - P_O\| + \|P_2 - P_O\|$ and $d = \|P_2 - P_1\|$, with P_1 and P_2 the foci of the ellipse and P_O any point on the ellipse. In other words, the foci of the ellipse are positioned close together compared to the size of the ellipse. From this assumption it follows that the definition of an ellipse with $r << d$, approaches the definition of a circle.

Therefore, this method converts any ellipse into a circle, where

$$P_C = \frac{1}{2}(P_1 + P_2) \tag{4.34}$$

is the center of the circle, and the radius of the circle is given by

$$r_c = \frac{1}{2}(\|P_1 - P_O\| + \|P_2 - P_O\|) \tag{4.35}$$

The estimated object position can then be found by the method from Section 4.3.1.

The main goal of this approach is the reduction of floating point operations for finding the intersection points, when compared to the exact ellipse approach. As the exact ellipse solution has a maximum of 650 floating point operations, where the exact circle solution has a maximum of 55 floating point operations.

Evaluation

All multilateration methods are evaluated on a theoretical scenario. A set of two sensors positioned at P_1 and P_2 is defined, as well as a region-of-interest R. Each point in R is taken as ground truth, a range measurement to the ground truth with additional Gaussian noise is generated for both sensors. From these range measurements, the estimated position of the ground truth is computed according to the respective method. This estimated position is compared to the ground truth by computing the absolute error. Furthermore, the computation times of all methods are compared.

4.3.2 Measurement association

The approaches described in Section 4.3.1 perform well when there is a single object present reflecting a single echo. However, when two or more objects are present in the field-of-view (or one object produces multiple echoes), a data association problem is introduced.

Given a system of two sensors and two objects, if both sensors receive an echo from each object, the number of combinations of measurements is $\binom{2+2}{2} = 4$.

Given that two measurements can have multiple points of intersection, positive detections can occur. Figure 4.7 shows an example of this.

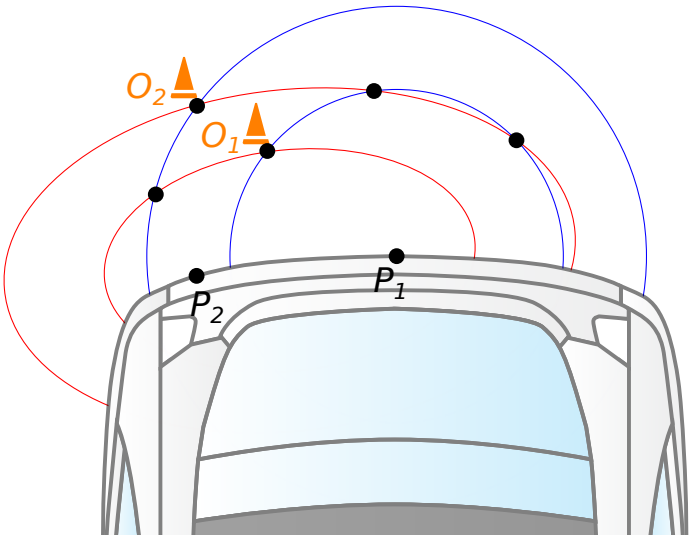


Figure 4.7: Multilateration using an ultrasonic system consisting of two sensors positioned at P_1, and P_2, respectively. The sensor at P_1 emits an ultrasonic pulse, which is reflected by objects O_1 and O_2. The echoes are received by both sensors, which creates two sets of two hypotheses each. Finding the estimated positions of the objects results in three false positive detections for this case.

The ambiguity resulting from the false positive detections can be reduced by adding a third sensor to the system. This adds two additional hypotheses, as shown in Figure 4.8. Instead of relying on the intersection of two measurements, the intersection of three measurements is considered as an estimated object position, which better corresponds to the actual object positions.

Since the range measurements may contain noise, the intersections of three circles or ellipses might occur at slightly different points, as shown in Figure 4.9. Nonlinear least squares solutions can still estimate the intersection point of three measurements with noise, but they do so for any arbitrary set of range measurements. To obtain a proper estimate of the object position, the range measurements provided as input to the nonlinear least squares solution can be filtered.

Filtering of range measurements on the input of the least squares solution is done by grouping range measurements with similar values together in a set. The least squares solution is then applied to each set of measurements. Filtering of results of the linear least squares solution is done by rejecting the solution if the residual sum of squared errors (SSE) exceeds a predetermined threshold $SSE_{\max}$.

When three noisy range measurements are provided as input to the exact (ellipse and circle) solutions, they will produce multiple intersection points. The validity of the intersection points can be assessed by the distance matrix of all intersection points. If a point does not have at least n neighbors within a predefined range r around it, the point could be rejected as a valid measurement.

Another approach is the use of clustering algorithms which are able to filter noise, such algorithms are DBSCAN, HDBSCAN, and OPTICS.

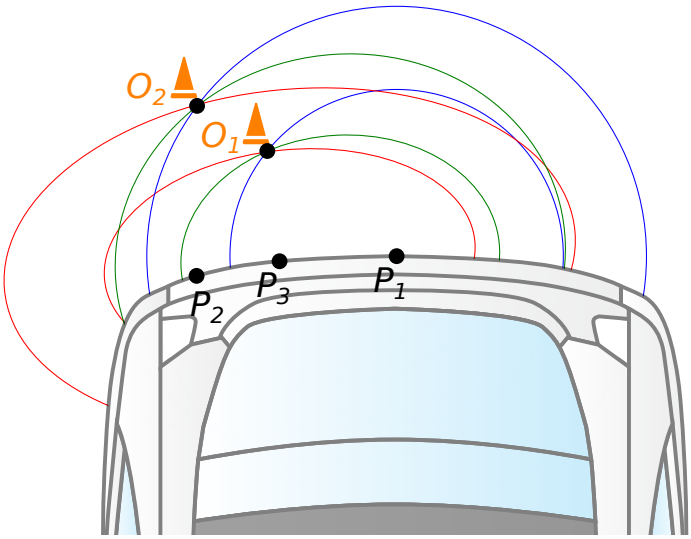


Figure 4.8: Multilateration using an ultrasonic system consisting of three sensors positioned at P_1, P_2, and P_3 respectively. The sensor at P_1 emits an ultrasonic pulse, which is reflected by objects O_1 and O_2. The echoes are received by all three sensors, which creates three sets of two hypotheses each. Finding the estimated positions of the objects results in three false positive detections.

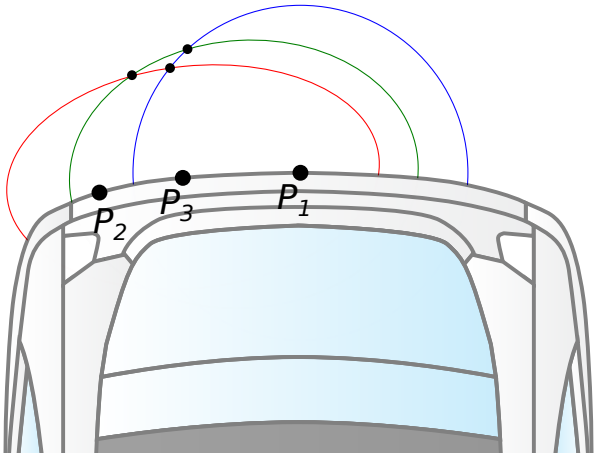


Figure 4.9: Multilateration using an ultrasonic system consisting of three sensors positioned at P_1, P_2, and P_3 respectively. Due to (exaggerated) noise in the range measurements, the intersections to not occur in the exact same point.

Evaluation

Given the proposed methods, three different approaches are proposed for evaluation:

- Nonlinear least squares solution with filtered input and filtered output
- Exact ellipse solution with distance matrix filter
- Exact ellipse solution with noise filtering by clustering algorithms

Each approach is evaluated on a real-world dataset in order to measure its performance. Before comparing the different approaches, the variables in each approach will be optimized.

Nonlinear least squares solution For the filtered input a value for the interval for which to group measurements together has to be determined. For the filtered output a value for the residual SSE has to be found.

Exact ellipse solution with distance matrix In order to filter noise based on their distance to other points, the values of number of points n that fall within radius r have to be determined.

Exact ellipse solution with DBSCAN In DBSCAN, ϵ determines the size of the neighborhood around each point. It controls how far away from a point a new point can be and still be considered part of the cluster. [25]

In order to find the optimized values for the different variables, precision and recall will be used as evaluation metrics.

$$\text{precision} = \frac{TP}{TP + FP} \tag{4.36}$$

$$\text{recall} = \frac{TP}{TP + FN} \tag{4.37}$$

Precision and recall are performance metrics for classifiers. In order to find the optimal value for the variable to optimize, both precision and recall need to be maximized simultaneously. This can be achieved by finding the point on the precision-recall curve, where the two metrics are equal. The precision-recall curve visualizes the relation between the precision and recall metrics and the threshold level.

Precision and recall are calculated from true positive, false positive, and false negative detections. A true positive detection is classified as a detection P which is positioned within a predefined range to a point belonging to the set of ground truth points Q.

$$TP = \begin{cases} 1 & \text{if } d \leq d_{\max} \text{ with } d = \min\limits_{k} \|P - Q_k\| \\ 0 & \text{otherwise} \end{cases} \tag{4.38}$$

A false positive detection is when there is a detection, but no ground truth point is present within the predefined range around the point:

$$FP = \begin{cases} 1 & \text{if } d > d_{\max} \text{ with } d = \min\limits_{k} \|P - Q_k\| \\ 0 & \text{otherwise} \end{cases} \tag{4.39}$$

A false negative detection is when a ground truth point does not have a detection point present within the predefined range.

$$FP = \begin{cases} 1 & \text{if } d > d_{\max} \text{ with } d = \min_k \|P_k - Q\| \\ 0 & \text{otherwise} \end{cases} \tag{4.40}$$

The precision and recall values are also used to compare the performance of the different data association methods.

4.4 Clustering

The 2d point clouds with object hypotheses, which are obtained by the methods described in Section 4.3.1 and Section 4.3.1, can contain multiple points per object. Clustering groups the detections into different groups. Clustering the detection points could improve the performance of the tracker by reducing the amount of points per object. Also, as mentioned in Section 4.3.2, the point clouds could still contain noise, induced by the multilateration of measurements. Some clustering algorithms are able to distinguish noise from detections, these clustering algorithms are DBSCAN, HDBSCAN and OPTICS. The performance of these clustering algorithms is evaluated on a dataset.

The performance is compared to K-Means clustering as baseline. K-Means clustering is a clustering algorithm without noise filtering, thus K-Means clustering is combined with the distance matrix based noise filter from Section 4.3.2. K-Means clusters data into k groups, minimizing the within-cluster-sum-of-squares. As K-Means requires the amount of clusters in the data as input, this number is found by using the method proposed in [49].

The resulting clusters from each clustering method are compared to a ground truth. This ground truth is obtained from point cloud data of the roof mounted lidar sensor of the Twizy R&D vehicle. For evaluation of the clustering performance, the cumulative Euclidean distance of each cluster centroid P to the nearest ground truth cluster centroid Q is computed, and vice versa

$$d = \sum_{k=1}^{K} \min_{l}^{L} \|Q_l - P_k\| + \sum_{l=1}^{L} \min_{k}^{K} \|Q_l - P_k\| \tag{4.41}$$

with K the total number of clusters and L the total number of ground truth clusters. This metric measures how much the clustering deviates from the ground truth clusters.

The ground truth clusters are obtained by Euclidean cluster extraction of the lidar point cloud data [50]. The clusters are projeced on the (x, y)-plane and all clusters not present in the region-of-interest of the ultrasonic detection system are discarded. All points belonging to the cluster are saved, as well as the centroid of the cluster, which is the average position of all points belonging to the cluster

$$x_c = \frac{\sum_{n=1}^{N} x_n}{N} \quad y_c = \frac{\sum_{n=1}^{N} y_n}{N} \tag{4.42}$$

with N the total number of points belonging to the cluster.

4.5 Tracking

Three different tracking methods are evaluated on the clusters found by the OPTICS clustering algorithm from Section 4.4. The features which are tracked are the centroids of the clusters. The following three tracking algorithms will be evaluated:

- Unscented Kalman filter (UKF) with nearest neighbor track association
- Unscented Kalman filter with joint-probabilistic data association (UKF-JPDA) of detections to tracks
- Gaussian mixture probability hypothesis density (GM-PHD) filter

4.5.1 Unscented Kalman filter (UKF)

The UKF tracks the state

$$\mathbf{x} = \begin{bmatrix} x & \dot{x} & y & \dot{y} \end{bmatrix}^{\mathrm{T}} \tag{4.43}$$

where the initial state is defined as

$$\hat{\mathbf{x}}_{\mathbf{0}} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}} \tag{4.44}$$

The state transition matrix is given by

$$\mathbf{F} = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.45}$$

Since the measurements are the positions of the points in the clusters, or the cluster centroids the measurement function is straightforward

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.46}$$

The measurement noise covariance matrix is given by

$$\mathbf{R} = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.2 \end{bmatrix} \tag{4.47}$$

The process noise is modeled by the discrete white noise model

$$\mathbf{Q} = \begin{bmatrix} \frac{1}{4}\Delta t^4 & \frac{1}{2}\Delta t^3 \\ \frac{1}{2}\Delta t^3 & \Delta t^2 \end{bmatrix} \sigma^2 \tag{4.48}$$

With these definitions the predict and update steps of the UKF [37] can be followed.

4.5.2 Gaussian mixture probability hypothesis density filter (GM-PHD)

The GM-PHD tracks the same state as the UKF

$$\mathbf{x} = \begin{bmatrix} x & \dot{x} & y & \dot{y} \end{bmatrix}^{\mathrm{T}} \tag{4.49}$$

Similarly the same strate transition matrix, measurement function, noise covariance matrix, and process noise are used. The merging and pruning thresholds are set at 5 and 1×10^{-6} respectively.

4.5.3 Evaluation

The tracks generated by the different tracking algorithms are evaluated by computing the optimal sub-pattern assignment (OSPA) metric [51] from the generated tracks and ground truth tracks. The ground truth tracks are collected by the roof mounted lidar sensor of the Twizy R&D vehicle.

Chapter 5

Experiments

In this chapter the experiments and results are presented.

5.1 Sensor calibration

In order to obtain the best detection performance of the sensor, the calibration method described in Section 4.1 was performed.

5.1.1 Datasets

Data collection

To collect data, the measurement setup from Figure 4.3 is setup in a controlled indoor environment. The radial lines of the polar grid from Figure 4.3 are projected on the ground plane with a laser level and marked on the ground with masking tape. The radial distances, with 0.1 m interval, are marked on the masking tape. The ultrasonic sensor is positioned at the origin of the polar grid at a height of 0.35 m, which corresponds to the mounting height of the sensor in the GRT vehicle.

As shown in Figure 5.1, an ultrasonic distance sensor and a BME280 sensor [52] are connected to a Arduino Uno development board [53]. The BME280 sensor measures temperature, relative humidity, and pressure of the air. The Arduino Uno is able to program both sensors and collect their measurement values, which are sent via a USB serial connection to a laptop. The laptop stores the received values in a database. An external power supply provides 12 V to the Arduino and ultrasonic sensor, the BME280 sensor receives its power from the Arduino.

The ISO pole and 2gt object are used as ground truth objects and sequentially positioned at each reference position on the polar grid. For each reference position the measurement sequence is as follows:

1. The ground truth object is manually positioned at the reference position on the grid

2. From the laptop a command to start the measurement sequence is sent to the Arduino development board via the serial connection

3. For each threshold voltage level ℓ of the ultrasonic sensor, the Arduino development board:
   (a) Programs the ultrasonic sensor to use threshold value ℓ
   (b) For a number of N measurement cycles, the Arduino development board:
      i. Instructs the ultrasonic sensor to perform a measurement cycle
      ii. Receives the sensor output
      iii. Calculates the time-of-flight for the detections in the sensor output
   (c) Collects the temperature, relative humidity, and pressure measurements from the BME280 sensor
   (d) Sends the time-of-flight, and ambient air measurements to the laptop
4. The laptop stores the received time-of-flight and ambient air measurements with the corresponding reference position and threshold level in a database

Each row in the resulting database contains the result of one measurement cycle. One row contains the measurements of the ambient air, the number of the measurement cycle n, and up to 10 time-of-flight measurements. The key for each row, i.e. measurement cycle, is the combination of the reference position and measurement cycle number. Table 5.1 shows an example of the data stored in the database.

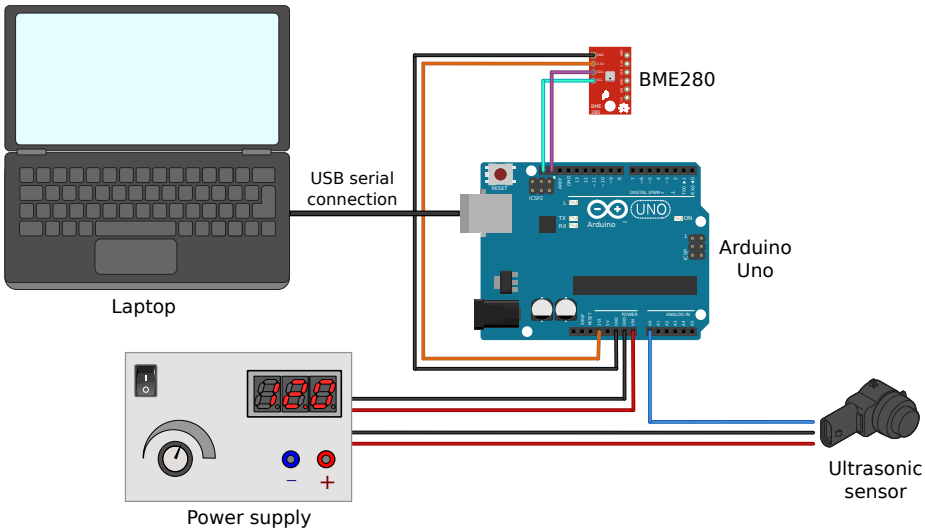


Figure 5.1: Diagram of the measurement setup, which consists of an ultrasonic distance sensor and a BME280 sensor [52] connected to an Arduino Uno development board [53]. The ultrasonic distance sensor measures the distance to the ground truth object, while the BME280 sensor measures the temperature, relative humidity, and pressure during the ultrasonic measurement. The Arduino Uno development board communicates with the sensors and collects all measurements, which are sent to a laptop over a USB serial connection.

Table 5.1: Example of the data stored in the database. Each row contains the reference position and number of the measurement cycle as index, information on the ambient air properties, and the time-of-flight values of the measurement cycle.

Refence position		**Measurement number**	**Ambient air measurement**			**Time-of-flight measurements**			
r	ϕ	n	T	$\%RH$	p	t_1	t_2	...	t_{10}

Data postprocessing

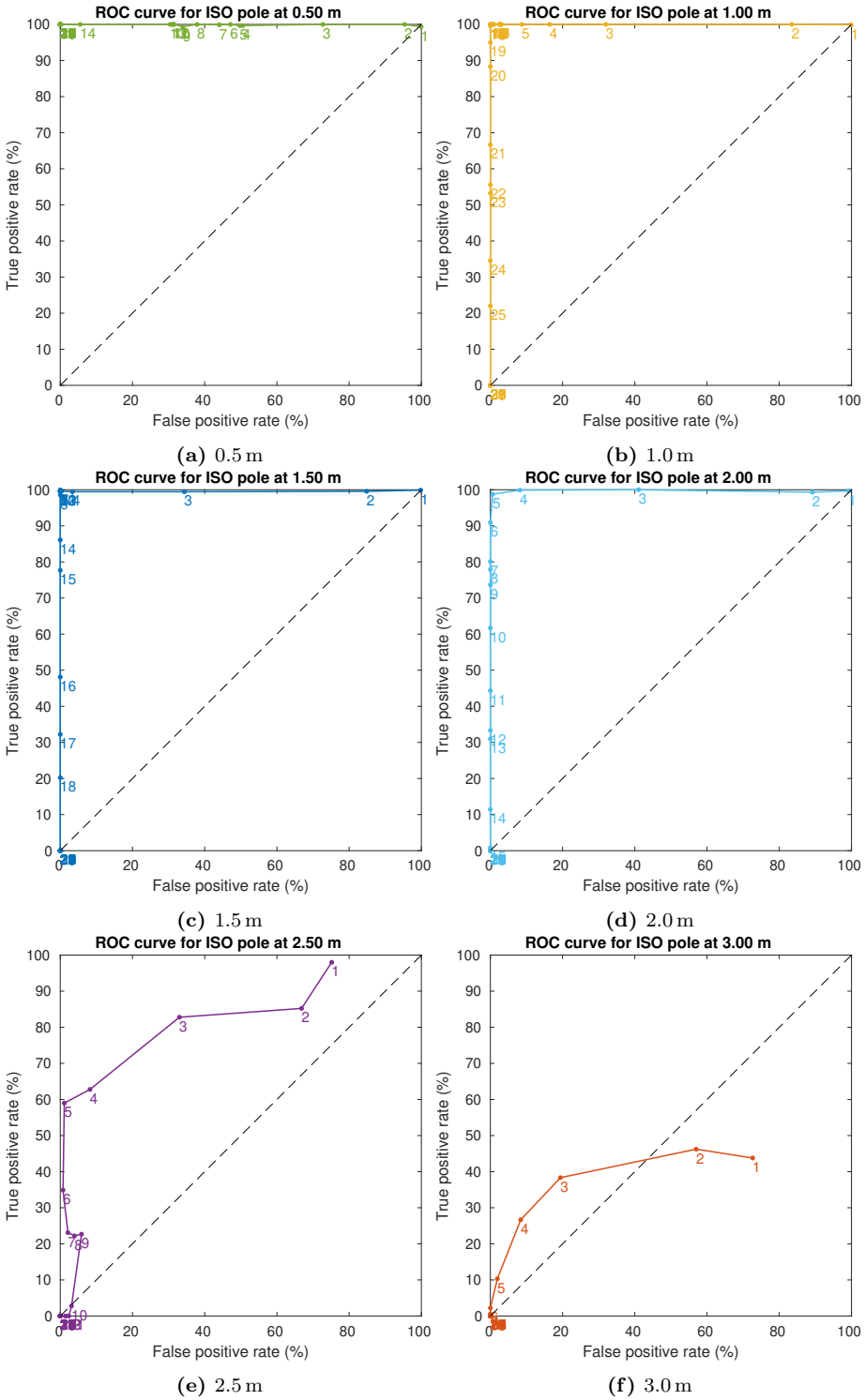
Once the data for each reference position on the polar grid is collected, the measured distances are derived from the time-of-flight measurements, taking into account the temperature, humidity, and pressure at the time of recording. From the definitions proposed in Section 4.1.2, the measurements are either classified as TP, FP, FN or TN, depending on the type of measurement. From these values the ROC curve for each distance r is calculated.

Results

Figure 5.2 shows the resulting ROC curves for the ground truth target positioned at $r_{\text{gt}} = 0.5, 1.0, 1.5, 2.0, 2.5,$ and $3.0\,\text{m}$ respectively. The values next to the ROC curve are the corresponding threshold values of the ultrasonic sensor, defining the threshold voltage level. From this data the optimal threshold value for each distance can be determined, the optimized threshold value being the point closest to the (100% TPR, 0%FPR). Figure 5.3 shows the resulting ROC curves for each reference distance. For each curve the optimized threshold value is indicated. For curves where multiple optimized values are found, the average value is shown.

Based on these values, Figure 5.4 shows the resulting optimal range for the threshold curve, as well as a polynomial fit of the data and the threshold curve as used in the remainder of this document. For the polynomial fit, the fifth degree polynomial was found to have the lowest mean squared error. The proposed threshold curve follows the polynomial fit, but is defined at fixed discrete intervals.

Figure 5.5 and Figure 5.6 show the resulting field of views for both the initial threshold curve and the optimized threshold curve for object types ISO pole and 2gt object, respectively.



(a) 0.5 m **(b)** 1.0 m **(c)** 1.5 m **(d)** 2.0 m **(e)** 2.5 m **(f)** 3.0 m

Figure 5.2: Reciever operating characteristic (ROC) curves for the ISO pole [8] ground truth object at distances $z_{\text{gt}} = \{0.5, 1.0, 1.5, 2.0, 2.5, 3.0\}$ m respectively. The threshold values corresponding to the points on the ROC curve are indicated by their value.

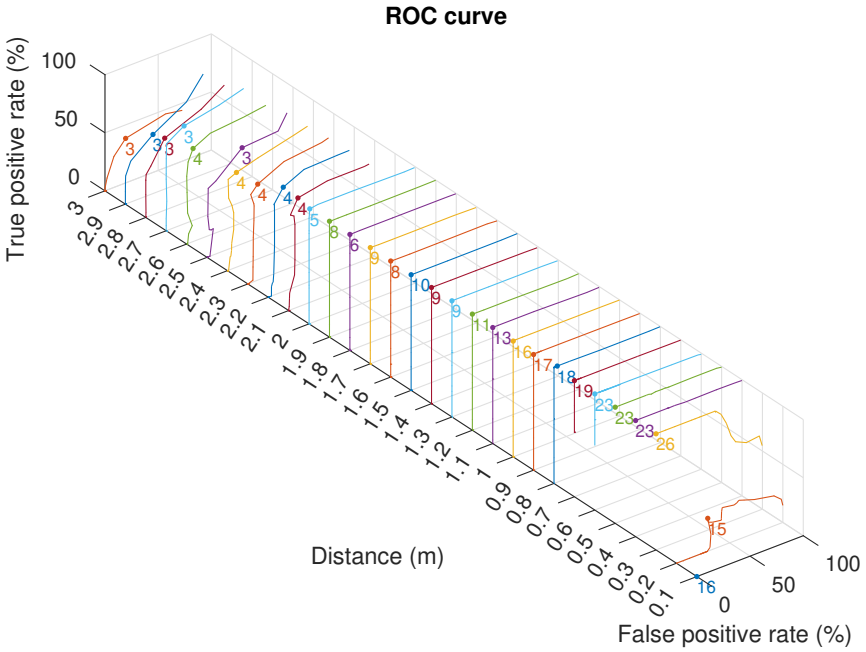


Figure 5.3: Reciever operating characteristic (ROC) curves for the ISO pole [8] ground truth object for each measurement distance. For each curve the point closest to (0 % FPR, 100 % TPR) is indicated, with its corresponding threshold value. Where there are multiple consecutive points closest to (0 % FPR, 100 % TPR), the mean value of the points is shown.

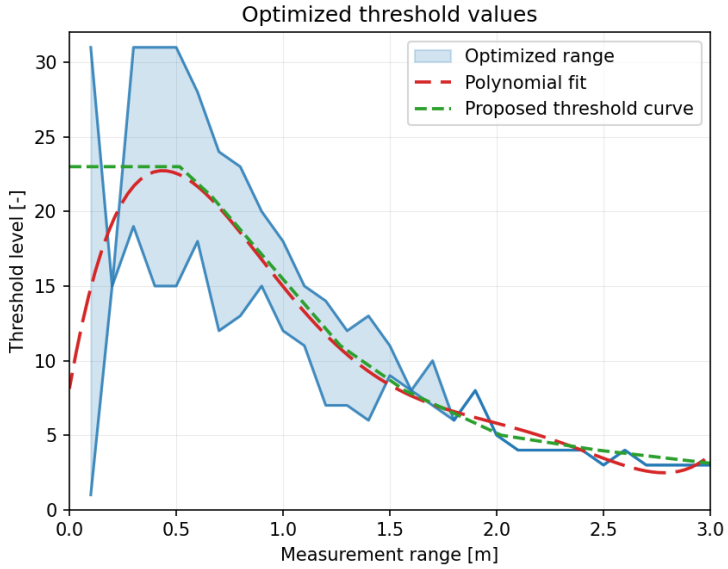
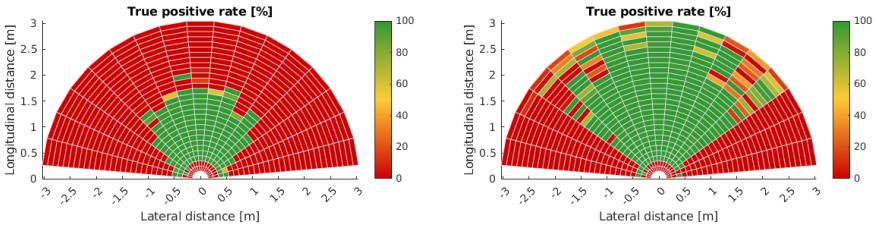
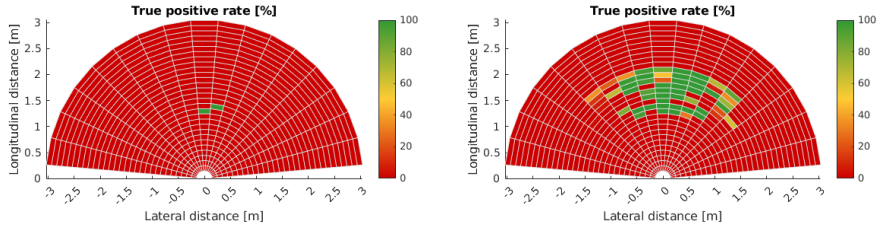


Figure 5.4: Range of optimized threshold values for each measurement range with proposed optimal threshold curve.



(a) Default **(b)** Proposed

Figure 5.5: The true positive rate for detection of the ISO pole object for (a) default threshold curve of the sensor and (b) optimized threshold curve from Figure 5.4.



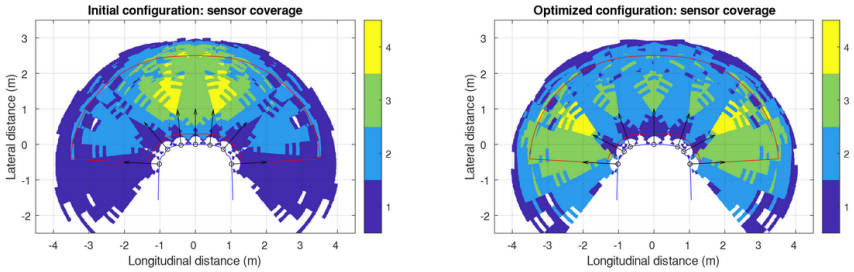
(a) Default **(b)** Proposed

Figure 5.6: The true positive rate for detection of the 2gt object for (a) default threshold curve and (b) optimized threshold curve from Figure 5.4.

5.2 Sensor positioning

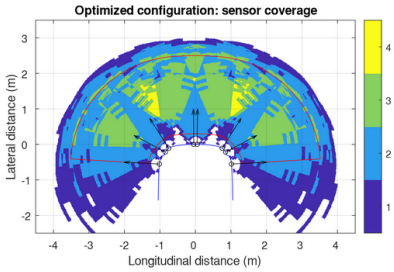
Figure 5.7 and Figure 5.8 show the initial, intermediate steps and final results of the proposed sensor positioning algorithm of a case study on a front bumper of a vehicle. Figure 5.7 shows the k-cover of the ROI, while Figure 5.8 shows the blind spot areas if multilateration were to be used with this ultrasonic detection system.

Results of the optimization are shown in Table 5.2.

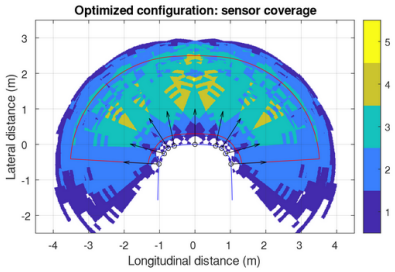


(a) Initial configuration

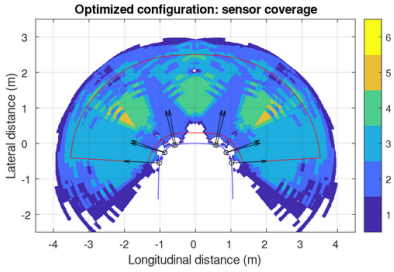
(b) Optimized 7 sensors



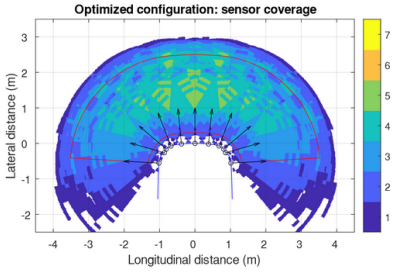
(c) Optimized 8 sensors



(d) Optimized 9 sensors

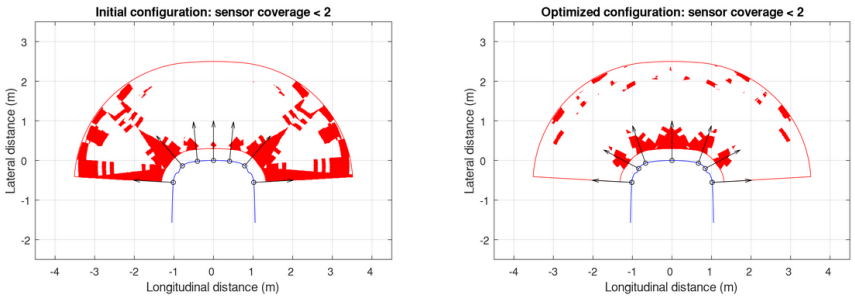


(e) Optimized 10 sensors



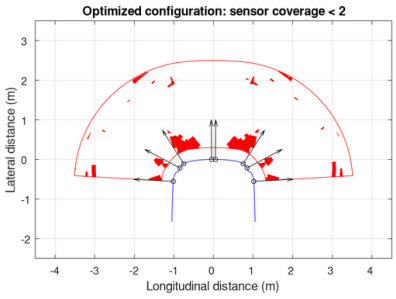
(f) Optimized 11 sensors

Figure 5.7: Sensor coverage for (a) the initial configuration, and optimized configuration for (b) seven to (f) eleven sensors

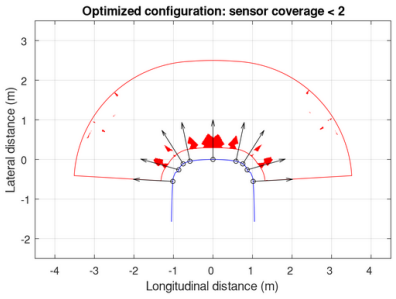


(a) Initial configuration

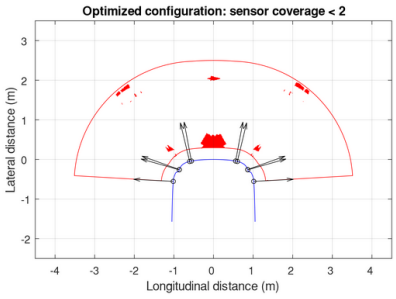
(b) Optimized 7 sensors



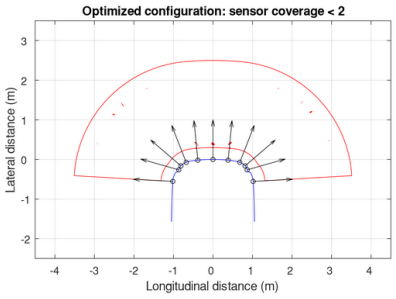
(c) Optimized 8 sensors



(d) Optimized 9 sensors



(e) Optimized 10 sensors



(f) Optimized 11 sensors

Figure 5.8: Areas where k-Cover < 2 for (a) the initial configuration, and optimized configuration for (b) seven to (f) eleven sensors. Since multilateration requires at least two range measurements from different sensors, areas which are covered with less than two sensors are considered blind spots of the system.

Table 5.2: Results of the optimization process.

	Initial configuration	**Optimized 7 sensors**	**Optimized 8 sensors**
k-cover ≥ 1 sensor (%)	99.6	100.0	99.6
k-cover ≥ 2 sensor (%)	74.8	88.3	94.7
k-cover ≥ 3 sensor (%)	32.0	41.6	42.2
Mean absolute error after multilateration $\times 10^{-1}$[m]	0.83	1.17	1.36
CRLB $\times 10^{-3}$	1.30	2.09	1.91
	Optimized 9 sensors	**Optimized 10 sensors**	**Optimized 11 sensors**
k-cover ≥ 1 sensor (%)	100.0	99.2	100.0
k-cover ≥ 2 sensor (%)	97.5	98.0	99.9
k-cover ≥ 3 sensor (%)	58.4	67.5	85.0
Mean absolute error after multilateration $\times 10^{-1}$[m]	1.13	1.68	1.01
CRLB $\times 10^{-3}$	2.23	2.02	1.89

5.3 Detection algorithm

5.3.1 Multilateration using direct echo

In Section 4.3.1 two methods for obtaining an estimated object position using direct echo measurements are presented: a nonlinear least squares solution and an exact solution. In this section these two methods are compared using simulated data.

Given two ultrasonic sensors s_1 and s_2 positioned at $(0, -0.2)$ m and $(0, 0.2)$ m, as shown in Figure 5.9. The effective region-of-interest of the system consist of the intersection of the regions-of-interest of the respective sensors. For each point p_i in the effective region-of-interest, distances r_1 and r_2 to s_1 and s_2 are calculated, respectively. With Gaussian noise added to distances r_1 and r_2, the position of p_i is estimated using both the nonlinear least squares solution and the exact solution. The difference between the linear least squares estimate and the exact estimate are compared, as well as the runtime of both methods. Table 5.3 shows that while the position estimate of p_i results in similar errors, the the average runtime of the nonlinear least squares method is higher than the exact solution.

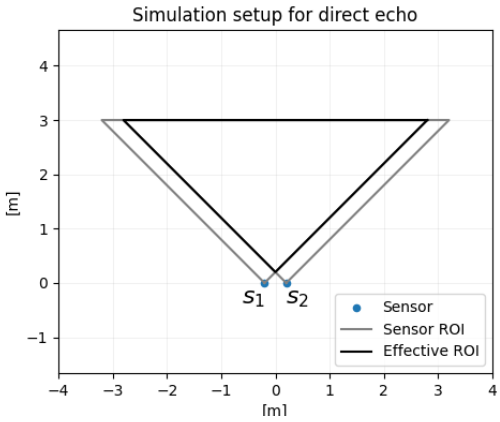


Figure 5.9: A system consisting of two ultrasonic sensors s_1 and s_2, with an effective region-of-interest as defined by the intersection of the regions-of-interest of the respective sensors.

Table 5.3: Results for multilateration methods from Section 4.3.1 on a scenario with three sensors and additional white noise

Method	**Average error** $\times 10^{-2}$ **[m]**	**Maximum error [m]**	**Average runtime [ms]**
Non-linear least squares	7.01 ± 6.01	0.55	1.01 ± 0.31
Circle	7.13 ± 6.01	0.55	0.04 ± 0.01

5.3.2 Multilateration using cross echo

In Section 4.3.1 three methods for obtaining an estimated object position using cross echo measurements are presented: a nonlinear least squares solution, an exact ellipse solution, and an exact circle solution. The exact circle solution is a simplified version of the exact ellipse solution.

The simulation of Section 5.3.1 is extended with an additional sensor, as shown in Figure 5.10. Three ultrasonic sensors s_1, s_2, and s_3 are positioned at $(-0.2, 0)$, $(0, 0)$, and $(0.2, 0)$ m, respectively. For each point p_i in the effective region-of-interest of the system, distances r_1, r_2, and r_3 from p_i to sensors s_1, s_2, and s_3 are calculated.

With Gaussian noise added to distance measurements r_1, r_2, and r_3, the position of p_i is estimated using the three different multilateration methods. Again, the positional error is defined as the Euclidean distance between the estimated position of p_i and the actual position of p_i. Table 5.4 shows the positional error of the three different methods, as well as the average runtime.

Figure 5.10 shows the positional error of simulation of the exact circle solution without Gaussian noise added to the distance measurements. This shows that making the assumption of using circles instead of ellipses introduces a maximum positional error of 0.04 m for this scenario, while reducing the average runtime when compared to the other methods.

Table 5.4: Results for multilateration methods from Section 4.3.1 on a scenario with three sensors and additional white noise

Method	Average error $\times 10^{-2}$ [m]	Maximum error [m]	Average runtime [ms]
Nonlinear least squares	3.61 ± 3.07	0.28	1.09 ± 0.34
Ellipse	3.63 ± 3.06	0.28	0.08 ± 0.01
Circle	3.94 ± 2.94	0.27	0.04 ± 0.01

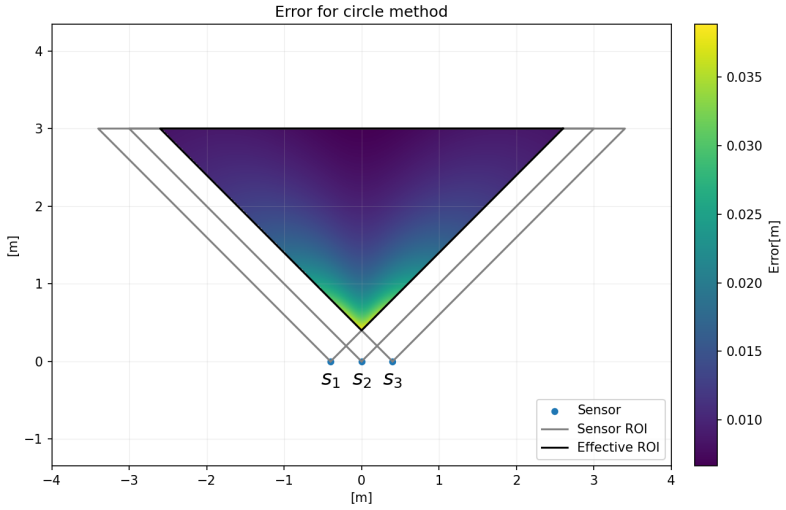


Figure 5.10: A simulation of a system consisting of three ultrasonic sensors s_1, s_2, and s_3, with an effective region-of-interest as defined by the intersection of the regions-of-interest of the respective sensors. The effective region-of-interest shows the error for the target position as estimated by the circle method. The error is defined as the Euclidean distance between the estimated position and the actual position. The maximum error inside the effective region-of-interest is 0.04 m.

5.3.3 Data association

This section describes the optimization of several parameters for the proposed noise filtering methods.

5.3.4 Dataset

In order to improve the performance of the system, this optimization is done using real-world data. This data is recorded with the Twizy R&D vehicle, Figure 5.11 shows the vehicle. The Twizy R&D vehicle is a small manually driven vehicle, equipped with various state-of-the-art sensors for automated and autonomous driving, which include:

- One roof mounted lidar sensor

- One front facing lidar sensor
- Three front facing radar sensor
- Front facing camera
- GNSS and IMU sensor
- 16 ultrasonic range sensors

The data from all sensors is recorded and stored locally, after which it can be offloaded for further analysis. The recorded datasets consist of the data from these sensors, and is accompanied by a description of the recorded scenario.

Since detection and tracking of pedestrians is a key element of the intended use of the system, a collection of datasets `pedestrian_pattern` was recorded where a pedestrian moves inside the ROI of the ultrasonic detection system of the Twizy. The pedestrian moves back and forth along a predefined trajectory, as shown in Figure 5.12.

Ground truths are extracted from the data of the roof mounted lidar. First the ground plane in the lidar data is detected, any points not belonging to the ground plane are extracted and clustered. If the clusters meet predefined thresholds on their size, the clusters are saved as ground truth. The saved ground truth data consist of the number of clusters, their points, bounding boxes and centroids.

The performance of the data association methods will be done on the `pedestrian_pattern_05` dataset from the `pedestrian_pattern` dataset collection.

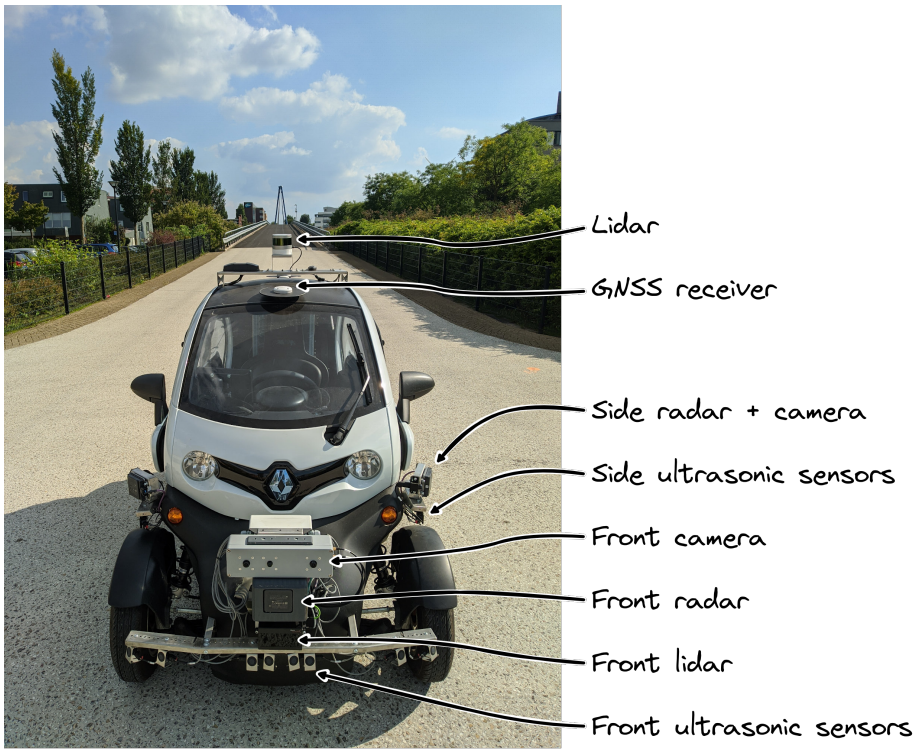


Figure 5.11: Front view of the Twizy R&D vehicle. The Twizy R&D vehicle is a manually driven vehicle equipped with state-of-the-art automotive sensors, which allows it to record datasets of its environment. The sensor set of the Twizy R&D vehicle consist of a roof-mounted lidar, GNSS receiver, front and side facing camera, front and side facing radar, front facing lidar and front and side facing ultrasonic sensors.

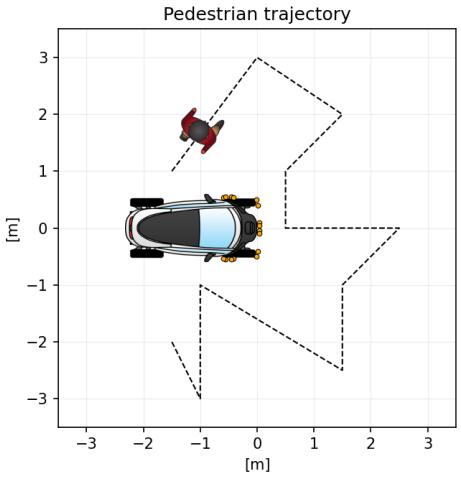


Figure 5.12: A top-down overview of the `pedestrian_pattern_05` dataset. The dataset consists of a parked ego-vehicle with a pedestrian walking back and forth along a predefined track. The pattern was created by connecting a set of randomly picked points on a grid with 0.5 m resolution.

5.3.5 Parameter optimization

NLLS solution: $SSE_{\max}$ and r optimization

Optimization of the nonlinear least squares solution, yields the results shown in Figure 5.13. This figure shows the precision-recall curve of the NLLS for various values of r and SSE_max.

As shown in the figure, increasing the value of r has a positive effect on precision and recall, but only up to a certain point, after which both precision and recall decrease with increasing r. This is probably a result of adding too much unrelated measurements into one set of NNLS equations. Varying the level of $SSE_{\max}$ has far little effect on precision, when compared to r.

The variant with The highest F_1-score of 0.433 is the NLLS with $SSE_{\max} = 0.1$ and $r = 0.04$.

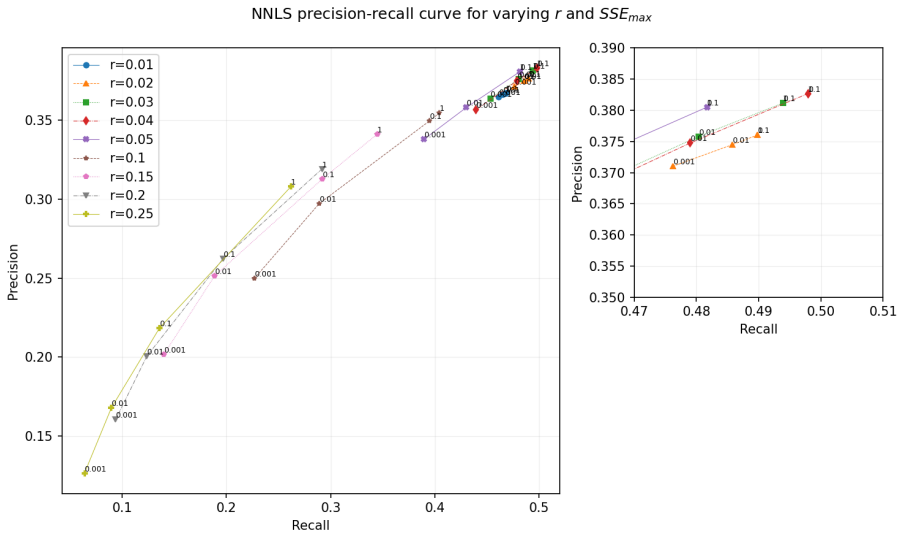


Figure 5.13: Precision-recall plot of NLLS solution with for various values of r with increasing $SSE_{\max}$.

Exact ellipse solution with distance matrix: n and r optimization

The precision-recall curves for the optimization of n and r for the exact ellipse solution, with filtering using a distance matrix is shown in Figure 5.14. Here an increase of n has a decremental effect on recall, while an increase of r has a decremental effect on precision.

The highest F_1-score of 0.493 is achieved with $n = 2$ and $r = 0.3$.

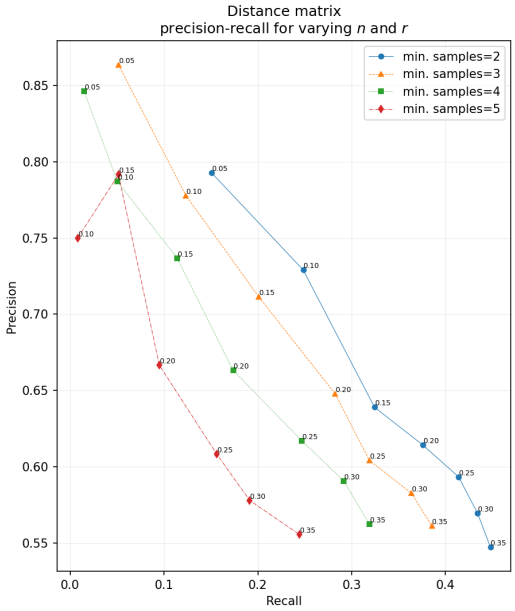


Figure 5.14: Precision recall plot for distance matrix solution. For each value of n neighboring points an increase of radius r is shown.

Exact ellipse solution with DBSCAN: minimum sample size and ϵ optimization

Figure 5.15 shows the precision-recall curve of the exact ellipse solution with DBSCAN clustering for different sample sizes and increasing ϵ values. An increase of ϵ results in an increase in recall, but a decrease in precision.

The highest F_1-score of 0.5284 is achieved with a minimum sample size of 2 and $\epsilon = 0.45$.

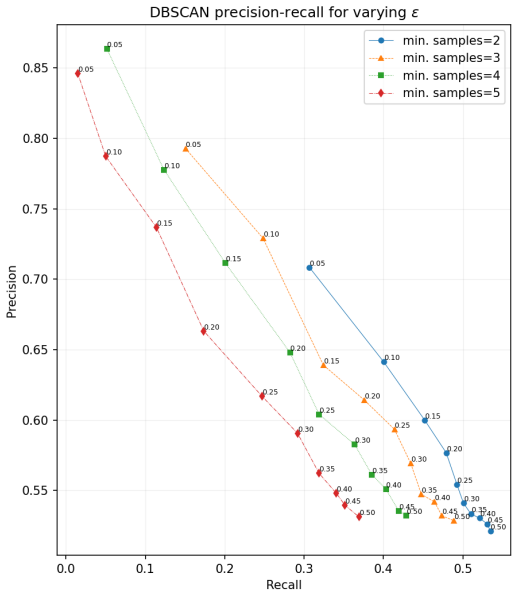


Figure 5.15: Precision recall plot for DBSCAN. For each minimum sample size an increase of ϵ is shown.

Exact ellipse solution with HDBSCAN: minimum cluster size optimization

Figure 5.16 shows the precision-recall curve of the exact ellipse solution with HDBSCAN clustering for various minimum cluster size values. Increasing the minimum cluster size has a decremental effect on recall.

The highest F_1-score of 0.336 is achieved with a minimum cluster size value of 2.

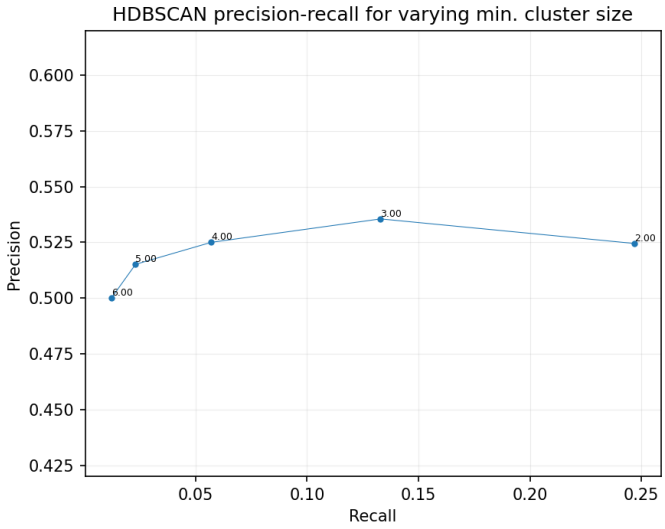


Figure 5.16

Exact ellipse solution with OPTICS: minimum sample size, ξ, and minimum cluster fraction optimization

Figure 5.17 shows the precision-recall curve of the exact ellipse solution with OPTICS clustering. Each line plot gives the precision and recall for increasing minimum cluster fraction for different values of minimum sample size and ξ. Increasing the minimum sample size has a decremental effect of recall. Increasing ξ does not seem to influence precision nor recall. Increasing the minimum cluster fraction has a positive effect on precision.

The highest F_1-score of 0.556 is achieved with a minimum sample size of 2, $\xi = 0.25$, and a minimum cluster fraction of 0.35.

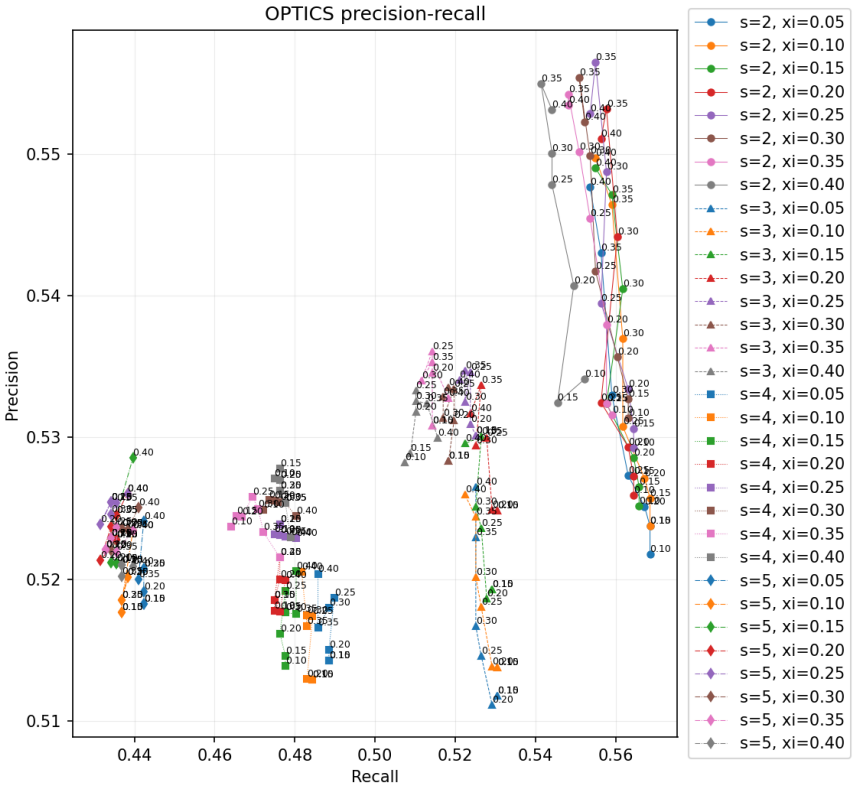


Figure 5.17: Precision-recall plot for OPTICS method. For each combination of sample size s and ξ an increasing minimum cluster fraction is shown.

5.3.6 Results

A comparison of the variant with the highest F_1-score of each method is shown in Figure 5.18. Except for the NNLS solution, all other methods use the detections from the exact ellipse solution as input. The clustering methods most robust to the noise of the ultrasonic detections are DBSCAN and OPTICS. OPTICS achieves the highest F_1-score, and has the second highest value for precision and the highest value for recall.

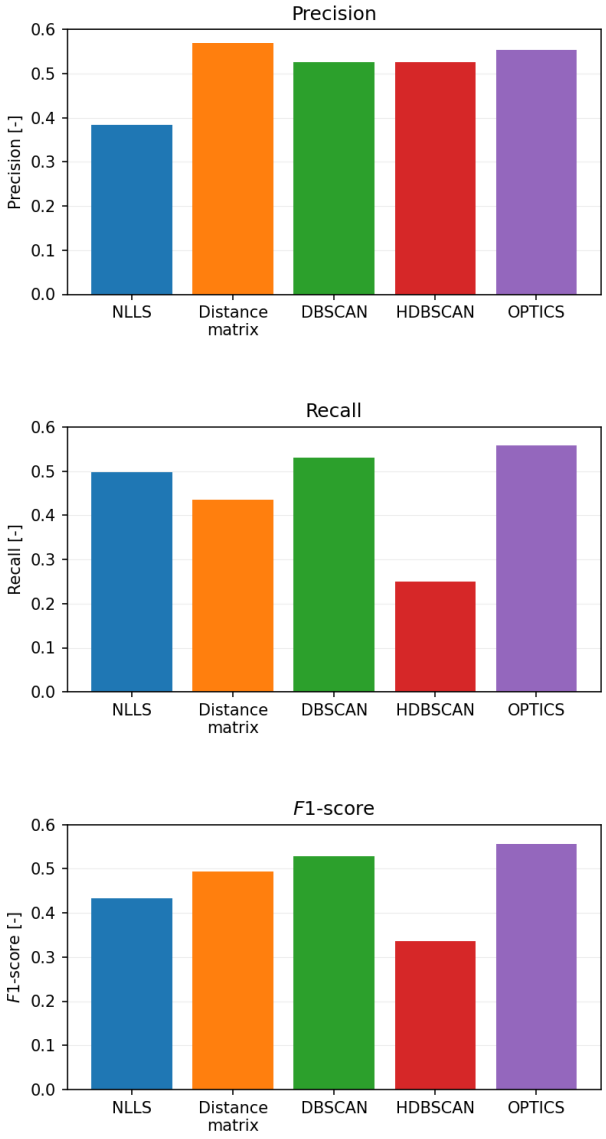


Figure 5.18: Comparison of the precision, recall, and F_1-score of the best performing variant of each noise filtering method.

5.4 Clustering

This section further evaluates the performance of the clustering algorithms from the previous section. The clustering algorithms evaluated are K-Means clustering with distance matrix noise removal, DBSCAN, HDBSCAN, and OPTICS. All clustering algorithms use the detections from the exact ellipse solution as input data. The clustering algorithms in this section will use their respective optimized parameters as found in the previous section.

5.4.1 Dataset

The dataset used to evaluate the clustering algorithms is the `pedestrian_pattern_05` dataset, as described in Section 5.3.4.

5.4.2 Results

The clusters found by the respective clustering methods are compared to ground truth clusters extracted from the data of the roof-mounted lidar sensor. The cumulative error from Equation (3.34) is used for evaluation of the different clustering methods. Figure 5.19 shows the results.

DBSCAN and OPTICS have the lowest cumulative error, indicating the clusters produced by these method most closely represent the ground truth clusters.

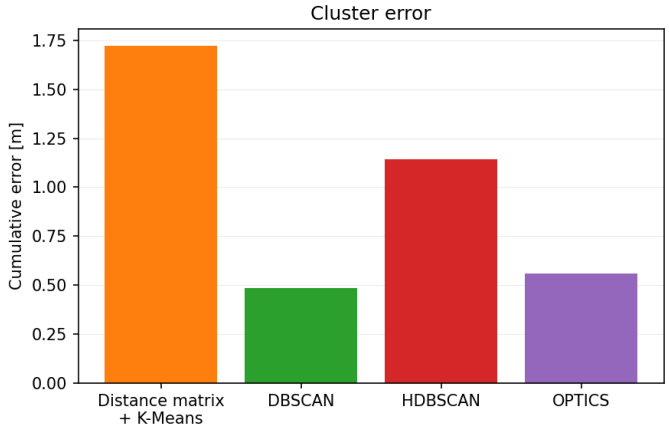


Figure 5.19: Cumulative error of the clustering algorithms compared with the ground truth. A lower error indicates that the clusters have a higher similarity with the ground truth clusters.

5.5 Tracking

In this section the results of the UKF, UKF-JPDA and GM-PHD tracking methods are evaluated and compared. The tracking methods are evaluated on the `pedestrian_pattern` dataset collection, `pedestrian_crossing` dataset collection, `iso_pole_stop_5kmh` dataset collection, and `iso_pole_driveby_5kmh` dataset collection.

5.5.1 Datasets

The tracking methods are evaluated on 4 different types of datasets.

pedestrian_pattern

The `pedestrian_pattern` dataset collection contains 10 recordings. As described in Section 5.3.4, the `pedestrian_pattern` contains recordings of a pedestrian walking back and forth along a predefined track inside the ROI of the ultrasonic system.

pedestrian_crossing

The `pedestrian_crossing` dataset collection contains 1 recording. In this recording the ego-vehicle is stopped at a station platform, a pedestrian walks back and forth 10 times over a crosswalk in front of the vehicle. The direction of the pedestrian is perpendicular to the heading of the vehicle. The distance between the vehicle and pedestrian is approximately 1.5 m.

iso_pole_stop_5kmh

The `iso_pole_stop_5kmh` dataset collection contains 10 recordings. In these recordings the ego-vehicle approaches a target of the type "ISO pole" at $5\,\mathrm{km\,h^{-1}}$ and performs an emergency stop. The "ISO pole" target is positioned in the center of the trajectory of the ego-vehicle.

iso_pole_driveby_5kmh

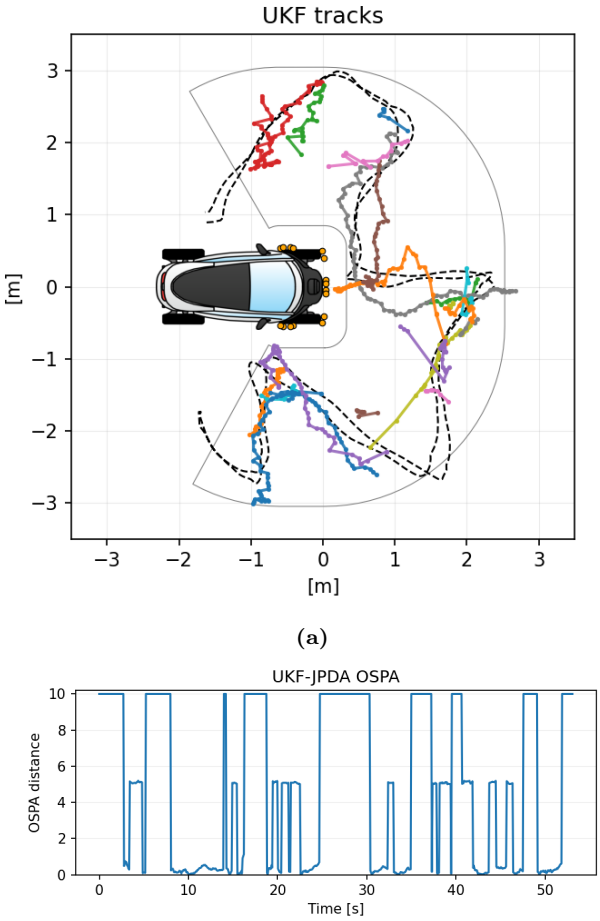
The `iso_pole_driveby_5kmh` dataset collection contains 10 recordings. In these recordings the ego-vehicle approaches a target of the type "ISO pole" at $5\,\mathrm{km\,h^{-1}}$ and drives past the target at a constant velocity. The "ISO pole" target is positioned outside of the trajectory of the ego-vehicle, with a distance of approximately 0.5 m between the target and ego-vehicle.

5.5.2 Results

The results of the tracking are the tracks themselves and the OSPA metric.

pedestrian_pattern

UKF tracker Figure 5.20 shows the tracks from the UKF tracker and the OSPA metric over time.

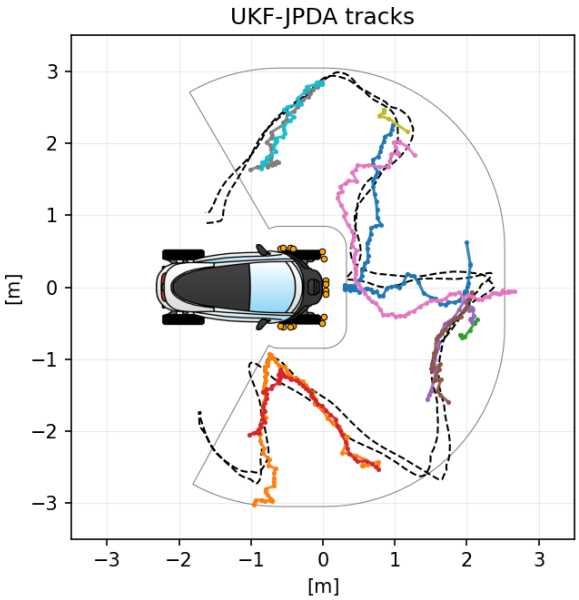


(a)

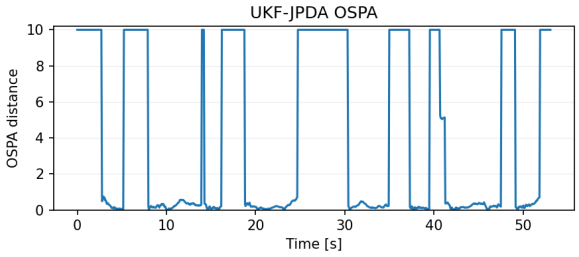
(b)

Figure 5.20: The (a) tracks and (b) OSPA metric of the UKF tracker for the `pedestrian_pattern_05` dataset.

UKF-JPDA tracker Figure 5.21 shows the tracks from the UKF-JPDA tracker and the OSPA metric over time.



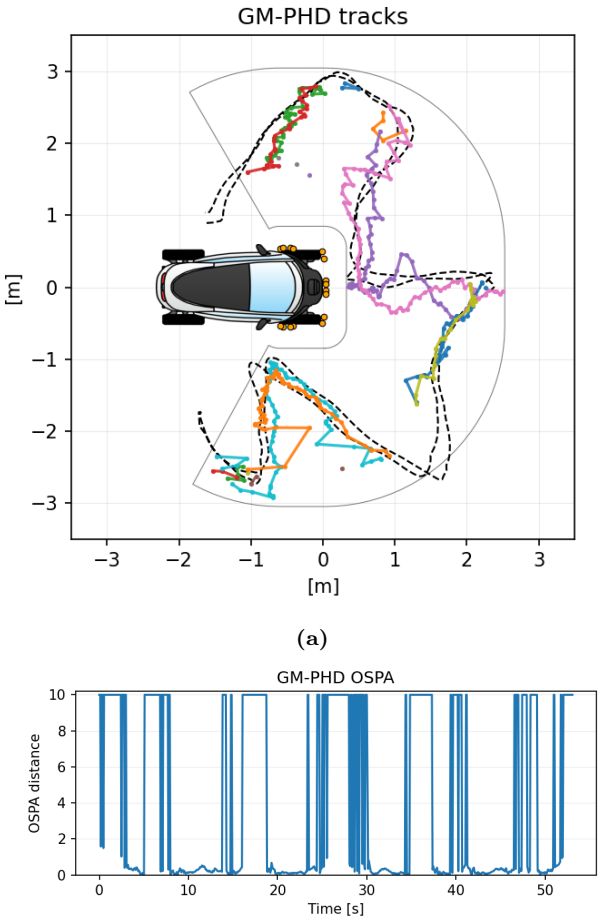
(a)



(b)

Figure 5.21: The (a) tracks and (b) OSPA metric of the UKF-JPDA tracker for the `pedestrian_pattern_05` dataset.

GM-PHD tracker Figure 5.22 shows the tracks from the GM-PHD tracker and the OSPA metric over time.



(a)

(b)

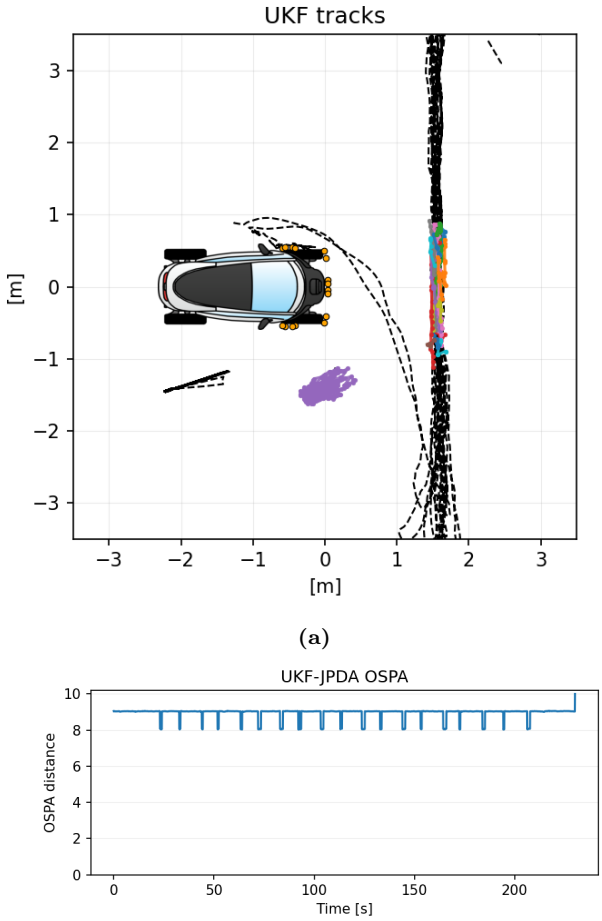
Figure 5.22: The (a) tracks and (b) OSPA metric of the GM-PHD tracker for the `pedestrian_pattern_05` dataset.

Table 5.5: Resulting OSPA values for each tracker for the `pedestrian_pattern` dataset collection. The tracked features are the cluster centroids.

Recording	UKF	UKF-JPDA	GM-PHD
`pedestrian_pattern_01`	4.87	4.02	5.05
`pedestrian_pattern_02`	6.26	6.82	7.19
`pedestrian_pattern_03`	4.32	3.42	4.55
`pedestrian_pattern_04`	6.01	6.85	7.28
`pedestrian_pattern_05`	4.88	4.03	4.51
Average	5.27	5.08	5.72

pedestrian_crossing

UKF tracker Figure 5.23 shows the tracks from the UKF tracker and the OSPA metric over time.

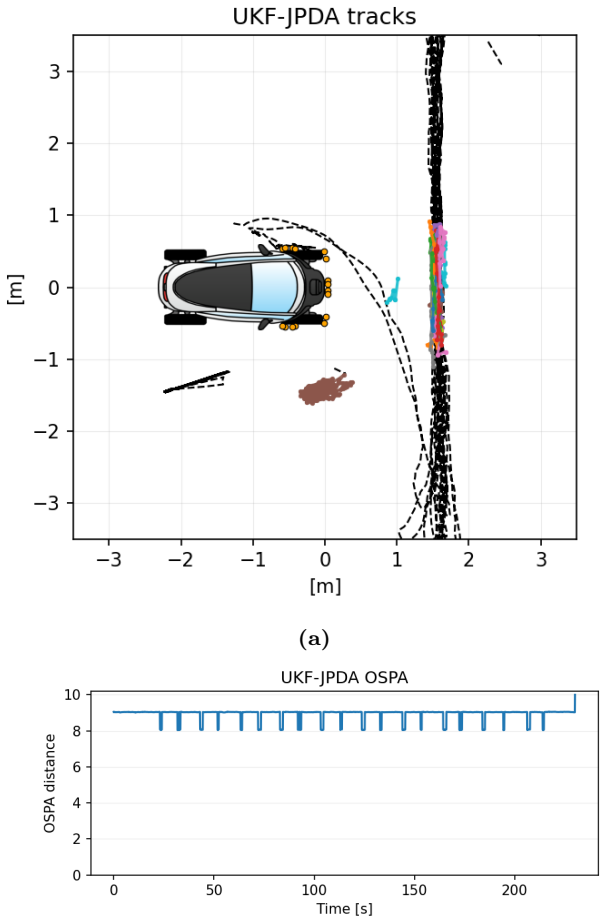


(a)

(b)

Figure 5.23: The (a) tracks and (b) OSPA metric of the UKF tracker for the `pedestrian_crossing` dataset.

UKF-JPDA tracker Figure 5.24 shows the tracks from the UKF-JPDA tracker and the OSPA metric over time.

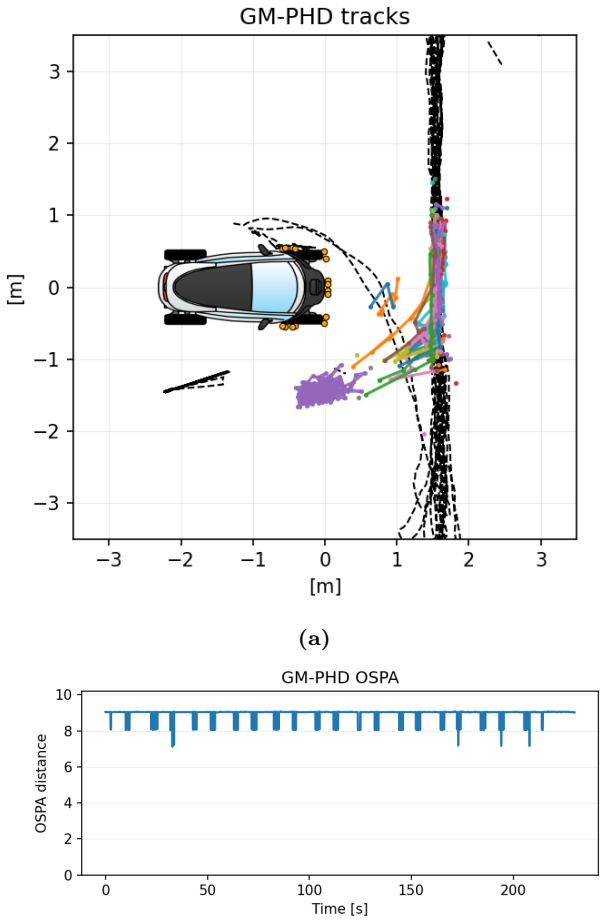


(a)

(b)

Figure 5.24: The (a) tracks and (b) OSPA metric of the UKF-JPDA tracker for the `pedestrian_crossing` dataset.

GM-PHD tracker Figure 5.25 shows the tracks from the GM-PHD tracker and the OSPA metric over time.



(a)

(b)

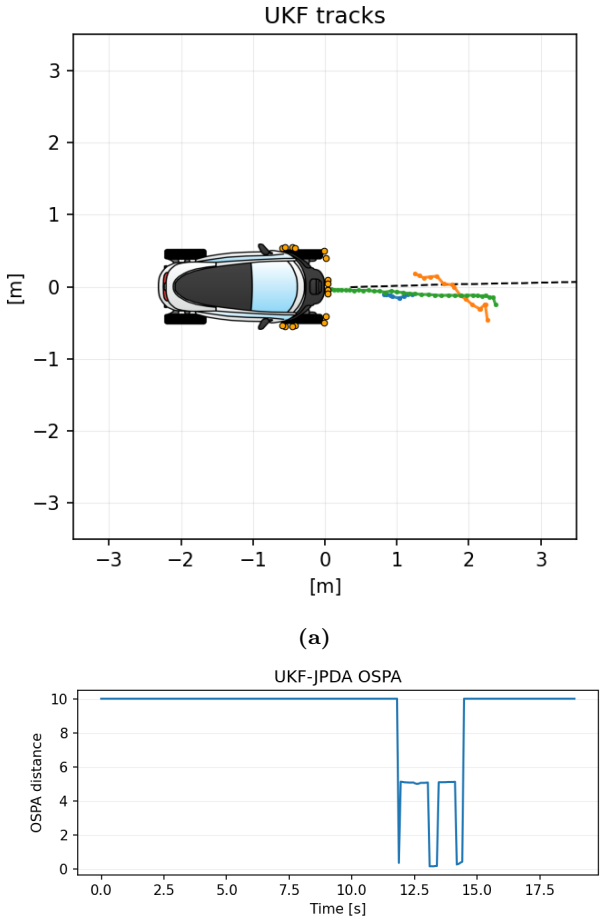
Figure 5.25: The (a) tracks and (b) OSPA metric of GM-PHD tracker for the `pedestrian_crossing` dataset.

Table 5.6: Resulting average OSPA values for each tracker for the `pedestrian_pattern` dataset collection.

Recording	UKF	UKF-JPDA	GM-PHD
`pedestrian_crossing`	8.96	8.94	8.94

iso_pole_stop

UKF tracker Figure 5.26 shows the tracks from the UKF tracker and the OSPA metric over time.

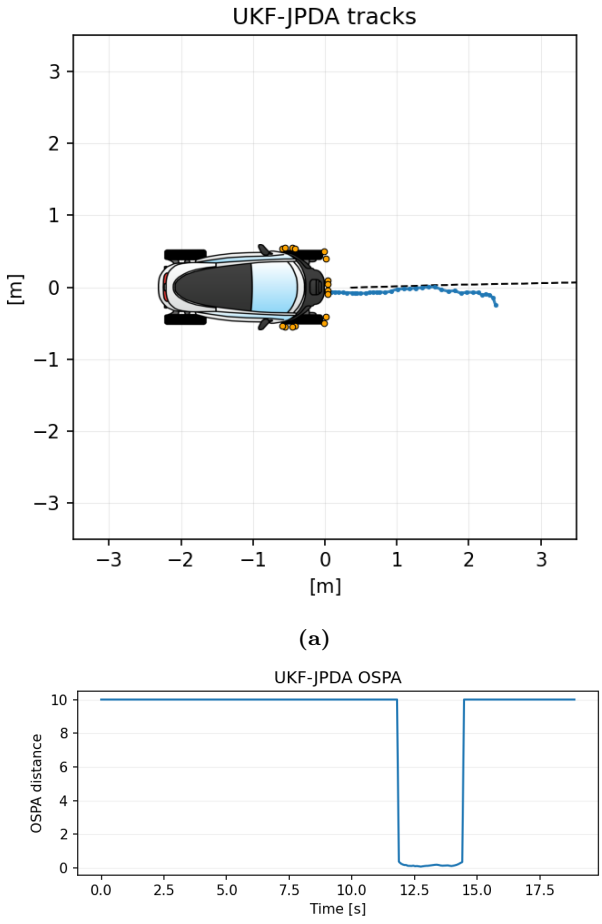


(a)

(b)

Figure 5.26: The (a) tracks and (b) OSPA metric of the UKF tracker for the `iso_pole_stop_05kmh_01` dataset.

UKF-JPDA tracker Figure 5.27 shows the tracks from the UKF-JPDA tracker and the OSPA metric over time.

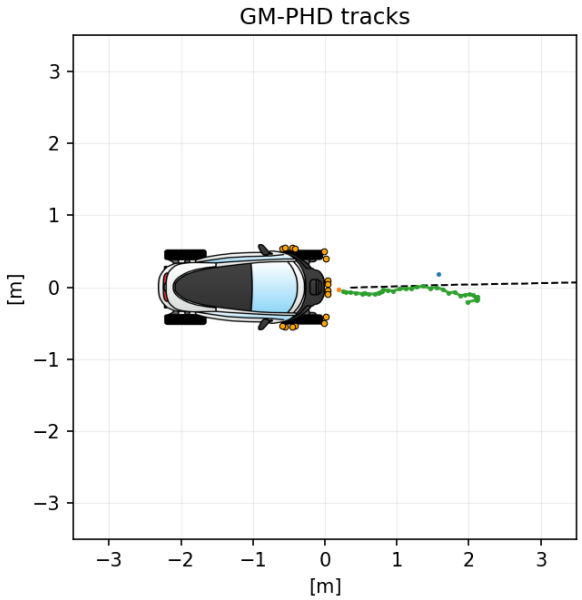


(a)

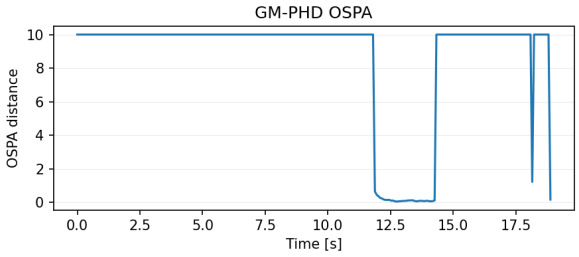
(b)

Figure 5.27: The (a) tracks and (b) OSPA metric of the UKF-JPDA tracker for the `iso_pole_stop_05kmh_01` dataset.

GM-PHD tracker Figure 5.28 shows the tracks from the GM-PHD tracker and the OSPA metric over time.



(a)



(b)

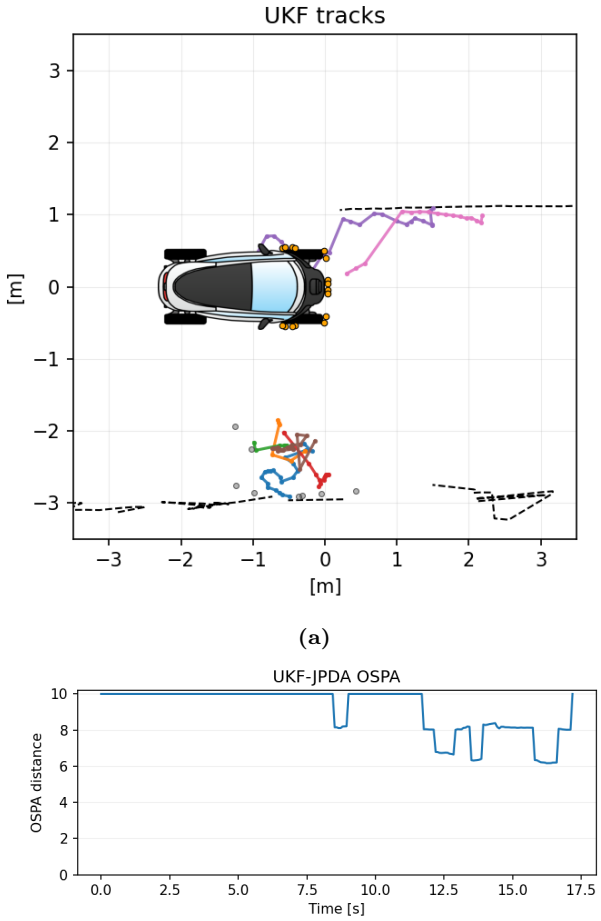
Figure 5.28: The (a) tracks and (b) OSPA metric of the GM-PHD tracker for the `iso_pole_stop_05kmh_01` dataset.

Table 5.7: Resulting OSPA values for each tracker for the `iso_pole_stop` dataset collection.

Recording	UKF	UKF-JPDA	GM-PHD
`iso_pole_stop_05kmh_01`	9.14	8.65	8.66
`iso_pole_stop_05kmh_02`	6.15	5.57	5.58
`iso_pole_stop_05kmh_03`	8.48	7.79	7.50
`iso_pole_stop_05kmh_04`	7.22	6.79	6.96
`iso_pole_stop_05kmh_05`	8.22	7.69	7.77
Average	7.84	7.30	7.29
`iso_pole_stop_10kmh_01`	5.42	5.42	5.22
`iso_pole_stop_10kmh_02`	8.61	8.61	8.48
`iso_pole_stop_10kmh_03`	8.79	9.04	9.11
`iso_pole_stop_10kmh_04`	7.08	6.40	6.52
`iso_pole_stop_10kmh_05`	6.40	6.01	5.96
Average	7.26	7.08	7.06

iso_pole_driveby

UKF tracker Figure 5.29 shows the tracks from the UKF tracker and the OSPA metric over time.

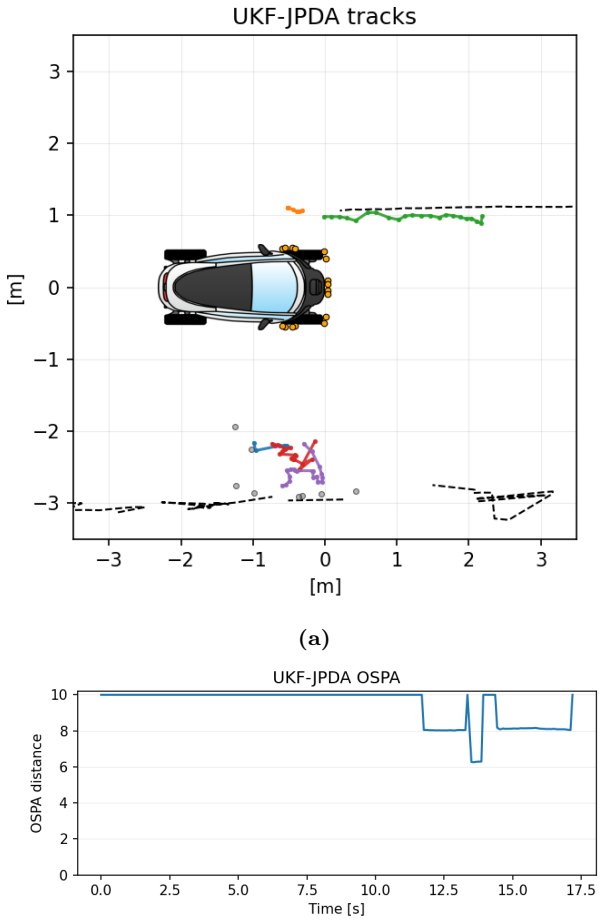


(a)

(b)

Figure 5.29: The (a) tracks and (b) OSPA metric of the UKF tracker for the `iso_pole_driveby_05kmh_01` dataset.

UKF-JPDA tracker Figure 5.30 shows the tracks from the UKF-JPDA tracker and the OSPA metric over time.

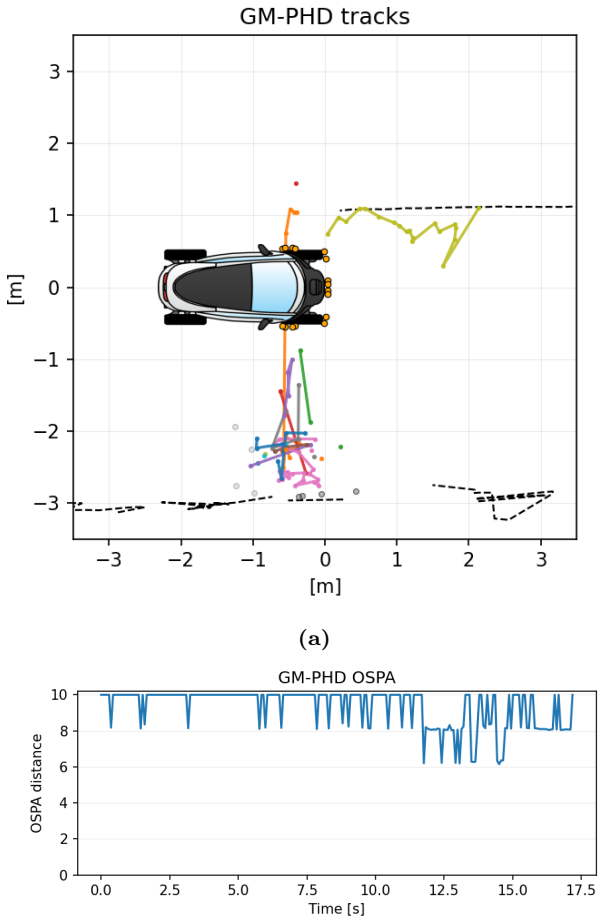


(a)

(b)

Figure 5.30: The (a) tracks and (b) OSPA metric of the UKF-JPDA tracker for the `iso_pole_driveby_05kmh_01` dataset.

GM-PHD tracker Figure 5.31 shows the tracks from the GM-PHD tracker and the OSPA metric over time.



(a)

(b)

Figure 5.31: The (a) tracks and (b) OSPA metric of the GM-PHD tracker for the `iso_pole_driveby_05kmh_01` dataset.

Table 5.8: Resulting OSPA values for each tracker for the `iso_pole_driveby` dataset collection.

Recording	UKF	UKF-JPDA	GM-PHD
`iso_pole_driveby_05kmh_01`	9.16	9.42	9.35
`iso_pole_driveby_05kmh_02`	7.84	8.20	8.67
`iso_pole_driveby_05kmh_03`	8.11	8.59	8.75
`iso_pole_driveby_05kmh_04`	7.55	8.00	8.02
`iso_pole_driveby_05kmh_05`	8.30	8.48	8.53
Average	8.19	8.54	8.66
`iso_pole_driveby_10kmh_01`	7.70	7.70	8.01
`iso_pole_driveby_10kmh_02`	6.59	6.79	6.73
`iso_pole_driveby_10kmh_03`	5.52	6.39	6.22
`iso_pole_driveby_10kmh_04`	4.64	4.49	4.52
`iso_pole_driveby_10kmh_05`	6.25	6.80	6.63
Average	6.14	6.43	6.42

Chapter 6

Conclusion and future work

This thesis covered the use of ultrasonic distance sensors for object detection and tracking in autonomous vehicles. Specifically, the use of ultrasonic distance sensors for detection and tracking of vulnerable road users for an autonomous vehicle intended for public transport. Due to the specific operational design domain of the vehicle, there is a high probability that vulnerable road users might come in close proximity to the vehicle.

Several aspects of ultrasonic sensors and several methods enabling the use ultrasonic distance sensors in autonomous vehicles for object detection and tracking were covered in the different chapters of this thesis. These topics included optimization of sensor detection performance, optimization of sensor positioning on the vehicle, evaluation of object positioning methods, and evaluation of object tracking methods.

Previous studies showed that the default detection range of automotive ultrasonic detection systems is limited and might not suffice in the detection of pedestrians [18]. This study showed that optimizing the time dependent threshold curve of the sensors using ground truth objects improved the detection performance of the sensor when compared to the default threshold curve of the sensor. This enabled better detection performance of objects within the region-of-interest of the system, up to the maximum detection range of the sensor, enabling the system to detect and track pedestrians inside the region-of-interest of the system.

In a case study, the positions of the ultrasonic sensors in the front bumper of a vehicle where optimized. It was shown that the proposed algorithm improved the theoretical performance of the sensor system when compared to the initial configuration. Using a cost function consisting of multiple parameters ensured that blind spots in the region-of-interest of the system were minimized, while maintaining detection accuracy.

The detection and tracking part of this thesis showed that the proposed multilateration methods were able to mitigate the drawbacks of the conventional ultrasonic detection system, as presented in the Figure 1.12. A comparison of several multilateration methods, showed that all met the requirements on the precision and accuracy. An interesting finding was that simplifying the multilateration method with the exact circle solution lead to a decrease in runtime of the algorithm, but only a small decrease in precision and accuracy.

In a comparison of clustering algorithms, which were selected to cluster the

data and remove noise, both DBSCAN [25] and OPTICS [27] clustering showed good performance once their parameters were optimized to the data.

Finally, three different tracking methods were evaluated, while the GM-PHD [39] had slightly better performance in terms of performance metrics over the UKF-JPDA tracker [43], the latter proved to have tracks which better matched the ground truth tracks by visual inspection. A topic that this study did not cover was the optimization of the trackers, with optimization of the parameters the performance could probably be improved further.

The resulting tracks of the tracking algorithms show that automotive ultrasonic distance sensor could be used for close-range sensing and tracking. Taking into account the intended use of the proposed system, detection and tracking of (vulnerable) road users in stop-start scenarios, this study showed promising results.

6.1 Future work

During the course of exploring the different aspects of ultrasonic sensors for use in detection and tracking in autonomous vehicles, several new problems and ideas arose which could not be covered in this thesis. While the calibration method proposed in this thesis is thorough, it is a time consuming task. Research into automatic or dynamic calibration methods could be subjects beneficial in the development and performance of ultrasonic detection systems.

In the case study on the optimization of sensor positions in the front bumper of an autonomous vehicle, it was shown that the blind spots of the system decreased while ensuring the theoretical performance of the system did not decrease. A possible extension to this case study would be a verification of the detection performance on a real vehicle.

A problem that arises in the multilateration of range measurements, namely association of the range measurements of different sensors to the correct detection. This study went for a straightforward approach by using clustering methods, but multiple data association methods could still be explored here.

A topic that was not covered in this thesis is machine learning. The multiple sensors of an ultrasonic detection system would be an interesting input layer for a convolutional neural network. A comparison between the performance of such an approach and the performance of a classical detection and tracking approach, as presented in this thesis, would make an interesting topic for further research.

References

[1] S. Abbas and P. Peng, "Basic Principles and Physics of Ultrasound," in *Ultrasound for Interventional Pain Management*, Springer International Publishing, Sep. 2019, pp. 1–31. DOI: 10.1007/978-3-030-18371-4_1.

[2] J. Lewiner, "Paul Langevin and the Birth of Ultrasonics," *Japanese Journal of Applied Physics*, vol. 30, no. S1, p. 5, Jan. 1991. DOI: 10.7567/jjaps.30s1.5.

[3] K. Reif, Ed., *Fahrstabilisierungssysteme und Fahrerassistenzsysteme*. Vieweg Teubner, 2010. DOI: 10.1007/978-3-8348-9717-6.

[4] M. H. Repacholi, M. Grandolfo, and A. Rindi, *Ultrasound: Medical applications, biological effects, and hazard potential*. Plenum, 1987, pp. 49–55, ISBN: 9780306424113.

[5] J. K. Petersen and P. Taylor, *Handbook of Surveillance Technologies*. CRC press, 2012, pp. 195–222, ISBN: 9781439873168.

[6] B. Rodriguez and A. Whitehead. "Where are ultrasonic sensors used? - part 2." (2017), [Online]. Available: https://e2e.ti.com/blogs_/b/analogwire/posts/where-are-ultrasonic-sensors-used-part-2 (visited on 2023-08-13).

[7] V. Magori, "Ultrasonic sensors in air," in *1994 Proceedings of IEEE Ultrasonics Symposium*, vol. 1, Oct. 1994, 471–481 vol.1. DOI: 10.1109/ULTSYM.1994.401632.

[8] "ISO 17386:2023: Intelligent transport systems — Manoeuvring aids for low-speed operation (MALSO) — Performance requirements and test procedures," International Organization for Standardization, Geneva, CH, Standard, May 2023.

[9] F. Niewels, S. Knoop, R. Jordan, and T. Michalke, "In-vehicle sensors," in *Encyclopedia of Automotive Engineering*. John Wiley & Sons, Ltd, 2013, pp. 1–27, ISBN: 9781118354179. DOI: https://doi.org/10.1002/9781118354179.auto175.

[10] C. Connette, J. Fischer, B. Maidel, *et al.*, "Rapid detection of fast objects in highly dynamic outdoor environments using cost-efficient sensors," in *ROBOTIK 2012; 7th German Conference on Robotics*, May 2012, pp. 1–4.

[11] F. Mirus, J. Pfadt, C. Connette, B. Ewert, D. Grüdl, and A. Verl, "Detection of moving and stationary objects at high velocities using cost-efficient sensors, curve-fitting and neural networks," Oct. 2012. [Online]. Available: https://www.elektromobilitaet.fraunhofer.de/content/dam/elektromobilitaet/de/documents/fsem_ii/fmirus_iros12_detection_of_moving_and_stationary_objects_at_high_velocities_using_cost-efficient_sensors_curve_fitting_and_neural_networks.pdf (visited on 2023-08-13).

[12] P. Köhler, C. Connette, and A. Verl, "Vehicle tracking using ultrasonic sensors joined particle weighting," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 2900–2905. DOI: 10.1109/ICRA.2013.6630979.

[13] A. T. Vemuri, "Using a fixed threshold in ultrasonic distance-ranging automotive applications," *Analog Applications Journal*, pp. 19–23, Q3 2012. [Online]. Available: https://www.ti.com/lit/an/slyt481/slyt481.pdf (visited on 2023-08-13).

[14] O. Cramer, "The variation of the specific heat ratio and the speed of sound in air with temperature, pressure, humidity, and CO_2 concentration," *The Journal of the Acoustical Society of America*, vol. 93, no. 5, pp. 2510–2516, 1993. DOI: 10.1121/1.405827.

[15] S. A. R. Zekavat and R. M. Buehrer, Eds., *Handbook of Position Location*. Wiley, Mar. 2018. DOI: 10.1002/9781119434610.

[16] A. Markel. "ADAS calibration: Sensor operation & calibration myths." (2022), [Online]. Available: https://www.tirereview.com/adas-calibration-sensor-opoeration/ (visited on 2023-06-05).

[17] i-car. "Typical calibration requirements for park assist sensors." (2016), [Online]. Available: https://rts.i-car.com/collision-repair-news/crn-474.html (visited on 2023-06-05).

[18] E. N. Mazzae and W. R. Garrott, "Evaluation of the performance of available backover prevention technologies for light vehicles," in *20th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, 2007. [Online]. Available: https://www.nhtsa.gov/document/experimental-evaluation-performance-available-backover-prevention-technologies-medium (visited on 2023-08-13).

[19] G. Fusco and H. Gupta, "Selection and orientation of directional sensors for coverage maximization," in *2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, Jun. 2009, pp. 1–9. DOI: 10.1109/SAHCN.2009.5168968.

[20] J. V. Kleef, J. Bergmans, L. Kester, and F. Groen, "Multiple-hypothesis trilateration and tracking with distributed radars," in *2006 9th International Conference on Information Fusion*, Jul. 2006, pp. 1–7. DOI: 10.1109/ICIF.2006.301617.

[21] J. Domhof, R. Happee, and P. Jonker, "Multi-sensor object tracking performance limits by the Cramer-Rao lower bound," in *2017 20th International Conference on Information Fusion (Fusion)*, 2017, pp. 1–8. DOI: 10.23919/ICIF.2017.8009637.

[22] D. Moreno-Salinas, A. Pascoal, and J. Aranda, "Optimal sensor placement for acoustic underwater target positioning with range-only measurements," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 3, pp. 620–643, Jul. 2016. DOI: 10.1109/JOE.2015.2494918.

[23] W. Y. Tan, T. Steiner, and N. V. Ruiter, "Automatic optimization of sensor positioning for an airborne ultrasound imaging system," in *2016 IEEE International Ultrasonics Symposium (IUS)*, Sep. 2016, pp. 1–4. DOI: 10.1109/ULTSYM.2016.7728392.

[24] O. Wijk, P. Jensfelt, and H. I. Christensen, "Triangulation based fusion of ultrasonic sensor data," in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 4, May 1998, pp. 3419–3424. DOI: 10.1109/ROBOT.1998.680966.

[25] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96, Portland, Oregon: AAAI Press, 1996, pp. 226–231.

[26] R. J. G. B. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Advances in Knowledge Discovery and Data Mining*, Springer Berlin Heidelberg, 2013, pp. 160–172. DOI: 10.1007/978-3-642-37456-2_14.

[27] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: Ordering points to identify the clustering structure," in *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '99, Philadelphia, Pennsylvania, USA: Association for Computing Machinery, 1999, pp. 49–60, ISBN: 1581130848. DOI: 10.1145/304182.304187.

[28] L. McInnes, J. Healy, and S. Astels. "How HDBSCAN works." (2016), [Online]. Available: https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html (visited on 2023-06-05).

[29] scikit-learn. "sklearn.cluster.OPTICS." (2023), [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.OPTICS.html (visited on 2023-06-05).

[30] A. Yilmaz, O. Javed, and M. Shah, "Object tracking," *ACM Computing Surveys*, vol. 38, no. 4, p. 13, Dec. 2006. DOI: 10.1145/1177352.1177355.

[31] W. Lu, Z. Zhou, L. Zhang, and G. Zheng, "Multi-target tracking by nonlinear motion patterns based on hierarchical network flows," *Multimedia Systems*, vol. 25, no. 4, pp. 383–394, Apr. 2019. DOI: 10.1007/s00530-019-00614-y. [Online]. Available: https://doi.org/10.1007/s00530-019-00614-y.

[32] B. Yang and R. Nevatia, "Multi-target tracking by online learning of nonlinear motion patterns and robust appearance models," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1918–1925. DOI: 10.1109/CVPR.2012.6247892.

[33] Bosch, *Detailed installation instructions Rear – Ultrasonic ParkPilot*, 2009.

[34] M. Dolatabadi, J. Elfring, and R. van de Molengraft, "Multiple-joint pedestrian tracking using periodic models," *Sensors*, vol. 20, no. 23, p. 6917, Dec. 2020. DOI: 10.3390/s20236917.

[35] H. W. Sorenson, *Kalman filtering: Theory and application*. IEEE Press, 1985, ISBN: 9780879421915.

[36] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Signal Processing, Sensor Fusion, and Target Recognition VI*, I. Kadar, Ed., International Society for Optics and Photonics, vol. 3068, SPIE, 1997, pp. 182–193. DOI: 10.1117/12.280797.

[37] E. Wan and R. Van Der Merwe, "The unscented kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, 2000, pp. 153–158. DOI: 10.1109/ASSPCC.2000.882463.

[38] S. E. Li, G. Li, J. Yu, *et al.*, "Kalman filter-based tracking of moving objects using linear ultrasonic sensor array for road vehicles," *Mechanical Systems and Signal Processing*, vol. 98, pp. 173–189, Jan. 2018. DOI: 10.1016/j.ymssp.2017.04.041.

[39] B.-N. Vo and W.-K. Ma, "The Gaussian mixture probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, Nov. 2006. DOI: 10.1109/tsp.2006.881190.

[40] D. E. Clark, K. Panta, and B.-N. Vo, "The GM-PHD filter multiple target tracker," in *2006 9th International Conference on Information Fusion*, 2006, pp. 1–8. DOI: 10.1109/ICIF.2006.301809.

[41] B. Ristic, D. Clark, B.-N. Vo, and B.-T. Vo, "Adaptive target birth intensity for PHD and CPHD filters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 2, pp. 1656–1668, 2012. DOI: 10.1109/taes.2012.6178085.

[42] L. Rakai, H. Song, S. Sun, W. Zhang, and Y. Yang, "Data association in multiple object tracking: A survey of recent techniques," *Expert Systems with Applications*, vol. 192, p. 116 300, Apr. 2022. DOI: 10.1016/j.eswa.2021.116300.

[43] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *IEEE Control Systems Magazine*, vol. 29, no. 6, pp. 82–100, 2009. DOI: 10.1109/MCS.2009.934469.

[44] R. S. Davis, "Equation for the determination of the density of moist air (1981/91)," *Metrologia*, vol. 29, no. 1, p. 67, Jan. 1992. DOI: 10.1088/0026-1394/29/1/008.

[45] R. Lord. "Calculation of speed of sound in humid air." (2005), [Online]. Available: http://resource.npl.co.uk/acoustics/techguides/speedair/ (visited on 2023-05-19).

[46] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005, ISBN: 0262201623. DOI: 10.5555/1121596.

[47] P. Bourke. "Spheres, equations and terminology." (1992), [Online]. Available: http://paulbourke.net/geometry/circlesphere/ (visited on 2023-06-05).

[48] J. Richter-Gebert, *Perspectives on Projective Geometry*. Springer Berlin Heidelberg, 2011, pp. 196–207. DOI: 10.1007/978-3-642-17286-1.

[49] O. Herman-Saffar. "An approach for choosing number of clusters for k-means." (2021), [Online]. Available: https://towardsdatascience.com/an-approach-for-choosing-number-of-clusters-for-k-means-c28e614ecb2c (visited on 2023-06-05).

[50] Point Cloud Library. "Euclidean cluster extraction." version 6c23c854. (2023), [Online]. Available: https://pcl.readthedocs.io/en/latest/cluster_extraction.html (visited on 2023-07-30).

[51] T. Vu and R. Evans, "A new performance metric for multiple target tracking based on optimal subpattern assignment," in *17th International Conference on Information Fusion (FUSION)*, 2014, pp. 1–8.

[52] Bosch, *BME280 Combined humidity and pressure sensor*, 2022. [Online]. Available: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme280-ds002.pdf (visited on 2023-06-28).

[53] Arduino S.r.l. "Arduino® UNO R3 – Product Reference Manual." (2005), [Online]. Available: https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf (visited on 2023-06-28).

Appendix A

Python code for speed of sound in air

```
import numpy as np

def speed_of_sound(T: float, Rh: float, p: float) -> float:
    r"""Compute speed of sound in air

    Code from: ‘Dr. Richard Lord‘ -
    http://www.npl.co.uk/acoustics/techguides/speedair

    Based on the approximate formula found in :footcite:t:‘cramer
    :1993‘.
    Saturation vapour pressure found in :footcite:t:‘davis:1992‘,
    and a
    mole fraction of carbon dioxide of 0.0004.
    The mole fraction is simply an expression of the number of
    moles of
    a compound divided by the total number of moles of all the
    compounds
    present in the gas.

    Args:
        T: Temperature :math:‘[^{\circ} \text{C}]‘
        Rh: Relative humidity :math:‘[\%]‘
        p: pressure :math:‘[\text{Pa}]‘

    Returns:
        speed of sound :math:‘C\:[\frac{\mathrm{m}}{\mathrm{s}}]‘

    .. footbibliography::

    """
    e = 2.71828182845904523536
    T_kel = 273.15 + T

    # Molecular concentration of water vapour calculated from Rh
    # using Giacomos method by Davis (1991) as implemented in DTU
    report 11b-1997
    ENH = 3.14e-8 * p + 1.00062 + T*T * 5.6e-7
    # These commented lines correspond to values used in Cramer (
    Appendix)
    # PSV1 = sqrt(T_kel)*1.2811805*pow(10,-5)-1.9509874*pow(10,-2)*
```

```
    T_kel ;
35  # PSV2 = 34.04926034-6.3536311*pow(10,3)/T_kel;
36  PSV1 = T_kel*T_kel * 1.2378847e-5 - 1.9121316e-2 * T_kel
37  PSV2 = 33.93711047 - 6.3431645e3 / T_kel
38  PSV = e**PSV1 * e**PSV2
39  H = Rh * ENH * PSV / p
40  Xw = H / 100.0
41  # Xc = 314.0*pow(10,-6);
42  Xc = 400.0e-6
43  # Speed calculated using the method
44  # of Cramer from JASA vol 93 p. 2510
45  C1 = 0.603055 * T + 331.5024 - T*T * 5.28e-4 + (0.1495874 * T +
     51.471935 - T*T * 7.82e-4) * Xw
46  C2 = (-1.82e-7 + 3.73e-8 * T - T*T * 2.93e-10) * p + (-85.20931
     - 0.228525 * T + T*T * 5.91e-5) * Xc
47  C3 = Xw*Xw * 2.835149 + p*p * 2.15e-13 - Xc*Xc * 29.179762 -
    4.86e-4 * Xw * p * Xc
48  C = C1 + C2 - C3
49
50  return C
```

Listing A.1: Python function to approximate the speed of sound in air.

Appendix B

Finding the points of intersection of two ellipses

For this the general quadratic form for an ellipse is created

$$Q(x, y) = Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0 \tag{B.1}$$

with

$$\begin{aligned} a &= x \\ b &= r_x^2 \\ c &= y \\ d &= r_y^2 \\ A' &= \cos\theta \\ B' &= \sin\theta \end{aligned}$$

where (x, y) is the center of the ellipse, r_x and r_y the radii in horizontal and vertical direction respectively and θ the rotation around the center. From this follows

$$\begin{aligned} A &= \frac{A'^2}{b} + \frac{B'^2}{d} \\ B &= \frac{2A'B'}{d} - \frac{2A'B'}{b} \\ C &= \frac{A'^2}{d} + \frac{B'^2}{b} \\ D &= \frac{2A'B'c - 2aA'^2}{b} + \frac{-2aB'^2 - 2A'B'c}{d} \\ E &= \frac{2aA'B' - 2B'^2c}{b} + \frac{-2aA'B' - 2A'^2c}{d} \\ F &= \frac{a^2A'^2 - 2aA'B'c + B'^2c^2}{b} + \frac{a^2B'^2 + aA'B'c + A'^2c^2}{d} - 1 \end{aligned}$$

With the quadratic equations for both ellipses

$$q_1(x,y) = a_1x^2 + b_1xy + c_1y^2 + d_1x + e_1y + f_1$$
$$q_2(x,y) = a_2x^2 + b_2xy + c_2y^2 + d_2x + e_2y + f_2$$

we can construct general quartic equation

$$f(x) = z_4x^4 + z_3x^3 + z_2x^2 + z_1x + z_0 \tag{B.2}$$

where

$$\begin{aligned}
z_0 &= a_1d_2^2f_1 + a_1^2f_2^2 - a_1d_1d_2f_2 + a_2^2f_1^2 - 2a_1a_2f_1f_2 - a_2d_1d_2f_1 + a_2d_1^2f_2 \\
z_1 &= a_2d_1^2e_2 - a_1b_1d_2f_2 - 2a_1a_2f_2e_1 - a_2b_2d_1f_1 + 2a_1b_2d_2f_1 + 2a_1^2e_2f_2 + a_1d_2^2e_1 - a_1e_2d_1d_2 \\
&\quad - 2a_1a_2e_2f_1 - a_2b_1d_2f_1 + 2a_2^2e_1f_1 - a_1b_2d_1f_2 - a_2d_1d_2e_1 + 2a_2b_1d_1f_2 \\
z_2 &= a_1^2e_2^2 + 2a_1^2c_2f_2 - a_2b_1d_2e_1 + a_2b_1^2f_2 - a_2b_2d_1e_1 - a_1b_1b_2f_2 - 2a_1a_2e_1e_2 + 2a_1b_2d_2e_1 - a_1c_2d_1d_2 \\
&\quad - 2a_1a_2c_2f_1 + a_1b_2^2f_1 + 2a_2b_1d_1e_2 + a_2^2e_1^2 - a_2c_1d_1d_2 - a_1b_2d_1e_2 + 2a_2^2c_1f_1 - a_2b_1b_2f_1 + a_2c_2d_1^2 \\
&\quad + a_1d_2^2c_1 - a_1b_1d_2e_2 - 2a_1a_2c_1f_2 \\
z_3 &= -2a_1a_2c_1e_2 + a_2b_1^2e_2 + 2a_2b_1c_2d_1 - a_2b_2c_1d_1 + a_1b_2^2e_1 - a_1b_1b_2e_2 - 2a_1a_2c_2e_1 - a_2b_1b_2e_1 \\
&\quad - a_1b_2c_2d_1 + 2a_1^2c_2e_2 + 2a_2^2c_1e_1 - a_2b_1c_1d_2 + 2a_1b_2c_1d_2 - a_1b_1c_2d_2 \\
z_4 &= a_1^2c_2^2 - 2a_1a_2c_1c_2 + a_2^2c_1^2 - a_1b_1b_2c_2 - a_2b_1b_2c_1 + a_2b_1^2c_2 + a_1b_2^2c_1
\end{aligned}$$

Now to find y, we need to find the roots of the quartic equation

$$z_4x^4 + z_3x^2 + z_2x^2 + z_1x + z_0 = 0 \tag{B.3}$$

Dividing by z_4

$$\begin{aligned}
b &= \frac{z_3}{z_4} \\
c &= \frac{z_2}{z_4} \\
d &= \frac{z_1}{z_4} \\
e &= \frac{z_0}{z_4}
\end{aligned}$$

gives $x^4 + bx^3 + cx^2 + dx + e = 0$. Substituting $\left(y - \frac{b}{4}\right)$ for x gives the depressed quartic equation

$$y^4 + py^2 + qy + r = 0 \tag{B.4}$$

where

$$
\begin{aligned}
p &= \frac{8c - 3b^2}{8} \\
q &= \frac{b^3 - 4bc + 8d}{8} \\
r &= \frac{-3b^4 + 256e - 64bd + 16b^2c}{256}
\end{aligned}
$$

Now

$$
\begin{aligned}
P &= -\frac{1}{12}p^2 - r \\
Q &= -\frac{1}{108}p^3 + \frac{1}{3}pr - \frac{1}{8}q^2 \\
U &= \left(-\frac{1}{2}Q + \sqrt{\frac{1}{4}Q^2 + \frac{1}{27}P^3}\right)^{\frac{1}{3}} \\
y &= \begin{cases} -\frac{5}{6}p - Q^{\frac{1}{3}} & \text{if } P = 0 \\ -\frac{5}{6}p + U - \frac{1}{3}\frac{P}{U} & \text{otherwise} \end{cases}
\end{aligned} \tag{B.5}
$$

$$
w = \sqrt{p + 2y}
$$

$$
x_1 = -\frac{1}{4}b + \frac{1}{2}\left((1)w + (1)\sqrt{-(p+2y) - 2\left(p + (1)\frac{q}{w}\right)}\right) \tag{B.6}
$$

$$
x_2 = -\frac{1}{4}b + \frac{1}{2}\left((-1)w + (1)\sqrt{-(p+2y) - 2\left(p + (-1)\frac{q}{w}\right)}\right) \tag{B.7}
$$

$$
x_3 = -\frac{1}{4}b + \frac{1}{2}\left((1)w + (-1)\sqrt{-(p+2y) - 2\left(p + (1)\frac{q}{w}\right)}\right) \tag{B.8}
$$

$$
x_4 = -\frac{1}{4}b + \frac{1}{2}\left((-1)w + (-1)\sqrt{-(p+2y) - 2\left(p + (-1)\frac{q}{w}\right)}\right) \tag{B.9}
$$


TWIZY
NL
74
XFX-8