Multipath Modelling for 60 GHz Short Range Radar

Thesis for the degree of Bachelor of Science,
Project Duration: 4 months

André Nüßlein

LUNDS
UNIVERSITET

Supervisors: Andreas Jakobsson
Department of Mathematics
Division: Mathematical Statistics
January 2021

Abstract

If radar surveillance systems are installed near a wall, their radiation is reflected not only from moving objects but also from the wall. In this way, non-existing so-called ghost targets can lead to false detections. This thesis exploits such ghost detections to infer the position and extension of a wall next to a radar sensor. This is done by geometric equations only and proves especially successful outdoors. With this knowledge, it will be possible to distinguish ghost targets from real ones and even use them for improved object detection.

Acknowledgements

I would first like to thank Andreas Jakobsson for accepting to supervise me at the end of the summer, with not much time left before the start of the new semester. Fredrik Tufvesson acted as an unofficial co-supervisor for which I am very grateful. I would also like to thank Sebastian and Anders who helped me various times a week, every week throughout the semester. Special thanks are extended to the rest of my colleagues who welcomed me without hesitation.

Abbreviations

ADC	analog-to-digital converter
AoA	angle of arrival
FFT	fast Fourier transform
FMCW	frequency-modulated continuous wave
IF	intermediate frequency
IO	interacting object
LOS	line-of-sight
RCS	radar cross-section
RX	receiver antenna
SNR	signal-to-noise ratio
TX	transmitter antenna

Contents

1. Introduction

When hearing the word “radar”, many people would probably immediately think of a detection system used mostly on airplanes or vessels. Maybe they could even guess that it is an acronym for “radio detection and ranging”, and know that it needs an antenna sending out electromagnetic waves in the microwave domain (radio waves). These waves then reflect back from the interacting objects (IO) to a second antenna waiting for the echoes to be received (see figure 1.1). The usage on airplanes and on water is rather simple because there are limited background scatterers such as walls or ceilings, and detectable objects are scarce. It is therefore not surprising that radar was pioneered in the Second World War [1] less than a century after electromagnetic waves were first postulated and experimentally discovered by James Clerk Maxwell and Heinrich Hertz, respectively [2].

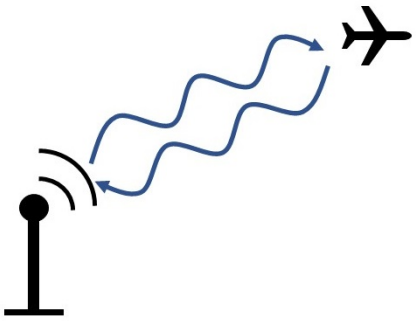


Figure 1.1: An antenna transmits a radio wave and detects its reflection from an airplane (in practice, transmitter antenna (TX) and receiver antenna (RX) are often two different devices).

Since then, radar technologies have found their applications in weather forecasting, police radar guns for detecting speeders, and automotive sensors [3]. Recently, stationary radar surveillance systems have conquered the market [4]. Compared to video surveillance, radar can detect perpetrators also in foggy conditions, and it is often considered as an alternative when privacy is important (an example could be public swimming pools).

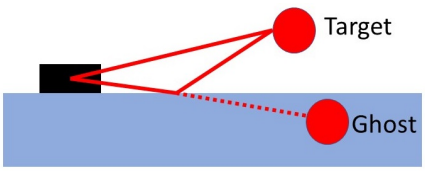


Figure 1.2: A radar (black box) detects both a target and its reflection in the ground (ghost target).

A radar sensor does, however, encounter difficulties when placed in dense environments such as narrow streets, and even more so indoors. One issue is named “multipath” because it is due to the electromagnetic waves reflecting from walls, thus taking more than just the line-of-sight (LOS) path between the antenna and IO. This leads to destructive and constructive interference effects, and to the appearance of ghost targets (see figure 1.2). Ghosts can lead to false detection but, especially if the floor-plan is known, they can help to increase the signal-to-noise ratio (SNR) [5]. In automotive applications, ghosts can be distinguished by comparing their orientation to their velocity [6].

Without knowing the floor plan and the targets’ orientation, this thesis tries to identify and exploit ghost targets that appear when a radar sensor is placed next to a wall. In particular, this was done using a commercially available evaluation platform, with which short measuring

campaigns were undertaken indoors and outdoors.

The thesis is structured as follows: the theory section includes radar fundamentals, basic signal processing, and electromagnetic wave theory. The measurement and analysis will be thoroughly explained in the method section. The results and their discussion conclude the thesis.

2. Background and Theory

The radar sensors used for this thesis transmit electromagnetic waves with a wavelength λ in the millimeter range. This allows for the antennas to be very compact (of the order of λ) and make highly accurate measurements. For example, an antenna with frequency $f = 60\,\mathrm{GHz}$ sends out waves with $\lambda \approx 5\,\mathrm{mm}$ (found via $\lambda = c/f$, where c is the speed of light) and can easily detect movements in centimeter resolution.

However, to understand the workings of a radar sensor in detail, one needs to first understand the theory of electromagnetic waves. The theory section thus starts with stating the wave equation for electromagnetic waves in both its real and complex notation. Section 2.2 then explains the signal processing going on inside of a radar sensor. This is followed by a deterministic analysis of the simplest multipath scenario - scattering on an even surface.

2.1 Electromagnetic waves

Electromagnetic waves are oscillations of electric and magnetic fields in sync with each other [7]. They travel at the speed of light c and, generally, the two fields are perpendicular to each other and perpendicular to the direction of propagation. The time dependent wave equations are solutions to Maxwell's equations and are sinusoidal. For the electric field, this means

$$\boldsymbol{E} = \boldsymbol{E}_0 \exp[j(2\pi f t + \phi)], \tag{2.1}$$

where $\boldsymbol{E}_0$ is the amplitude, t is time, and ϕ is a phase term. If an antenna emits such a radio wave whose phase is considered zero at its origin, at a distance d it will have a phase of $\phi = 2\pi d f/c$. The imaginary part of equation 2.2 is the physical wave:

$$\boldsymbol{E} = \boldsymbol{E}_0 \sin(2\pi f t + \phi), \tag{2.2}$$

which is also a solution to Maxwell's equations. In practice, it is convenient to use its complex representation and only evaluate its real part when necessary.

It often suffices to consider the waves to be uniform plane waves, which means that the fields are independent of their transverse coordinates and that their propagation vector follows simple geometric rules. For our purposes, the most important one is the law of reflection. It states that the angle of reflection is equal to the angle of incidence. Knowing this, basic trigonometric identities can be exploited to measure distances. Exactly how this is done is explained after the treatment of radar fundamentals.

2.2 FMCW Radar Fundamentals

By capturing the reflected signal and mixing it with the TX signal, a radar can easily determine the range, velocity, and angle of moving objects. How the first two parameters can be deduced will be treated schematically in the following subsections, and the angle analysis will be explained even more qualitatively. A more detailed account can be found in [8].

2.2.1 Range Measurement

The devices used in this thesis are frequency-modulated continuous wave (FMCW) radars. Their signals are called chirps because their frequency increases linearly with time. A chirp configuration is almost completely characterized by its start frequency f_c, bandwidth B, the chirp to chirp time T_c and slope S. Figure 2.1 shows the frequency of such a signal as a function of time with typical values for the aforementioned parameters. The figure depicts two sent signals (TX) and the first received signal (RX). The latter is the same as the TX signal but shifted in time because it must travel to and from an IO. The time delay is simply $\tau = 2d/c$, where d is the distance to the IO, and c is the speed of light. In one *frame* multiple signals are sent out after each other, but for the range measurement only the first one is of interest.

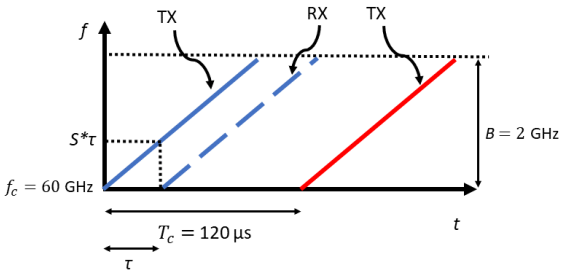


Figure 2.1: Chirp signal with frequency as a linear function of time with slope S.

As a first step in the signal processing, a *frequency mixer* combines the RX and the TX signal to an intermediate frequency (IF) signal. Because these two inputs are both sinusoidal with characteristic angular frequencies ω_i and and phases ϕ_i (see equation 2.2), the mixer's output is also sinusoidal:

$$x_{out} = \sin\left[(\omega_1 - \omega_2)t + (\phi_1 - \phi_2)\right]. \tag{2.3}$$

In the time interval where the TX chirp overlaps with the RX chirp, the IF signal has frequency

$$f_0 = S \cdot \tau = \frac{S2d}{c}, \tag{2.4}$$

which can be seen graphically by looking at figure 2.1. More explicitly, this is the TX frequency when the RX signal first arrives. Similarly, the initial phase of the IF signal is

$$\phi_0 = 2\pi f_c \tau = \frac{4\pi d}{\lambda}. \tag{2.5}$$

When there are several objects, the chirp is delayed by a different amount of time proportional to the distance to each object. Hence, the IF signal consists of different tones. Fourier transforming this signal reveals the signal's different frequencies. In practice, this is done using a fast Fourier transform (FFT) algorithm. The different frequencies correspond to the distances to the IOs. If two IOs are at the same distance from the radar but at different angles, they will share one peak. How these can be distinguished will be shown later on, where what has been done so far will be referred to as range-FFT processing. From Fourier transform theory, it can be shown that the range resolution depends only on the bandwidth [8]:

$$d_{res} = \frac{c}{2B}. \tag{2.6}$$

To give an example, one of the chirp configurations used for this thesis had a resolution of roughly $6\,\mathrm{cm}$. Theoretically, a radar can see very far, but the frequency change becomes higher for objects further away. Real radars are equipped with an analog-to-digital converter (ADC), which can sample up to a maximum frequency. The maximum distance a radar can see is

$$d_{max} = \frac{F_s c}{2S}, \tag{2.7}$$

where F_s is the ADC sampling rate.

2.2.2 Velocity Measurement and Range-Doppler Plot

To measure the velocity v, two chirps separated by T_c are transmitted. Note from figure 2.1 that the second signal starts some time after the end of the first signal; this is called the idle time. Because the signal can be assumed to travel much faster than the IO, the range FFT will have peaks in the same location, but with a different phase. This difference is derived from equation 2.5 as

$$\Delta\phi = \frac{4\pi v T_c}{\lambda}. \tag{2.8}$$

Since the velocity measurement is based on the phase difference, the measurement is unambiguous only if $|\Delta\phi| < \pi$. From this, the maximum relative speed measured by two chirps spaced T_c apart can be determined to be

$$v_{max} = \frac{\lambda}{4T_c}. \tag{2.9}$$

If multiple objects with different velocities are at the same distance from the radar, the range-FFT will result in a single peak and the phase comparison technique will not work. This can be reconciled by transmitting a frame of N equally spaced chirps ($N = 128$ for example). Consider the example of two equidistant objects with velocities v_1 and v_2. Range-FFT processing their set of IF signals results in a set of N identically located peaks, each with a different phase.

These are sketched in grey in figure 2.2a. When a second FFT is performed over the N peaks, the result is a function with local maxima at two different frequencies f_1 and f_2, marked in color in figure 2.2a. This second FFT is also called the Doppler-FFT. The velocities can then be computed via

$$v_i = \frac{\lambda f_i}{4\pi T_c}. \tag{2.10}$$

From the theory of discrete Fourier transforms, it can then be shown that the velocity resolution is

$$v_{res} = \frac{\lambda}{2T_f}, \tag{2.11}$$

where $T_f = NT_c$.

All previously mentioned peaks actually appear as bins in a discrete matrix as shown in figures 2.2b and c. The final matrix is referred to as the Range-Doppler plot.

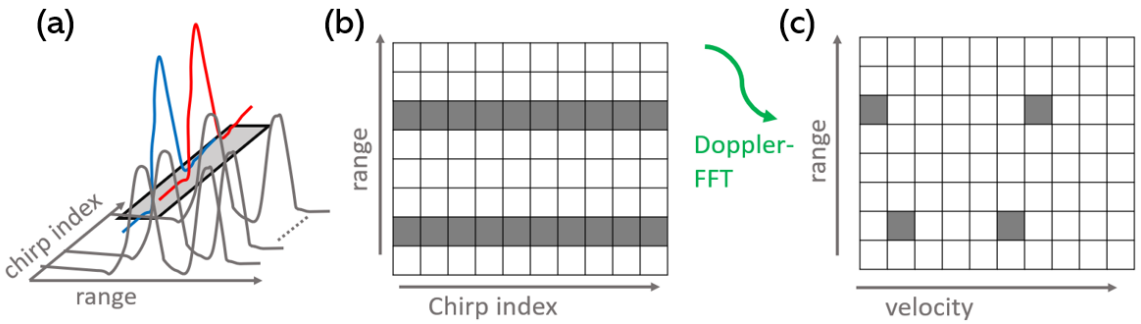


Figure 2.2: (a) A range-FFT gives the grey curves, a subsequent Doppler-FFT results in the colorful curves. (b) The peaks of the grey curves appear as dark range bins. (c) The Doppler-FFT on the rows of the first matrix results in a range velocity plot.

2.2.3 Angle measurement

An FMCW radar system can also estimate the angle of a reflected signal in the horizontal plane. This angle of arrival (AoA) approximation is based on the fact that a small change in the distance to an object results in a significant phase change in the peak of the range-FFT or Doppler-FFT. Therefore, two RX placed next to each other such as in figure 2.3 will result in two distinguishable signals.

In practice, one can also place several TX next to each other, and this thesis uses two TX and four RX, resulting in eight Range-Doppler matrices. This method is called multiple-input and multiple-output (MIMO) and is used heavily in wireless communication [9]. The eight matrices are combined to a three-dimensional matrix, sketched in figure 2.4, and performing an angle-FFT over the range-velocity bins along the layers reveals the phase ϕ of the signal. If

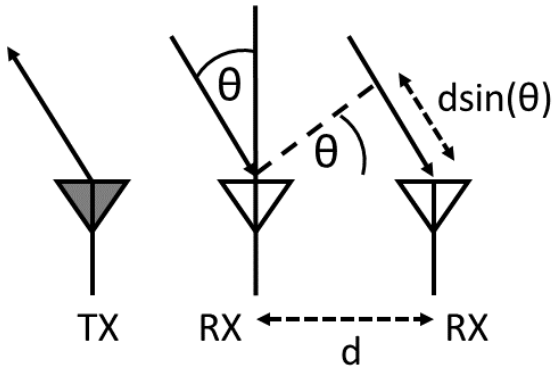


Figure 2.3: Sketch of an antenna array to resolve the AoA.

there are two objects at different angles but with the same velocity and range, there will be two distinct peaks after this last FFT. The angle(s) can then be found by

$$\theta = \arcsin\left(\frac{\lambda\phi}{2\pi d}\right), \tag{2.12}$$

where d here is the RX spacing. If $d = \lambda/2$, the field of view is maximized and the above equation simplifies conveniently.

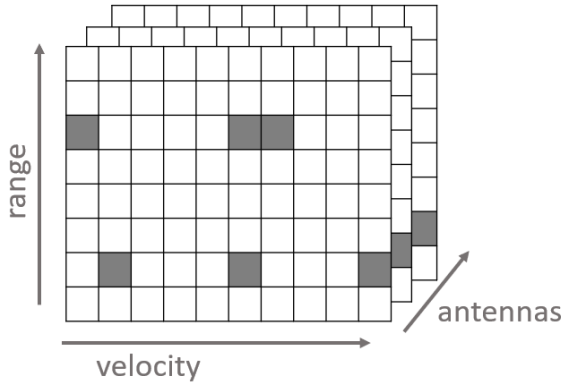


Figure 2.4: The range velocity cube along whose third dimension the angle-FFT is performed.

2.2.4 Radar Range Equation

The radar range equation deterministically relates the received echo power P_r to transmitted power P_t in terms of design parameters and distance d between radar and IO without taking into account reflections from randomly distributed obstacles. Its simplest derivation assumes a point target [10]. This target collects all energy in an area around it, called the radar cross section (RCS) σ, and reradiates it isotropically. From these considerations, it can be shown that

$$P_r = \frac{P_t G^2 \lambda^2 \sigma}{(4\pi)^3 d^4}, \tag{2.13}$$

where G is the antenna gain, taking into account that the antenna does not radiate equally in all directions. While not derived here, it is clear that the d^4 and two of the three (4π) in the denominator stem from both the antenna and the target radiating isotropically (the third (4π) is a result of how the antenna receives the signal).

A slightly more involved derivation for volume targets [10] gives that

$$P_r \propto \frac{\eta}{d^2}, \tag{2.14}$$

where η is the volume reflectivity. For our purposes, it is not instructive to follow the complete derivation because an actual IO cannot be considered a point target nor a perfect volume target. It might best be approximated by a dominant volume target and a few point targets. The expectation for measurement is thus that $P_r \propto d^{-n}$ with $1 < n < 4$. The exact behaviour depends on the IO's shape and its orientation.

2.2.5 Specular Ground Reflection

As soon as multipath is introduced, even a metallic sphere as the IO would not perfectly reproduce equation 2.13. At high enough distance, the range equation will actually be proportional to d^{-8}, as will be shown in the following derivation for the simplest multipath case, which is adapted from [11].

Consider the geometry of figure 2.5, where the transmit antenna is placed slightly higher than the moving target. By the law of reflection, the two angles marked in the figure are identical. The LOS path length from TX to IO is denoted by d_{direct} while the length that E_{refl} covers is d_{refl}. From basic trigonometry, these are

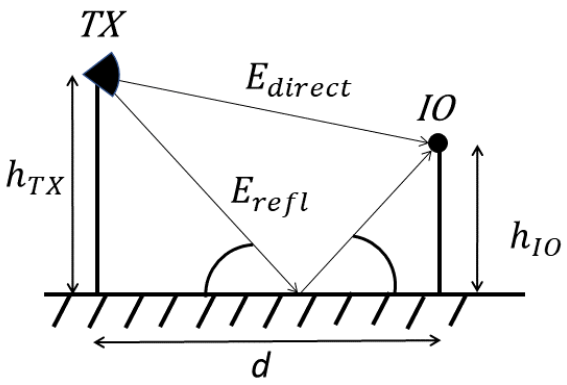


Figure 2.5: The transmit antenna placed slightly higher above ground than the moving IO. They are a distance d apart and the two different paths the wave can take have length d_{direct} and d_{refl}.

$$d_{direct} = \sqrt{(h_{TX} - h_{IO})^2 + d^2}, \quad d_{refl} = \sqrt{(h_{TX} + h_{IO})^2 + d^2}.$$

The field strengths at the IO for the two paths are

$$E_{direct} = E\frac{1}{d_{direct}} \exp\left[j\left(2\pi f_c t - 2\pi f_c \frac{d_{direct}}{c}\right)\right], \tag{2.15}$$

$$E_{refl} = (-1)E\frac{1}{d_{refl}} \exp\left[j\left(2\pi f_c t - 2\pi f_c \frac{d_{refl}}{c}\right)\right], \tag{2.16}$$

where E is the amplitude and the factor (-1) is the ground reflection coefficient, approximately true at an angle of incidence close to 90° [12]. When the heights are smaller than the distance d, the amplitudes of both waves are approximately the same. At the IO the total field is the sum of both:

$$E_{tot}(d) = E\frac{1}{d} \exp\left[j\left(2\pi f_c t - 2\pi f_c \frac{d_{direct}}{c}\right)\right]\left\{1 - \exp\left[-j\left(2\pi f_c t \frac{d_{refl} - d_{direct}}{c}\right)\right]\right\}. \tag{2.17}$$

The received power at the IO is proportional to the square of the magnitude of the field strength:

$$|E_{tot}|^2 \propto \frac{1}{d^2}\left|1 - \exp\left[-j\left(2\pi f_c t \frac{d_{refl} - d_{direct}}{c}\right)\right]\right|^2, \tag{2.18}$$

which, at short distances, is proportional to d^{-2} with perturbations. A result of destructive interference, these perturbations can be quite important and are also referred to as *fast fading*. At longer distances, equation 2.18 is proportional to d^{-4}, which can be seen by plotting equation 2.18 or by Taylor expanding it [11]. Since the reflection follows the same path, the power at the receiving antenna is simply proportional to the square of equation 2.18 and thus eventually decays as d^{-8}.

In principle, this derivation also holds for a specular wall reflection. However, since the reflection depends on whether the electric field is parallel or orthogonal to the reflecting surface, the two cases might differ in practice.

2.2.6 Wall reflection

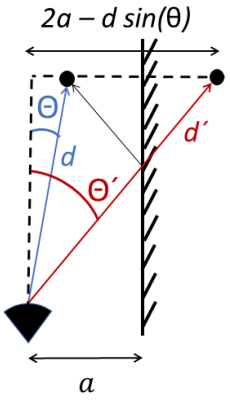


Figure 2.6: Top-view of an IO moving next to a wall.

When a radar keeps the the area around a modern office building under surveillance, its dominant multipath component will probably be due to the reflection in the windows rather than in the ground. This simple situation is sketched in figure 2.6. In principle, the multipath components can occur when the signal travels to the object via LOS, then to the wall, and then back (direct/indirect), or first to the wall, then to the object, and then back via LOS (indirect/direct), or by taking the indirect path in both directions (indirect/indirect). The last is attenuated by two reflections and is generally not dominant, but it dominates if corner reflectors are used (see method section). In this case, the ghost's measured distance d_{ghost} should be equal to d' indicated in the figure. This and other parameters can be found by simple trigonometry [13]:

$$d' = \sqrt{4a^2 + d^2 - 4ad\sin\theta}, \tag{2.19}$$

$$\theta' = \arctan\left(\frac{2a - d\sin\theta}{d\cos\theta}\right), \tag{2.20}$$

$$v_r' = v_r\cos\left(\theta + \theta'\right), \tag{2.21}$$

where v_r is the main signal's radial velocity, which is the one recorded by the radar. Solving equation 2.19 for a, and taking only the positive solution, gives

$$a = \frac{4d\sin\theta + \sqrt{(4d\sin\theta)^2 - 16(d^2 - d'^2)}}{8}. \tag{2.22}$$

Similarly, solving equation 2.20 for a gives

$$a = \frac{d}{2}(\cos\theta\tan\theta' + \sin\theta). \tag{2.23}$$

A final inference for a can be achieved by inserting equation 2.20 into equation 2.21 and solving for a:

$$a = \frac{d}{2}\Big[\cos\theta \tan\left(\arccos\left(\frac{v_r'}{v_r}\right) - \theta\right) + \sin\theta\Big]. \tag{2.24}$$

Similar geometric equations can be found for the cases direct/indirect, indirect/direct. Instead of $d_{ghost} = d' = (d' + d')/2$, the ghost will then appear to be at a distance of $d_{ghost} = (d + d')/2 < d'$. If $\theta = 0$, the equivalent of equation 2.22 becomes

$$a = \sqrt{d_{ghost}^2 - d_{ghost} \cdot d}. \tag{2.25}$$

This means that if the indirect/direct case is assumed while indirect/indirect is actually the case, the distance to the wall will be overestimated.

3. Method

The data analyzed in this thesis was recorded for this purpose only in non-laboratory environments. To this end, commercial antennas were used under a specific configuration. If not otherwise specified, the main target was a corner reflector held by a person walking at a comfortable walking pace. All recordings were postprocessed in python.

3.1 Sensors

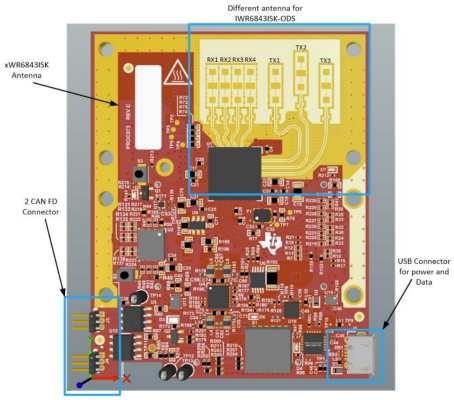


Figure 3.1: TI IWR6843ISK radar sensor, front view, adapted from [14].

A technical drawing of the Texas Instruments IWR6843-ISK sensor evaluation kit, which was used throughout the thesis, can be seen in figure 3.1, with its antennas in the top right corner. Their polarisation is such that the electric field is in the vertical plane and the magnetic field in the horizontal plane. It has a 60 to 64 GHz range and of its 3 transmit antennas, only 2 were used, whereas all of its 4 receive antennas were used. The parameters of the configuration configuration are found in table 3.1. The bandwidth can be computed by multiplying the ramp time with the slope such that $B = 2340\,\mathrm{MHz}$. From equation 2.7 it follows that $d_{max} \approx 40\,\mathrm{m}$, such that each range bin represents $40\,\mathrm{m}/128 \approx 30\,\mathrm{cm}$. Because two TX were used, the chirp to chirp time for each antenna is $T_c = 2 \cdot (43 + 78)\,\mu\mathrm{s}$. From equation 2.9 $v_{max} \approx 5.14\,\mathrm{m/s}$ and then each velocity bin represents $5.14/64\,\mathrm{m/s} \approx 0.08\,\mathrm{m/s}$, since 64 bins are allocated to positive velocity and 64 bins to negative velocity.

3.2 Measurement Set Up

If not otherwise specified, the radar sensor was placed on a tripod, $1.5\,\mathrm{m}$ above the ground, and a person was recorded walking away from it. This person would hold a corner reflector at hip height, $\sim 1\,\mathrm{m}$, pointing towards the sensor. Because of how the radio waves reflect on the reflector's surfaces (see figure 3.2), their direction is exactly reversed. This ensures that the dominant multipath is always indirect/indirect.

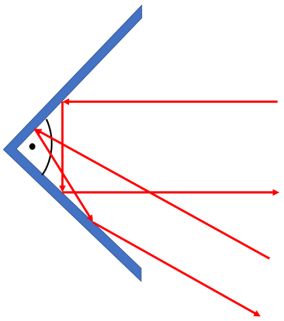


Figure 3.2: Working principle of a corner reflector.

Some of the environments in which the radar sensor is placed are

Table 3.1: Specifications of the chirp configuration.

Parameter	Value
start frequency f_c	60 GHz
idle time T_i	43 µs
ramp time	78 µs
frequency slope S	30 MHz/µs
ADC samples N	128
ADC sampling frequency F_s	8000 ksps

shown in figure 3.3a-c: a parking lot, a rather open indoors space with glass surfaces on one side, and 2.5 m next to an office building. In the latter case, the wall is 3.2 m from the fence to its left. The most basic scenario involves the reflector being carried away in a straight line at almost zero AoA. This, together with two other tracks, is sketched in figure 3.3d.

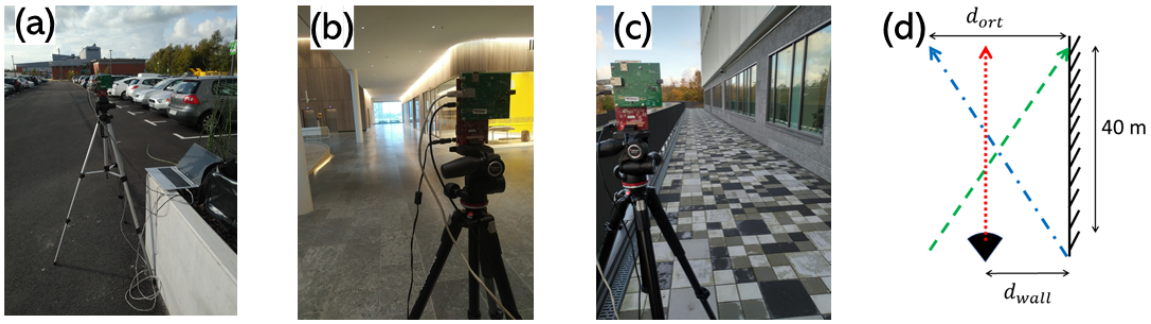


Figure 3.3: (a)-(c) Different environments in which the radar is placed. (d) Three different target tracks next to a wall.

3.3 Data Analysis

The data recorded by the sensor is stored in a three-dimensional matrix ($128 \times 128 \times 8$): each of the 128 chirps' data fills one column of length 128; the depth stems from the 2×4 antenna combinations. A Fourier transform over its first axis gives the range, a Fourier transform over its second axis gives the velocity, and a Fourier transform over its last axis gives the angle. In Python, the first and second Fourier transform can be combined with the numpy.fft.fft2 command that is part of the NumPy library. In preparation to this, a two-dimensional Hanning window is applied. This leads to eight Range-Doppler plots at each time step. These can be averaged to a single plot for visualization. It is common practice to display the plots such that zero velocity is found in the center. Furthermore, the Range-Doppler plots are converted to the decibel scale. How the result would look like in practice is shown in the result section, figure 4.2, and it might be useful to skip ahead for clarification.

Moving objects will clearly appear in such a Range-Doppler plot, and if only one object is

present, the highest peak in the plot can directly be associated with it. Similarly, the second highest peak can be associated with its ghost. To avoid false detection, the ghost must not move faster than the main IO, nor must it appear nearer. If there is only a wall on the right (left), its AoA is restricted to be positive (negative).

The angle of arrival of any point in the Range-Doppler plot is easily found by doing a Fourier transform along the third axis. Of course, this is only meaningful if an IO has been detected in the Range-Doppler plot to begin with. Because the third axis has a depth of only eight bins, the third Fourier transform is zero padded to include $n = 128$ bins in total, which is a parameter in the NumPy discrete Fourier transform tool.

The Python code used for the analysis is attached in appendix A. While it cannot be executed without the relevant files and packages, it might be helpful to see how the discrete Fourier transforms are implemented in practice.

4. Results

Results are presented for the simulation of the specular ground reflection. Then a typical Range-Doppler plot is presented from which the signal power can be extracted. This is plotted as a function of range to confirm that the power law from the simulation also holds in practice. Next to the distance to IO and ghost target, the extracted features include velocity and AoA. From these, the wall's position is predicted. Combining a set of outdoor measurements enables a particular robust method.

4.1 Power Law Simulation

As mentioned in subsection 2.2.5, a reflection in the ground will eventually result in the power decreasing even more steeply as a function of distance. When this occurs, for the specific frequency and radar height, can most easily be determined by plotting the square of equation 2.18. This is done in figure 4.1, showing that the tipping point for the d^{-8} behaviour only occurs after hundreds of meters. Since the short-range radars used for this thesis only see up to $\sim 40\,\mathrm{m}$, this effect should not be of concern. That the d^{-4} law actually holds in practice will be shown shortly. The effects of fast fading can also be seen in the figure: the interference results in large deviations from the LOS signal signal (black line) even at short distances. While constructive interference increases the signal power at some distances, destructive interference suppresses the signal at other distances. This is relevant in practice, as will be shown later on.

4.2 Range Doppler Plot

The methods described in subsection 3.3 lead to Range-Doppler plots as seen in figure 4.2. Everything to the right of the center are objects moving away from the radar. If the highest intensity peak shall be associated with the main IO, the clutter in the center must first be ignored a priori. Similarly, to associate the second highest peak with the ghost, a small region around the main peak must be ignored deliberately. And, as described in subsection 3.3, the ghost must not be allowed to come from regions of higher velocity and closer proximity than the main peak. As can be seen from the figure, such regions are not empty but occupied by signals that could stem from, for example, arms and legs swinging forward.

If a video is created from successive images stitched together, it can be clearly seen that the ghost signal approaches the main signal (as long as we observe an IO moving away from the radar). Also, the intensity of the peaks drops with their range. Since the peak intensity is directly proportional to the received power, it is checked next if this decrease can be described by a power law.

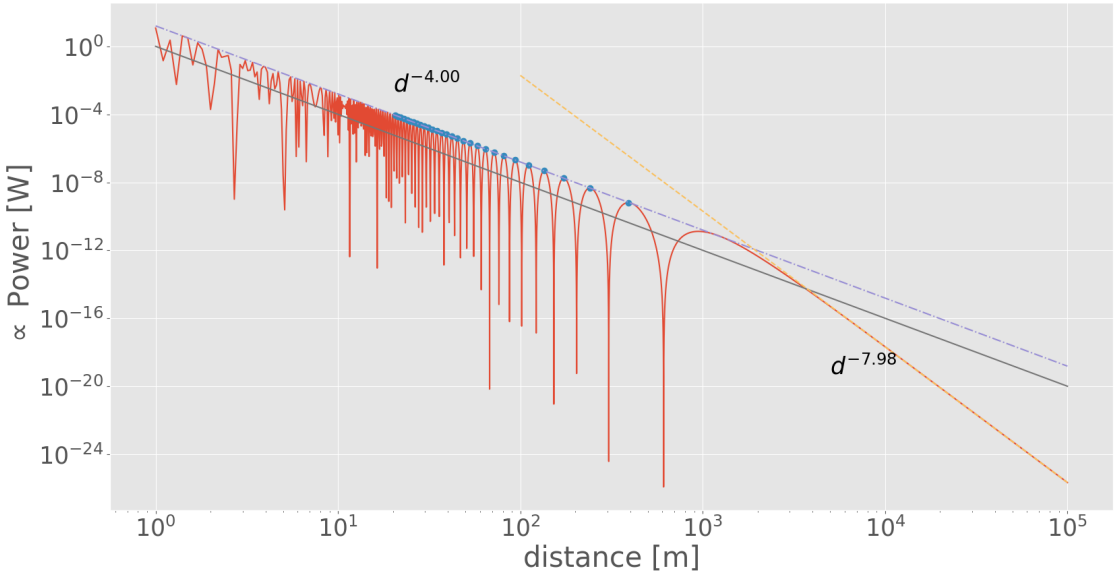


Figure 4.1: Plot of the square of equation 2.18 for $f_c = 61\,\mathrm{GHz}$, $h_{TX} = 1.5\,\mathrm{m}$, and $h_{IO} = 1.0\,\mathrm{m}$. Fits to the two different regions show an initial d^{-4} behaviour followed by d^{-8}. The blue linear fit is taken through the marked local maxima and the yellow linear fit is taken only through the data with $d > 10^4$. The black straight line is the result if the ground reflection is not taken into account.

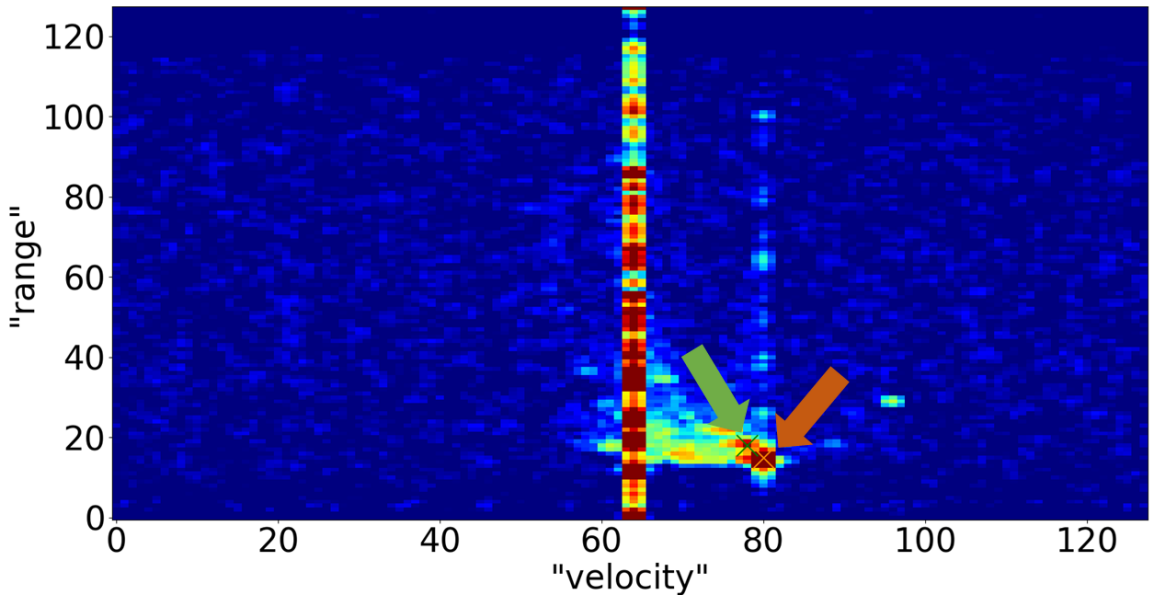
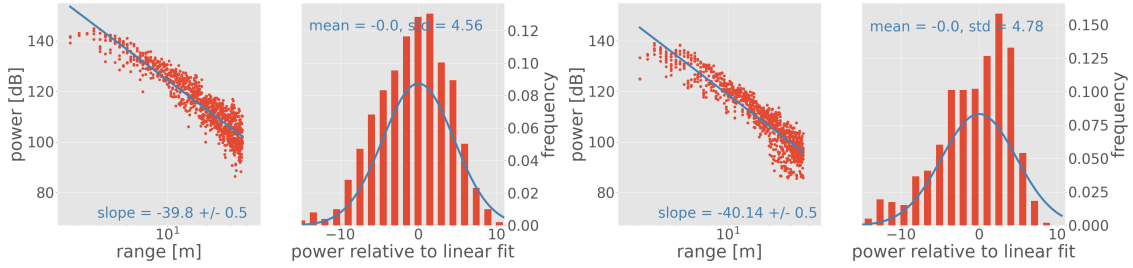


Figure 4.2: A typical Range-Doppler plot where the strongest and second to strongest signals are marked with crosses and arrows in orange and green.

4.3 Power Range Behaviour

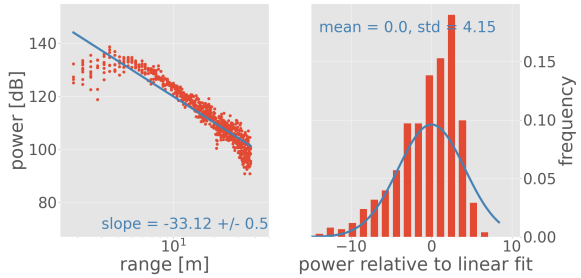
Because of the direct proportionality, the value of the highest peak in a Range-Doppler plot can be treated as the received power. It is an easy feat to obtain the distance at which this peak appears, which allows the power to be plotted as a function of distance. The result for three different environments can be seen in figure 4.3. A linear fit for the first two scenarios, a narrow corridor indoors, and an open air parking lot, reveals a slope of ~ -40. Since these are semi-log plots, this quantity is equivalent to the d^{-4} power law expected for point scatterers. If, however, the sensor is placed outdoors next to an office wall (figure 3.3c), a linear fit is no longer as suitable. Such a fit would give a slope of ~ -33, which is due the decreased power especially at short distances ($< 10\,\mathrm{m}$). Although excluding these points leads again to a d^{-4} behaviour. Hence, these results confirm what was to be expected from the power law simulation.

The same experiments were also performed with $h_{TX} = 2\,\mathrm{m}$, and $2.5\,\mathrm{m}$; results not shown here. In these cases, the linear fit becomes less and less suitable and gives slopes of ~ -16 and ~ -5 if applied. The reason for this is that the antenna directivity is not isotropic, especially not in the elevation angle. In light of these results, the antenna was never set higher than $1.5\,\mathrm{m}$ in subsequent measurements.



(a) A narrow corridor indoors.

(b) Outdoor, on a parking lot.



(c) Outdoor, next to a wall with windows.

Figure 4.3: Power vs range plots (semi-log), their linear fits and how the data is distributed around these fits (together with Gaussian fits of the histograms).

4.4 Geometric Inference with Apriori Model

The results for this section are found in figures 4.4 to 4.9. They concern measurements in an open indoor space (as seen in figure 3.3b), and different tracks next to an office building. Since

they are all similar to each other, it suffices to explain them in detail for the indoor case and then the outdoor cases can be interpreted accordingly.

The two relevant measurements in figure 4.4 are the AoA of the strongest and second strongest signals. They are plotted in green and blue for different distances of the IO. Since the IO was simply moving straight away from the sensor, its AoA is close to zero for all d. A first conclusion from this is that it is appropriate to consider the strongest signal in the Range-Doppler plot to be the LOS response of the corner reflector. The second strongest signal, however, is not necessarily due to the ghost, which is expected to follow the red line in the figure. This red line is simply the realization of equation 2.20, which is the expected θ' given θ, d, and $a = 2.5\,\mathrm{m}$. For up to $d = 16\,\mathrm{m}$ the measured θ' is not at all close to this expectation, which is because there is no wall to the right for the first $8\,\mathrm{m}$. After this, the ghost's AoA does, however, behave as expected, with occasional deviations. In a first step, these deviations, together with the data from the first $16\,\mathrm{m}$ can be discarded by setting a threshold of how far the measured θ' may differ from its expected value. The result is then the "noise reduced" line in figure 4.4.

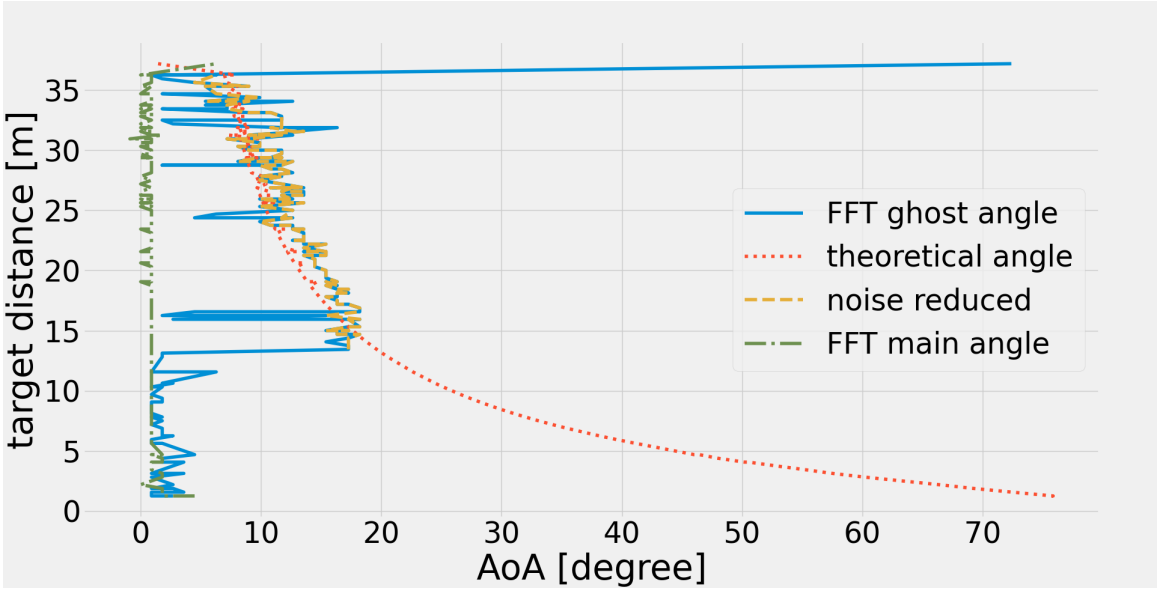


Figure 4.4: Data taken in an indoor environment, the target moving orthogonally away from the sensor. The plot shows the angle of arrival of the main and ghost target, the latter also with its theoretical angle to identify erroneous classification. If the measurement is in the vicinity of the red graph, it is marked in yellow and considered noise reduced.

Once ghost signals are identified, however, their distance and angle information can be used to plot the ghost's position over time in Cartesian coordinates. The same can be done for the main signal's position, and the result is presented in figure 4.5. While both positions follow a path that is roughly as theorized, especially the ghost signal is unstable in the horizontal direction. The variance in the main signal's horizontal position after $20\,\mathrm{m}$ can be explained by the drop in SNR with increasing distance but increasing the angle resolution would make these jumps less significant. One way to do so would be by padding the signal with more zeros in the angle-FFT.

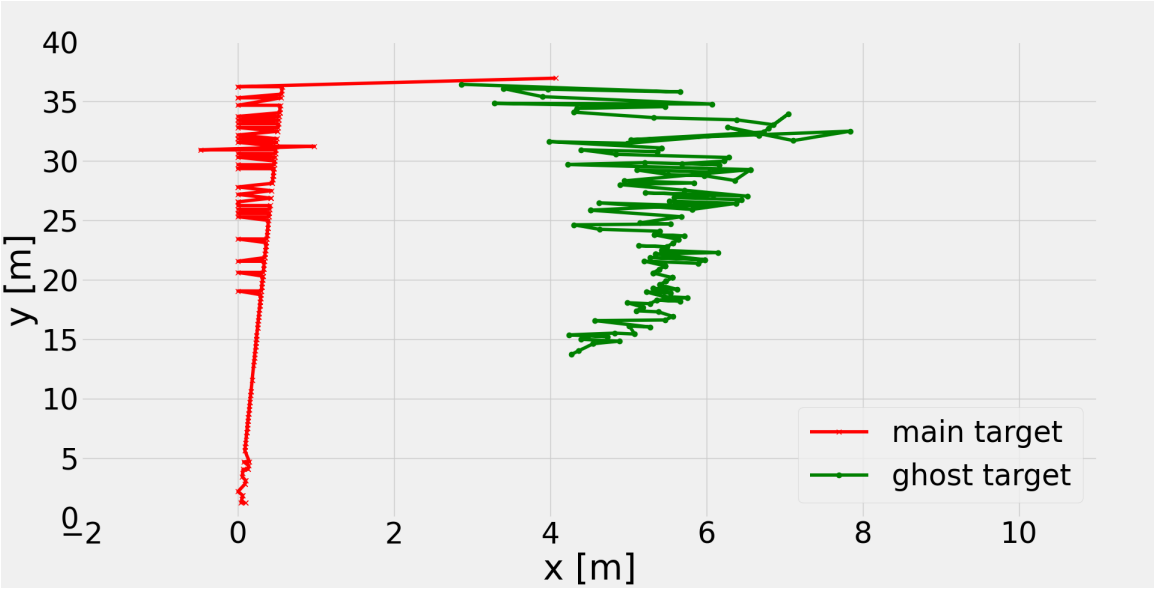


Figure 4.5: The Cartesian coordinates of main and ghost target in an indoor environment. For the first $8\,\mathrm{m}$ there is no wall to the right of the radar.

As shown in subsection 2.2.6, there are three different equations to infer a from the measurements, and their differing results can be seen in figure 4.6. All equations require θ as an input, but the only other variable needed in equation 2.22 is d. As can be seen, it systematically overestimates the distance to the wall and this estimate increases with the distance to the main signal. This increase is shared among all three methods and is linear in the case equation 2.24 is used. The latter makes use of the velocity of the ghost signal. Lastly, equation 2.23 requires the measurement of θ'. It is not clear why these methods differ so much and in the indoor scenario it is difficult to decide on an optimal method. For the outdoor case, this will become more obvious.

Outdoors, next to an office building, the tracks covered by the IO are as indicated in figure 3.3d. Repeating the whole analysis as in the indoor case, the inferred distances a to the building's wall are much closer to $2.5\,\mathrm{m}$ throughout the whole range of d in figures 4.7c, 4.8c, and 4.9c if equation 2.23 is used. It is from this observation that we conclude that this equation is the one best suited for estimating a. There is still a visible slope and variance in the estimate, but it is good enough for basic applications, which are discussed in the conclusion. Moreover, the variance can be reduced if several measurements are averaged, as will become apparent in the following section.

4.5 Cumulative Geometric Inference

Throughout this thesis ghosts are assumed to be the second strongest signal. In the previous section false ghost detections are identified by how far they are from an expected position. This cleaning step requires that a is known beforehand, but there is an alternative that combines the data from several measurements, which is presented in the following.

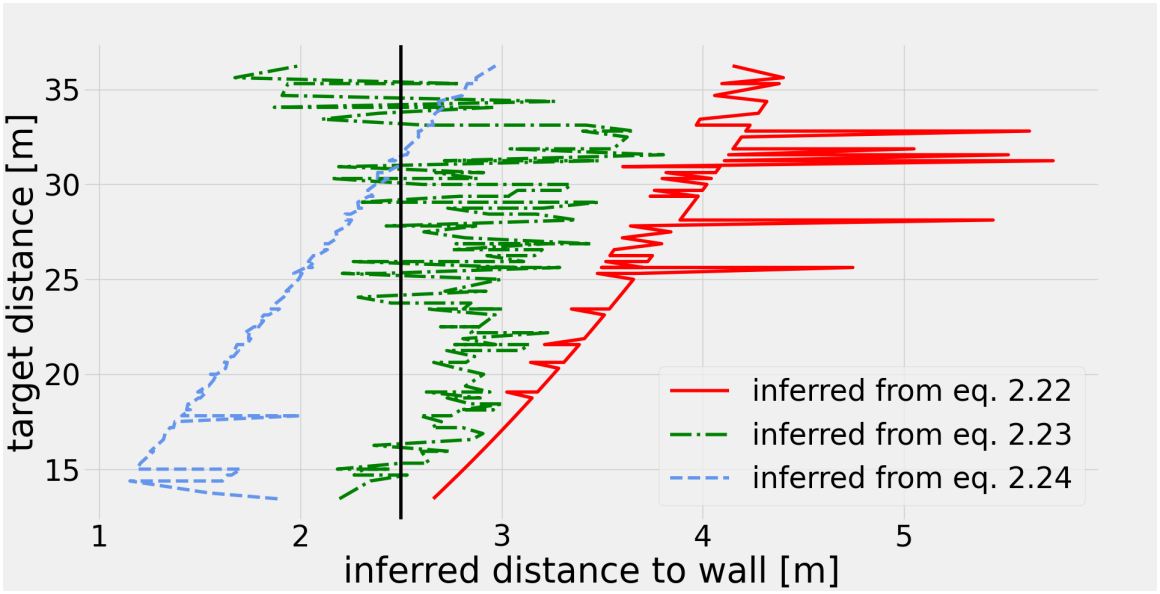
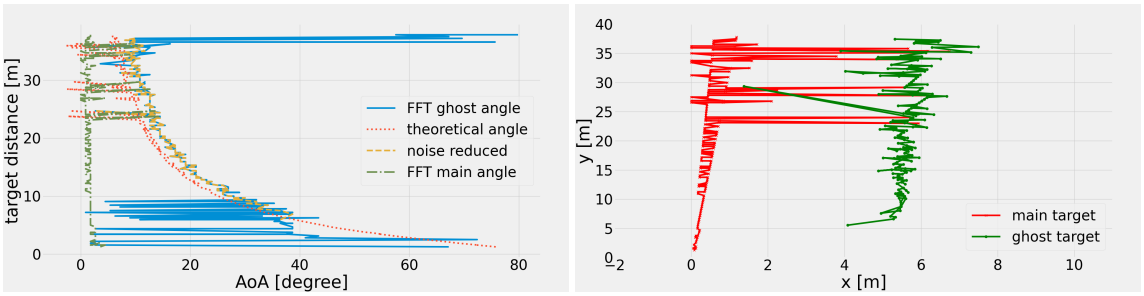
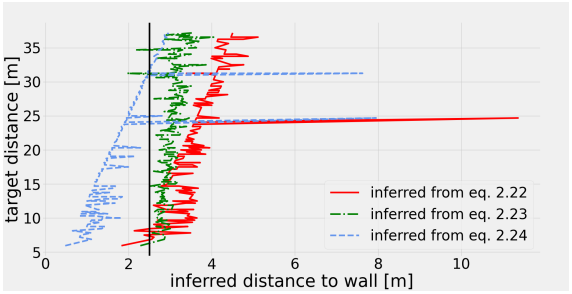


Figure 4.6: The inferred distance to an indoors wall using three different methods (see main text). In reality the wall is 2.5 m away, marked by the black horizontal line.



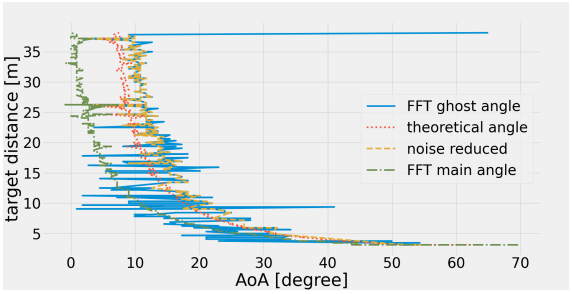
(a) The angle of arrival of the main and ghost target (the latter also with its theoretical angle to discern erroneous classification).

(b) The Cartesian coordinates of both targets.

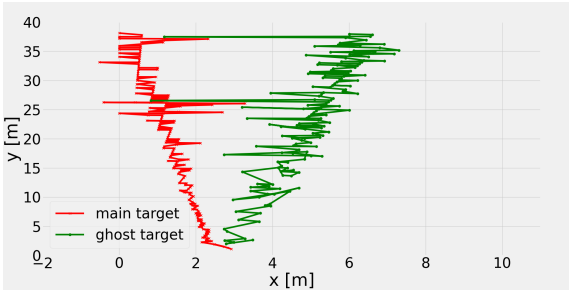


(c) The inferred distance to the wall (in reality 2.5 m away, black horizontal line) using three different methods (see main text).

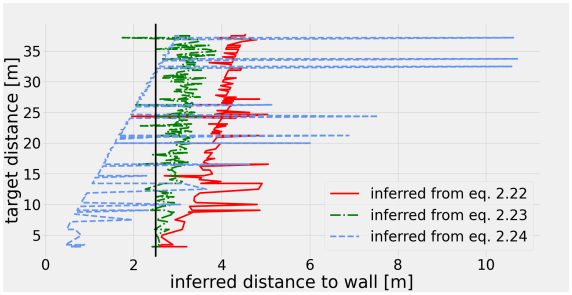
Figure 4.7: Analysis of data taken outdoors next to an office building, the target moving orthogonally away from the sensor.



(a) The angle of arrival of the main and ghost target (the latter also with its theoretical angle to discern erroneous classification).

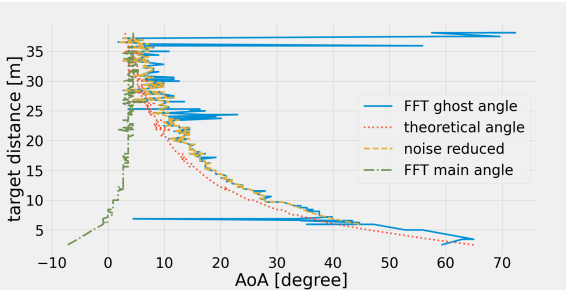


(b) The Cartesian coordinates of both targets.

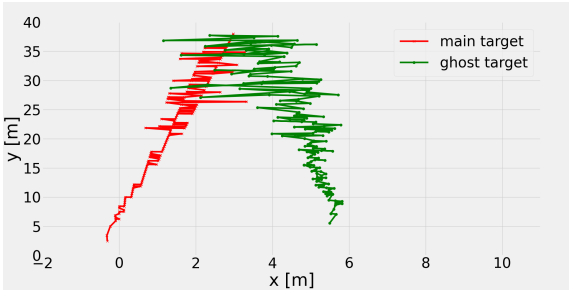


(c) The inferred distance to the wall (in reality 2.5 m away, black horizontal line) using three different methods (see main text).

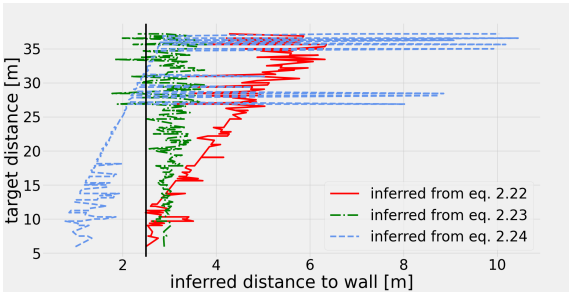
Figure 4.8: Analysis of data taken outdoors next to an office building, the target moving diagonally away from the sensor (from right to left).



(a) The angle of arrival of the main and ghost target (the latter also with its theoretical angle to discern erroneous classification).



(b) The Cartesian coordinates of both targets.



(c) The inferred distance to the wall (in reality 2.5 m away, black horizontal line) using three different methods (see main text).

Figure 4.9: Analysis of data taken outdoors next to an office building, the target moving diagonally, from left to right, away from the sensor.

Since all measurements were performed three times, there are nine available data sets that each predict a in the outdoor case. For each set, a list with pairs of $d \cdot \cos\theta$ and a can be created and these lists can then be combined. This combined list can be plotted as a two-dimensional heat map, which can be seen in figure 4.10. Because the outliers are not the same for each measurement, but the correct inferences are, the outliers will be negligible in such a representation. This is exactly what can be observed in figure 4.10.

In particular, the heat map has a horizontal resolution of $5\,\mathrm{cm}$ and a vertical resolution of $40\,\mathrm{cm}$. A lower resolution is possible, but then the inferred wall would be less visible because each "correct" bin might then only contain two or three counts instead of one count for the bins containing an outlier.

To make the wall even more visible, the heat-map can be smoothed by applying a Gaussian filter, which can be seen in the right of figure 4.10. From there, it can be seen that the estimate of a changes almost by half a meter along the vertical. Considering the original heat map on the left reveals some kind of structure in the inference. It seems as if the wall is interrupted at regular intervals. This could be due to patterns in the wall such as glass alternating with concrete, which should result in a more large scale structure. Thus, it is more likely that fast fading is responsible for the valleys and peaks in this representation.

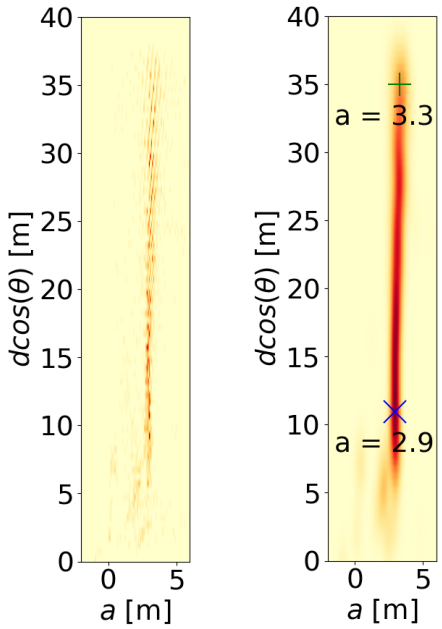


Figure 4.10: The inferred distance to the wall (a) at different vertical distances to the radar. To remove outliers, the results of nine different measurements are overlaid as a heat map (see main text). The heat map on the right is simply a blurred version of the one to the left. Two exemplary inferred distances are marked to visualize the slope.

5. Conclusion

When radar sensors are placed next to a wall the radio waves reflect from this surface and ghost targets appear. This thesis succeeded in showing that this permits to infer the geometry of a simple environment. More precisely, when there is only one wall, a single object moving in different ways in front of the radar gives rise to a main signal. The second most intense signal can then be associated with the ghost target. There are three different trigonometric formulae that can be used to infer the distance to the wall and one of them was shown to result in a satisfying inference. With this one, the distance to the wall was systematically overestimated and the wall was also inferred to not be straight. The following paragraph explains why these minor insufficiencies are, however, not of great concern.

Once the position of the wall is known, an obvious next step to mitigate multipath would be to discriminate all signals that seem to originate from behind the wall. Even if the wall's position is not inferred exactly correct, it is good enough to mitigate most of the possible ghost signals. Even more helpful would be to translate the ghost targets back to their actual sources' position. This could be useful if the main target alone reflects too weakly to be detected, as could be the case for small animals or drones. For this application, it is even less important that the environment is predicted correctly. Indeed, it would be worse to provide the radar with a real floor plan because the systematic shift would then be incorporated into the translation to the main target. As long as the ghost can be correctly associated with its real source, it is irrelevant if the wall's position is predicted correctly.

In practice, this would mean that the environment's features would have to be determined at installation. In this step, it is important that only one IO moves in front of the radar because otherwise the second strongest signal would most definitely not be due to a ghost target. The big advantage is that the floor plan could be relearned at regular intervals. If new furniture is moved in or parts of a building are remodeled, the radar could simply adapt to its new environment.

The most successful inference was achieved for outdoor environments with a continuous wall on one side of the radar, but especially for indoor environments with high ceilings and regular walls, the results should be applicable there, too. And even for more dense environments such as corridors, the geometric methods should work. In this case, the ghosts can appear on both sides of the radar, such that the algorithm would only have to be rewritten slightly. Thus, this thesis takes indoor radar systems one step closer.

Bibliography

[1] Simon Watts. *Airborne Maritime Surveillance Radar, Volume 1*. 2053-2571. Morgan & Claypool Publishers, 2018.

[2] Salvo D'Agostino. *Hertz's Experiments on Electromagnetic Waves*, pages 135–166. Springer Netherlands, 01 2000.

[3] Michael Parker. Chapter 20 - automotive radar. In *Digital Signal Processing 101 (Second Edition)*, pages 253 – 276. Newnes, 2017.

[4] C.J. Baker and B.D. Trimmer. Short-range surveillance radar system. *Electronics & Communication Engineering Journal*, 12:181 – 191, 09 2000.

[5] Erik Leitinger. *Cognitive Indoor Positioning and Tracking using Multipath Channel Information*. PhD thesis, 01 2016.

[6] F. Roos, M. Sadeghi, J. Bechter, N. Appenrodt, J. Dickmann, and C. Waldschmidt. Ghost target identification by analysis of the doppler distribution in automotive scenarios. In *2017 18th International Radar Symposium (IRS)*, pages 1–9, 2017.

[7] John David Jackson. *Classical electrodynamics*. Wiley, 3rd ed. edition, 1999.

[8] Cesar Iovescu and Sandeep Rao. *The fundamentals of millimeter wave sensors*. Texas Instruments Incorporated, 2020.

[9] M. A. Jensen and J. W. Wallace. A review of antennas and propagation for mimo wireless communications. *IEEE Transactions on Antennas and Propagation*, 52(11):2810–2824, 2004.

[10] M.A. Richards. *Fundamentals Of Radar Signal Processing*. McGraw-Hill Education, 2005.

[11] Andreas F. Molisch. *Wireless Communications*. Wiley Publishing, 2nd edition, 2011.

[12] Ronald Driggers Craig Hoffman, editor. *Encyclopedia of Optical and Photonic Engineering*. CRC Press, 2015.

[13] F. Roos, M. Sadeghi, J. Bechter, N. Appenrodt, J. Dickmann, and C. Waldschmidt. Ghost target identification by analysis of the doppler distribution in automotive scenarios. In *2017 18th International Radar Symposium (IRS)*, pages 1–9, 2017.

[14] Chethan Kumar Y.B. *User's Guide 60 GHz mmWave Sensor EVMs*. Texas Instruments Incorporated, 2020.

Appendices

A. Python Code

```
import matplotlib.pyplot as plt
import numpy as np
import csv
import math
import matplotlib.cm as cm
from mpmath import *
from scipy.stats import norm
from collections import Counter

plt.style.use('fivethirtyeight')
plt.rcParams.update({'font.size': 35})

from data_loaders import ReadPickle

from radar_rec_utils.utils.unpack_cube import unpack_cube

frame = 80
cl_width = 2
wall_distance = 2.5
factor = 0.3125

def rd_process(cube):
    ns = cube.shape[0]
    nr = cube.shape[1]
    win = np.outer(np.hanning(ns), np.hanning(nr))

    fft_size = (ns, nr)
    rd = np.fft.fft2(cube * win[:, :, np.newaxis], s=fft_size, axes=(0, 1))
    rd = np.fft.fftshift(rd, axes=(1, ))

    rd_db = 20 * np.log10(np.mean(np.abs(rd), axis=2))
    nf = np.median(rd)
    nf_db = np.median(rd_db)

    return rd, rd_db, nf, nf_db

def angle_process(rd, rd_db, peak_idx):
    ang_fft = np.fft.fft(rd[peak_idx], n = 128)
    ang_fft = np.fft.fftshift(ang_fft)
    angle_peak_idx = np.argmax(np.abs(ang_fft))

    angle = 2 * (angle_peak_idx - ang_fft.shape[0] // 2) / (ang_fft.shape[0])
    angle = np.rad2deg(np.arcsin(angle))

    return ang_fft, angle

def find_peak(range_velocity, clutter_width, factor = 0.3125):
```

```
    nv = range_velocity.shape[1]
    rv_mod = range_velocity.copy()
    rv_mod[:,nv//2-clutter_width:nv//2+clutter_width] = 0 #Set the
    clutter to zero
    idx = np.unravel_index(rv_mod.argmax(), rv_mod.shape)
    power = rv_mod[idx]
    distance = idx[0]*factor
    return rv_mod, power, distance, idx

def find_second_peak(range_velocity, idx, factor = 0.3125):
    rv_mod = range_velocity.copy()
    rv_mod[idx[0]-2:idx[0]+3,idx[1]-3:idx[1]+4] = 0 #Set around main
    peak to zero
    rv_mod[:idx[0]+1,:] = 0 #Set shorter distances to 0
    rv_mod[:,idx[1]:] = 0 #Set higher velocities to 0
    idx_2 = np.unravel_index(rv_mod.argmax(), rv_mod.shape)
    distance_second = idx_2[0] * factor
    power_2 = rv_mod[idx_2]
    return idx_2, distance_second, power_2

def inverse_wall_non_radial(d_direct, v_real, d_ghost, v_ghost,
    theta_ghost, theta = 0, factor = 0.3125):
    theta_rad = np.deg2rad(theta)
    theta_ghost_rad = np.deg2rad(theta_ghost)
    b = 4*d_direct*math.sin(theta_rad) # b in quadratic equation
    d_wall_d = factor/8 * (b + math.sqrt(b**2-16*(d_direct**2-d_ghost
    **2)))
    arg = math.acos(v_ghost/v_real) - theta_rad
    d_wall_v = factor/2 * d_direct * (math.cos(theta_rad)*math.tan(arg)+
    math.sin(theta_rad))
    d_wall_angle = factor/2 * d_direct * (math.cos(theta_rad)*math.tan(
    theta_ghost_rad) + math.sin(theta_rad))
    return d_wall_d, d_wall_v, d_wall_angle

def find_ghost_AoA_non_radial(d, d_wall = 2.5, theta = 0 ):
    arg = (2*d_wall-d*math.sin(np.deg2rad(theta)))/(d * math.cos(np.
    deg2rad(theta)))
    theta_ghost = np.rad2deg(math.atan(arg))
    return theta_ghost


if __name__ == '__main__':
    folder = "/home/andrenu/radar_data/2020_11_25/_raw_data_2020-11-25
    _10-53-24"
    reader = ReadPickle(folder)

    data = reader.read_frame(frame)
    cube = unpack_cube(data)
    cube_med = np.median(np.abs(cube.real))

    rd, rd_db, nf, nf_db = rd_process(cube)

    rv_m, power, distance, idx = find_peak(rd_db, cl_width, factor)
    idx_2, distance_ghost, power_2 = find_second_peak(rv_m, idx, factor)
```

```
94
95      plt.figure()
96      plt.imshow(rd_db, vmin=nf_db, vmax=nf_db + 40, cmap='jet', aspect='
    auto', origin='lower')
97      plt.scatter(idx[1], idx[0], s=500, marker='x', c = 'darkorange')
98      plt.scatter(idx_2[1], idx_2[0], s=500, c='green', marker='x')
99      plt.xlabel('"velocity"')
100     plt.ylabel('"range"')
101
102     ang_fft, angle = angle_process(rd, rd_db, idx)
103
104     plt.figure()
105     plt.plot(20 * np.log10(np.abs(ang_fft)))
106     plt.ylabel('dB')
107     plt.xlabel('Anglebin')
108     plt.title(f'AngleFFT (angle={angle})')
109
110     power_list = []
111     power_2_list = []
112     distance_list = []
113     d_wall_d_list = []
114     d_wall_v_list = []
115     d_wall_angle_list = []
116     d_wall_av_list = []
117     d_wall_1_list = []
118     d_wall_2_list = []
119     angle_list = []
120     theta_list = []
121     angle_direct_list = []
122     distance_ghost_list = []
123     error1_list = []
124     error2_list = []
125     d_wall_3_list = []
126
127
128     for ind, frame in reader.read_frames():
129         power_sum = []
130         cube = unpack_cube(frame)
131
132         rd, rd_db, nf, nf_db = rd_process(cube)
133         rv_m, power_db, distance, idx = find_peak(rd_db, cl_width,
    factor)
134         idx_2, distance_ghost, power_2 = find_second_peak(rv_m, idx,
    factor)
135
136         ang_fft_direct, angle_direct = angle_process(rd, rd_db, idx)
137         ang_fft, angle = angle_process(rd, rd_db, idx_2)
138         d_wall_d, d_wall_v, d_wall_angle = inverse_wall_non_radial(idx
    [0], idx[1], idx_2[0], idx_2[1], angle, angle_direct, factor)
139         theta_ghost = find_ghost_AoA_non_radial(distance, wall_distance,
     angle_direct)
140
141         if power_db > nf_db + 20 and ind < 400:
142             g_rel = ghost_rel_list[d_list_bin.index(idx[0])]
```

```
            v_rel = v_rel_list[d_list_bin.index(idx[0])]
            plt.figure(3)
            plt.clf()

            plt.subplot(121)
            plt.imshow(np.abs(cube[:, :, 0].real), vmin=cube_med, vmax=
    cube_med * 4, cmap='jet', aspect='auto')
            plt.xlabel('chirp')
            plt.ylabel('discrete samples')
            plt.title('Baseband')

            plt.subplot(122)
            plt.imshow(rd_db, vmin=nf_db, vmax=nf_db + 40, cmap='jet',
    aspect='auto')
            plt.scatter(idx[1], idx[0], s=500, c='darkorange', marker='x
    ')
            plt.scatter(idx[1]*v_rel, idx[0]*g_rel, s=500, c='crimson',
    marker='x')
            plt.scatter(idx_2[1], idx_2[0], s=500, c='green', marker='x'
    )
            plt.xlabel('"velocity"')
            plt.ylabel('"range"')
            plt.title('No repair (noise = {:.2f}dB)'.format(nf_db))

            plt.suptitle(f'Frame {ind}, {idx}')

            plt.pause(0.01)


        #disregard noise ("ind" manually chosen)
        if power_db > nf_db + 20 and ind < 360 and ind > 50 and angle >
    0:
            power_list.append(power_db)
            power_2_list.append(power_2)
            distance_list.append(distance)
            d_wall_d_list.append(d_wall_d)
            d_wall_v_list.append(d_wall_v)
            d_wall_angle_list.append(d_wall_angle)
            angle_list.append(angle)
            theta_list.append(theta_ghost)
            angle_direct_list.append(angle_direct)
            distance_ghost_list.append(distance_ghost)



    plt.figure()#Power-Range for main peak
    plt.plot(distance_list, power_list, '-o')
    plt.xlabel('range [m]')
    plt.ylabel('power [dB]')
    log_distance_list = np.log10(distance_list)
    z = np.polyfit(log_distance_list, power_list, 1)
    x = np.linspace(min(log_distance_list), max(log_distance_list),
    1000)
    plt.plot(10**x, z[0]*x + z[1])
```

```
    plt.xscale("log")
    print(z)
    plt.text(10, 110, f'slope = {np.round(z[0], 2)}', fontsize=15)

    #discern between useful data and noise:
    angle_list_reduced = []
    distance_list_reduced = []
    d_wall_d_list_reduced, d_wall_v_list_reduced, d_wall_angle_list_reduced, distance_ghost_list_reduced = [], [], [], []
    power_2_list_reduced = []
    for i in range(len(distance_list)):
        if angle_list[i] > theta_list[i] - 3 and angle_list[i] < theta_list[i] + 5: #these parameters are manually decided
            angle_list_reduced.append(angle_list[i])
            distance_list_reduced.append(distance_list[i])
            d_wall_d_list_reduced.append(d_wall_d_list[i])
            d_wall_v_list_reduced.append(d_wall_v_list[i])
            d_wall_angle_list_reduced.append(d_wall_angle_list[i])
            distance_ghost_list_reduced.append(distance_ghost_list[i])
            power_2_list_reduced.append(power_2_list[i])

    plt.figure() #Plot AoAs
    plt.plot(angle_list, distance_list, label = 'FFT ghost angle')
    plt.plot(theta_list, distance_list, linestyle = ':', label = 'theoretical angle')
    plt.xlabel('AoA [degree]')
    plt.ylabel('target distance [m]')
    plt.plot(angle_list_reduced, distance_list_reduced, linestyle = '--', label = 'noise reduced')
    plt.plot(angle_direct_list, distance_list, linestyle = '-.', label = 'FFT main angle')
    plt.legend()

    plt.figure()#Wall inference
    plt.plot(d_wall_d_list_reduced, distance_list_reduced, c = 'r', label = 'inferred from eq. 2.22')
    plt.plot(d_wall_angle_list_reduced, distance_list_reduced, c = 'g', linestyle = '-.', label = 'inferred from eq. 2.23')
    plt.plot(d_wall_v_list_reduced, distance_list_reduced, c = 'cornflowerblue', linestyle = '--', label = 'inferred from eq. 2.24')
    plt.axvline(x = 2.5, c = 'k')
    plt.xlabel('inferred distance to wall [m]')
    plt.ylabel('target distance [m]')
    plt.legend()

    #Plot in cartesian coordinate system:
    k = 0
    plt.figure()
    plt.ylim(0, 40)
    plt.xlim(-2, 11)
    plt.xlabel('x [m]')
    plt.ylabel('y [m]')
    x_list, y_list, x_ghost_list, y_ghost_list = [], [], [], []
```

```
235     for i in range(len(distance_list)):
236         x_list.append(distance_list[i] * math.sin(np.deg2rad(
    angle_direct_list[i])))
237         y_list.append(distance_list[i] * math.cos(np.deg2rad(
    angle_direct_list[i])))
238         if distance_list[i] == distance_list_reduced[k] and k < len(
    distance_list_reduced) - 1:
239             x_ghost_list.append(distance_ghost_list_reduced[k] * math.
    sin(np.deg2rad(angle_list_reduced[k])))
240             y_ghost_list.append(distance_ghost_list_reduced[k] * math.
    cos(np.deg2rad(angle_list_reduced[k])))
241             k += 1
242     plt.plot(x_list, y_list, '-x', c = 'r', label = 'main target')
243     plt.plot(x_ghost_list, y_ghost_list, '-o', c = 'g', label = 'ghost
    target')
244     plt.legend()
245 
246     plt.show()
```