Modelling of Hand Gestures and Motions

ERIK INGEMARSSON

MASTER´S THESIS

DEPARTMENT OF ELECTRICAL AND INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING | LTH | LUND UNIVERSITY

Modelling of Hand Gestures and Motions

Erik Ingemarsson

Department of Electrical and Information Technology
Lund University

Supervisors: Lars Ohlsson Fhager & Sebastian Heunisch

Examiner: Mats Gustafsson

July 7, 2019

Printed in Sweden
Tryckeriet i E-huset, Lund

Abstract

Measuring the radar response has proven to be a valid input for neural networks used for gesture and motion recognition. Previous work have been based on measuring the radar response of hand gestures, but it has been shown that models based on the analytic expressions for basic objects are useful when simulating larger motions, such as pedestrians. This work explores if a hand model made of basic objects can reach the same results as actual hands. To do this, this basic objects model is implemented in two different solvers to compare the difference in value and run-time between analytic expressions and Finite Element Method solutions. The Finite Element solution is shown to change value based on the simulated gesture, but the analytic model is proving to be far faster. In the end, the latter is not capable of reproducing the response of all motions which have been measured. Implementing object interaction into the analytic model should give this model the details required to simulate even the smaller gestures. Therefore, this model is still work in progress.

Popular Science Summary

The initial use for radar-technology was for long-range airplane detection, but its uses have branched off into a different purpose. This new branch is instead based on short-range detection. One major part of this branch is the use of radar in traffic situations. With the availability of this short-range motion detection, the usability for radar in motion and gesture recognition has been explored. A contact-free, motion-based system control has long been a goal, and previous steps have been efficient at motion capture but inefficient when considering power consumption.

Based on the success of previous work done on radar-based gesture recognition, the following work focuses on the possibility of simulating accurate gesture-response. Previous studies have shown that radar-based gesture recognition is possible by using neural networks. To make these neural networks able to do what they are capable of, training-data is needed. One method for obtaining this training-data is to record individuals performing motions and gestures. This would be strenuous and tedious, which is why simulated data would be of great assistance. What if simulated data could be calculated close enough to actual measurements? To answer this, this work will examine one method of simulating radar response.

The simulations will be based on analytical expressions for the radar cross section (RCS) of three basic objects. One plate, fourteen spheres, and fourteen cylinders are used to represent the structure of the hand, so how well these objects are simulated will first be controlled. This is done by implementing a plate, a sphere, and a cylinder, and then first examining the frequency-response of the formulas for the individual objects. The objects, which will be simulated as made of perfectly electrically conductive (PEC) materials, will then be compared to results from a Physical Optics (PO) approximation. And, lastly, the inherent implementation of a time delay will be examined so that the objects placed at a distance from the radar will be simulated to be at said distance. The important reason as to why a model based on analytic expressions would be preferable to a model based on the physical optics approximation, is that the physical optics model would take far longer time. This will therefore be examined for both the comparison of the simulations for the basic objects, and also in the next step when a hand is simulated.

After the objects have been examined and made sure to have been implemented properly, a model of a hand will then be implemented. This hand will describe

the positions and directions of the objects. This will be done by first defining the position and direction of the palm. The fingers and the thumb will then be implemented by defining the location of the first joint of each in relation to the position and direction of the palm. Then the first phalanx of each finger will be implemented based on the position of the first joint and the direction of said phalanx. In the end, this leads to that the whole hand can be defined by using the location and direction of the palm and the angle of each of the joints as input. The implemented hand will then be examined and compared to physical optics approximation results from two gestures, and, lastly, four gestures will then be compared to measured data.

With this analytical model it will be shown that the response of different gestures can be simulated. However, the way that the objects have been implemented means that the objects do not interact, and the simulated data is therefore not similar enough to be used as training data.

Acknowledgements

I would like to thank my two supervisors, Lars and Sebastian, for their valuable input.
I would also like to thank Hannes Dahlberg and Anton Evertsson for letting me use their measurements as part of my results.

Nomenclature

PEC	Perfect Electrical Conductor
GD	Group Delay
IP	Impulse Response
RCS	Radar Cross Section
RT	Range-Time
RV	Range-Velocity

Table of Contents

1 Introduction 1

2 Theory 3
- 2.1 Radar Basics 3
- 2.2 Electromagnetic Wave Propagation 5
- 2.3 Radar Cross Section 7
- 2.4 Impulse Response and Group Delay 9
- 2.5 Basic Radar Signal Processing 10

3 Method 13
- 3.1 Simulating Hand Gestures 13
- 3.2 Generating Radar Response 14
- 3.3 Implementation of Basic Objects 14
- 3.4 Analytic Model vs. Physical Optics Approximation 14
- 3.5 Simulated Radar Response vs. Measured Data 15

4 Results and Discussions 17
- 4.1 RCS of the Basic Objects 17
- 4.2 Impulse Response of Basic Objects in MATLAB 18
- 4.3 RCS of Basic Hand Gestures 19
- 4.4 Simulations of Motions and Gestures 20
- 4.5 Measured Motions and Gestures 24

5 Conclusion and Outlook 27
- 5.1 Conclusion 27
- 5.2 Outlook 27

References 29

List of Figures

2.1 A general bistatic radar system. 3
2.2 Vectors involved in (2.8). 5
2.3 Geometry of the plate. 8
2.4 Geometry of the cylinder. 9
2.5 Time scales for radar. 10
2.6 Applying a Fourier transform along slow-time values extracts Doppler frequencies of the object. 11

3.1 The steps taken to simulate a Time-Domain signal. 13
3.2 The defined Time-Domain signal, and the corresponding frequency spectrum. The images have been zoomed in on the relevant intervals. 14
3.3 The two gestures simulated as a test. 15
3.4 Motions simulated as part of final result. 16

4.1 RCS comparison of the basic objects. 17
4.2 The impulse response for each of the basic objects, with the objects being placed at a distance of 0.3 m away from the radar. The figure zooms in on the relevant interval and the values are given in dB. . . 18
4.3 The group delay for the transfer function C. 19
4.4 The simulated RCS for the two hand poses for the two simulators. . 19
4.5 The simulated signal data for the open hand moving towards the radar. Distance decreasing from 0.6 m at the start, to 0.3 m at then end. . 21
4.6 A comparison of the transmitted signal in Time-Domain and the received signal of frame 3 in Fig. 4.5a. 21
4.7 The simulated signal data for the open hand moving towards and away from the radar. Distance decreasing from 0.5 m at the start, to 0.3 m in the middle, and then back to 0.5 m. 22
4.8 The simulated signal data for the open hand moving in cross-range at a minimum distance of 0.3 m from the radar. 23
4.9 The simulated signal data for a hand at a distance of 0.45 m from the radar wiggling the index and the middle finger. 23
4.10 The simulated signal data for a hand at a distance of 0.4 m from the radar folding the index finger. 24

4.11 The (a) measured and (b) simulated signal data for an open hand moving towards the radar, and then moving back. Range varying in the interval of 0.3 to 0.5 m. 24
4.12 The (a) measured and (b) simulated signal data for a hand moving cross-range at a distance of roughly 0.45 m. 25
4.13 The (a) measured and (b) simulated signal data for a clenched hand stationary at a distance of about 0.45 m, index- and middle-finger see-sawing. 25
4.14 The (a) measured and (b) simulated signal data for a clenched hand stationary at a distance of about 0.4 m, bending the index finger toward the radar, and then straighten it out. 26

Chapter 1
Introduction

To control any system, one needs to know what the controlling signal looks like. When using a keyboard to control a computer, the signal can easily be measured and controlled. But a more complex step is to use hands waving around in the air in front of some kind of sensor. Having a non-physical way of controlling a system is hardly a new idea, and solutions are present already.

Using a camera to recognize recorded images of different gestures has been proven to work [1, 2, 3], but a drawback is the amount of storage needed for images. An alternative to image recognition is therefore explored, which leads to using radar for gesture recognition.

Previous work has already been done exploring Frequency Modulated Continuous Wave, FMCW, radar using different frequencies, ranging from a few GHz [4, 5, 6, 7], into the tens of GHz [8, 9, 10, 11], and even up into the THz region [12]. This has proven that FMCW-based gesture recognition works, and that it can be implemented.

This work is instead going to focus on using pulsed-radar for gesture recognition, but instead of exploring the frequency-range of a few GHz [13, 14], this work is going to focus on a radar setup using a 60 GHz pulse.

To do this, a model of a hand using basic objects as its building blocks will be simulated. A basic object model could be implemented using the Physical Optics (PO) approximation [15]. But seeing as the PO being a relatively slow process, a model based on analytic expressions is proposed instead [16, 17]. This method has already been explored, simulating the radar response for a walking human. Seeing that it works on a larger scale, the goal is then to see if a similar simulation would be possible for use in hand gesture recognition.

To answer this, a model of a hand will be simulated. The hand will be implemented in two different simulators, using three basic objects as its building blocks. One of the simulators will be the PO solver FEKO and the other will be MATLAB. These two methods of calculating the Radar Cross Section, RCS, will be used to determine if there is any difference in accuracy and run-time between using a FEKO and calculating the analytic expression for the PO. The next step is then to define four different gestures and motions for the model. Using a simulated signal corresponding to a 60 GHz pulsed radar, the radar response for these four motions will be calculated. As a final step, the simulated response will be compared to measured data of the same four gestures.

Chapter 2

Theory

In this chapter the theory necessary to understand how motion, and gesture, recognition might be possible, will be explained. Notice! As the letter i is being used in this report to denote incidence of signal to object, $j = \sqrt{-1}$.

2.1 Radar Basics

Starting off with how power is transmitted through free-space [18, p.44-45][19, p.42-48], let's look at the general structure of a bistatic radar system.

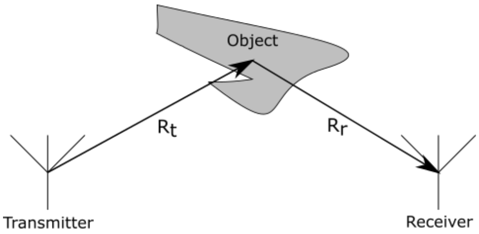


Figure 2.1: A general bistatic radar system.

The first step is to calculate how much power would be received if it was transmitted isotropically from the location of the object. One of the coefficients in this formula would be the effective aperture of the receiver [18][19].

$$A = G_r \frac{\lambda^2}{4\pi}, \tag{2.1}$$

which fits into the formula for received power as [18][19],

$$P_r = \frac{P_{t,obj} A}{4\pi R_r^2}. \tag{2.2}$$

$P_{t,obj}$	Power transmitted by the object
A	Effective aperture of the receiver
G_r	Gain of the receiving antenna
R_r	Distance from the receiver to the object
λ	Wavelength/-s of the signal

The next step is to calculate how much power is being intercepted from the transmitter. This is also the step where the RCS is introduced, with [18][19],

$$P_{t,obj} = P_t G_t \frac{\sigma_{obj}}{4\pi R_t^2} \tag{2.3}$$

P_t	Power transmitted by the transmitting antenna
G_t	Gain of the transmitter
R_t	Distance from the object to the transmitter
σ_{obj}	Radar cross section of the scanned object

Inserting (2.1) and (2.3) into (2.2), the received power in a bistatic radar system can be calculated as [18][19],

$$P_r = P_t G_t G_r \frac{\sigma_{obj} \lambda^2}{(4\pi)^3 R_t^2 R_r^2}. \tag{2.4}$$

For a monostatic radar, (2.4) can be reduced to [20],

$$P_r = P_t G^2 \frac{\sigma_{obj} \lambda^2}{(4\pi)^3 R^4}, \tag{2.5}$$

where R is the distance from the transmitting/receiving antenna to the object, and G is the gain of said antenna.

The response of multiple objects can be estimated by field superposition. This, however, neglects interaction between the different objects. The interesting measure is therefore not power, P_r,

$$\begin{aligned} |P_r| &= |E_r| \cdot |H_r| \\ &= \frac{1}{Z_0} |E_r|^2 \end{aligned}$$

but voltage, E_r, which is defined as,

$$\begin{aligned} E_r &= \sqrt{Z_0 P_r} \\ &= \sqrt{Z_0 P_t} G \frac{\lambda \sqrt{\sigma_{obj}}}{(4\pi)^{3/2} R^2} \end{aligned} \tag{2.6}$$

Though all components are supposed to be real to fit into the definition of power, the square root of the Radar Cross Section (RCS), $\sqrt{\sigma_{obj}}$, is assumed to be complex. This assumption is made to implement a phase in the received voltage which is calculated object-by-object. Summing the received voltages would then contain complex information which would allow for response interference, and the actual simulated radar response is then,

$$E_r = \sum_k E_{r,k}. \tag{2.7}$$

Here, k is a summation index over all scanned objects in the system. This is then simulated for each step of the simulated motion.

2.2 Electromagnetic Wave Propagation

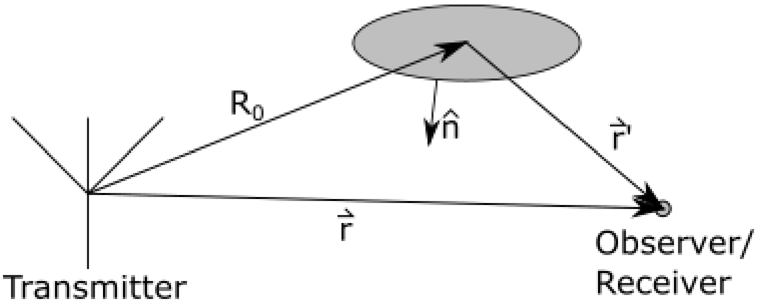


Figure 2.2: Vectors involved in (2.8).

When the signal is transmitted by the radar, an electromagnetic field will be generated propagating like a wave. Eventually this field will reach an object, and when it does it will interact with said object. The way that the electromagnetic field interacts with the object and is scattered depends on the shape and direction of the object and how the electromagnetic field scatters from the object in a given direction is described by integrating the scattered field over the entire surface of the object [21]. Some simplifying assumptions [22] are needed to reduce the surface field integral to a very easily solvable integral.

1. That the field is solved over either a non-closed surface or a finite portion of a closed surface.
2. That the receiver/observer is located in the far-field
3. That the fields at the surface can be represented in terms of the incident field

This then reduces the surface field integral into,

$$\bar{E}^s(\bar{r}) = \frac{j\omega\mu_0 e^{ik_0R_0}}{4\pi R_0} \int_S [(\hat{n} \times \bar{H}^T) - \hat{s}\cdot (\hat{n} \times \bar{H}^T)\hat{s} - \sqrt{\frac{\epsilon_0}{\mu_0}}(\hat{n} \times \bar{E}^T) \times \hat{s}] e^{-jk_0\hat{s}\cdot\bar{r}'} dS \tag{2.8}$$

and the scattered magnetic field as,

$$\bar{H}^s(\bar{r}) = -\frac{j\omega\epsilon_0 e^{ik_0R_0}}{4\pi R_0}\int_S [(\hat{n}\times\bar{E}^T) - \hat{s}\cdot(\hat{n}\times\bar{E}^T)\hat{s} + \sqrt{\frac{\mu_0}{\epsilon_0}}(\hat{n}\times\bar{H}^T)\times\hat{s}]e^{-jk_0\hat{s}\cdot\bar{r}'}dS \tag{2.9}$$

$\bar{r}$	Point in space where the scattered field is calculated, relative to transmitter location
ω	Angular frequency
μ_0	Free-space electric permeability
k_0	Free-space wave number
R_0	Distance from transmitter to scattering object
$\hat{n}$	Surface normal of scattering object
$\bar{H}^T$	Total magnetic field at each point on object surface
$\bar{E}^T$	Total electric field at each point on object surface
$\hat{s}$	Scattering direction, unit vector from object center to $\bar{r}$
ϵ_0	Free-space electric permittivity
$\bar{r}'$	Distance vector from each point on the surface of the scattering object to $\bar{r}$

Using the assumption that the object which is scattering the field is made out of a PEC material, the following rule [22] can be introduced,

$$\hat{n}\times\bar{E}^T = \hat{n}\times(\bar{E}^i + \bar{E}^s) = 0 \tag{2.10}$$

$$\hat{n}\times\bar{H}^T = 2\hat{n}\times\bar{H}^i \tag{2.11}$$

The index i denotes that the field is incident to the surface, and the index s denotes that the field is scattered. This rule is what is called the Physical Optics Approximation, and is what helps reduce (2.8) and (2.9) into,

$$\bar{E}^s(\bar{r}) = \frac{j\omega\mu_0 e^{ik_0R_0}}{2\pi R_0}\int_S [(\hat{n}\times\bar{H}^i) - \hat{s}\cdot(\hat{n}\times\bar{H}^i)\cdot\hat{s}]e^{-jk_0\hat{s}\cdot\bar{r}'}dS \tag{2.12}$$

$$\bar{H}^s(\bar{r}) = -\frac{j\omega\epsilon_0 e^{ik_0R_0}}{2\pi R_0}\int_S \sqrt{\frac{\mu_0}{\epsilon_0}}(\hat{n}\times\bar{H}^i)\times\hat{s}e^{-jk_0\hat{s}\cdot\bar{r}'}dS \tag{2.13}$$

Using that the electric far-field easily can be transformed into the magnetic far-field, and back again [18] with,

$$\bar{H}^s = Y_0\hat{s}\times\bar{E}^s,$$

where Y_0 is the free-space admittance, one can arbitrarily choose which field to calculate. Let's go with the electric field. Introducing further simplifications [18],

$$\begin{aligned}
\bar{H}^i &= H_0\hat{h}^i \\
\omega &= kc \\
Z_0 &= \sqrt{\mu_0/\epsilon_0} = \mu_0 c \\
E_0 &= H_0 Z_0
\end{aligned}$$

the scattered electric field can be simplified into,

$$\bar{E}^s = \frac{-j2kE_0e^{jk_0R_0}}{2\pi R_0}\int_S [(\hat{n}\times\bar{H}^i) - \hat{s}\cdot(\hat{n}\times\bar{H}^i)\cdot\hat{s}]e^{-jk_0\hat{s}\cdot\bar{r}'}dS \tag{2.14}$$

Under the assumption that there can be no component of the surface field distribution along the direction of scattering [18], this integral finally reduces to,

$$\bar{E}^s = \frac{-j2kE_0e^{jk_0R_0}}{2\pi R_0}\int_S (\hat{n}\times\bar{H}^i)e^{-jk_0\hat{s}\cdot\bar{r}'}dS \tag{2.15}$$

2.3 Radar Cross Section

The radar cross section, RCS, of an object is a description of how much of the radiated energy that an object scatters in a given direction [18][19]. One way to define the RCS is [18, p.190],

$$\sqrt{\sigma} = \lim_{R\longrightarrow\infty} 2\sqrt{\pi}R\frac{\bar{E}^s\cdot\hat{e}_r}{E_0}e^{-jkR} \tag{2.16}$$

Even though the scattered electric field can be used directly as a means of calculating the response, it is here used to help define the RCS of the objects which have been used. As such, inserting the derived scattered electric field, (2.15), into (2.16) gives the final definition of the square root of the RCS,

$$\sqrt{\sigma} = -j\frac{k}{\sqrt{\pi}}\int_S \hat{n}\cdot\hat{e}_r\times\hat{h}_i e^{jk\bar{r}\cdot(\hat{i}-\hat{s})}dS \tag{2.17}$$

There is a caveat. (2.17) can only be evaluated exactly for a handful of cases [18, p.190] and the hand which is being simulated is therefore comprised of three basic objects. A two-dimensional plate, and some cylinders and spheres, and the material for these objects are assumed to be perfectly conducting. The analytical expressions for objects relevant to this thesis are described in detail in [18, p.189-199].

An important part to get a realistic model is to account for object obstruction. What this means is that if one object blocks the line of sight from another object to the radar, the blocked object should not fully contribute to the total signal. An attempt to take this into consideration is not to simulate the received power, but the received electrical field. If, in (2.7), one were to sum complex values, one would also naturally get interference. Since all components of (2.6) are real by definition, the only component which can be complex is the square root of the RCS of the objects. Starting with the plate, the square root of the RCS is given by[18, p.191],

$$\sqrt{\sigma_{plate}} = -j\frac{kLW}{\sqrt{\pi}}\hat{n}\cdot\hat{e}_r\times\hat{h}_i\cdot e^{jk\bar{r}_0\cdot(\hat{i}-\hat{s})}\cdot sinc(\frac{1}{2}k\bar{L}\cdot(\hat{i}-\hat{s}))\cdot sinc(\frac{1}{2}k\bar{W}\cdot(\hat{i}-\hat{s})) \tag{2.18}$$

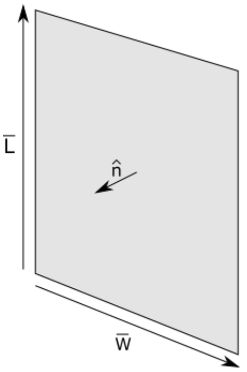


Figure 2.3: Geometry of the plate.

where,

$\hat{i}$	Incident wave direction
$\hat{s}$	Scattered wave direction
$\bar{r}_0$	The position vector of a signal origin on or near the plate
$\hat{n}$	Object surface normal
$\hat{e}_r$	Electrical polarization of receiver
$\hat{h}_i$	Magnetic polarization of incoming wave
$\bar{L}$	Direction and size, L, of a vector aligned with the length of the plate
$\bar{W}$	Direction and size, W, of a vector aligned with the width of the plate
k	Wavenumber of incident signal ($k = \frac{2\pi}{\lambda}$)

And to further clarify, $sinc(x)$ is the so-called Sampling Function, $sinc(x) = sin(x)/x$. The next object is the cylinder, which RCS square root is described as [18, p.195],

$$\sqrt{\sigma_{cyl}} = -jl\sqrt{\frac{2ka}{\hat{n}_0 \cdot (\hat{i} - \hat{s})}} sinc((1/2)kl\hat{z} \cdot (\hat{i} - \hat{s})) \times (\hat{n}_0 \cdot \hat{e}_r \times \hat{h}_i) e^{jk\bar{r}_0 \cdot (\hat{i}-\hat{s})} e^{jka\hat{n}_0(\hat{i}-\hat{s})-j\pi/4} \tag{2.19}$$

where,

$\hat{z}$	The z-direction of the cylinder
a	Radius of the cylinder
l	Cylinder length
$\hat{n}_0$	Surface normal such that normal, $\hat{z}$, and $\hat{i}$ are all in the same plane.

In the simulations, the RCS has been calculated assuming back scattering, as is the case with monostatic radar systems. Therefore $\hat{i} = -\hat{s}$. In monostatic radar systems, the transmitter is also the device receiving the signal, and so, $\hat{e}_r = \hat{e}_t$. This then leads to $\hat{e}_r \times \hat{h}_i = \hat{i}$ which leads to (2.18) being reduced to,

$$\sqrt{\sigma_{plate}} = -j\frac{kLW}{\sqrt{\pi}} \hat{n} \cdot \hat{i} \cdot e^{j2k\bar{r}_0 \cdot \hat{i}} \cdot sinc(k\bar{L}\hat{i}) \cdot sinc(k\bar{W}\hat{i}) \tag{2.20}$$

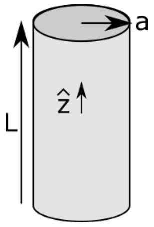


Figure 2.4: Geometry of the cylinder.

Using the simplifications that a monostatic radar system allows, the square root of the RCS of the cylinder is [18, p.196],

$$\sqrt{\sigma_{cyl}} = -jl\sqrt{ka\hat{n}_0\cdot\,\hat{i}}\cdot\, sinc(kl\hat{z}\cdot\,\hat{i})e^{j2k\bar{r}_0\cdot\hat{i}}e^{j2ka\hat{n}_0\cdot\hat{i}}e^{-j\pi/4} \tag{2.21}$$

where,

$\hat{z}$	The z-direction of the cylinder
a	Radius of the cylinder
l	Cylinder length
$\hat{n}_0$	Surface normal such that normal, $\hat{z}$, and $\hat{i}$ are all in the same plane.

Lastly, the square root of the RCS of the sphere can be described as [18, p.199],

$$\sqrt{\sigma_{sph}} = \sqrt{\pi}a[(1+\frac{1}{j2ka})e^{-j2ka} - \frac{1}{j2ka}], \tag{2.22}$$

where a here is the radius of the sphere. The sphere does not have a bistatic expression.

The values calculated here are the complex roots and as such contain phase information that will be used in the inverse Fourier Transform to give the proper delay of the response. This phase has to be added manually in the form of a phasor in the case of the sphere, since that expression does not contain any phase information due to distance between radar and object.

2.4 Impulse Response and Group Delay

To investigate whether the objects behave as they should, the first step have been to compare them to a FEKO-solved counterpart. But to also check if the implemented code applies the objects properly or not, investigating the transfer function,

$$C(f) = \sqrt{\frac{P_r}{P_t}} = G\frac{\sqrt{\sigma_{obj}(f)}\lambda(f)}{(4\pi)^{3/2}R^2}, \tag{2.23}$$

should give the necessary information regarding distancing. As described for (2.6), all components would normally considered to be real, but for the sake of this method the square root of the RCS, $\sqrt{\sigma}$, is here assumed to be complex.

A first examination is to investigate the phase of C for each of part of the signal frequency spectrum. Differentiating the extracted phase with respect to the angular frequency component gives the group delay for C. I.e.,

$$\begin{aligned} \tau_g &= \frac{d}{d\omega} arg(C) \\ arg(C) &= Arg(C) + 2\pi n, -\pi < Arg(C) \leq \pi \end{aligned} \tag{2.24}$$

The phase of C is wrapped so as to fit in the interval of $(-\pi\pi]$ and a necessary step is therefore to unwrap it. The group delay, τ_G, is given in seconds and is a measure of how much of a time delay is applied to each of the examined frequencies. As an example, an object at a distance of 0.3 m should have a τ_G of 2 ns for all frequencies. Half of that time to reach the object, half to get back.

Applying an inverse Fourier Transform to this transfer function, C, gives the impulse response of the object. The impulse response can then be quickly examined to see if the phase information from the analytical expressions is implemented properly, otherwise the signal might be distorted in unexpected ways.

2.5 Basic Radar Signal Processing

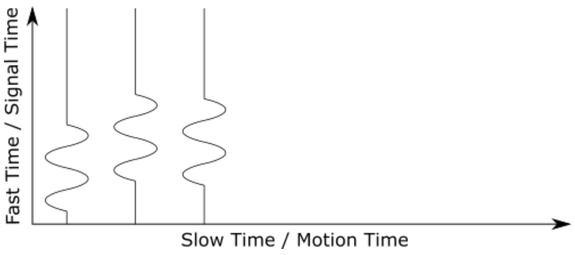


Figure 2.5: Time scales for radar.

As is explained in [23, p.289-293], in radar there are two time scales of importance, illustrated in Fig. 2.5. Slow-time, or motion time, is on the scale of signal repetition time, or more importantly, the time between recording received pulses. This time is on the scale of milliseconds, as the motions and gestures are being simulated in timesteps of about $2 - 3$ ms. Then there is fast-time, or signal time. Fast-time in this case is in the scale of picoseconds, with the transmitted pulse being repeated every ten nanoseconds. With this major difference in scale, each point of the motion can be considered to be frozen in the time it takes for the signal to reach it and return to the radar.

By multiplying fast-time with the speed of light in whatever medium the signal is passing through, one can calculate the maximum unambiguous range that a given pulse repetition time can cover.

$$R_{max} = \frac{T_d c}{2} \tag{2.25}$$

T_d is the pulse repetition time, how often the pulse is being transmitted, and c is the speed of light. This product is divided by two, due to the fact that the pulse needs to do a round trip to be received.

Each sampled signal is then considered as a column in a matrix, with the number of columns corresponding to the number of samples taken during the time of one motion or gesture. The number of rows in this final matrix corresponds to how fast the signal can be sampled for each sample point of the motion. To properly represent the signal, this sampling speed needs to follow the Nyquist criteria, i.e. the sampling frequency need to be greater than twice the main frequency of the signal. In the end, this gives a matrix, with each column representing the signal for the corresponding point of a motion or gesture.

When calculating the Time-Domain received signal based on the frequency spectrum of the transmitted signal and C, one can choose to calculate either the real value signal, or the analytic, complex-valued, signal. Transforming from a two-sided frequency spectrum would result in the real value signal, while only transforming from a one-sided spectrum would result in an analytic signal.

The Range-Time image should not depend on which type of signal that is calculated to any greater extent, but the Range-Doppler imaging might. Since the one-sided spectrum gives an analytic signal without further tampering, that is what is going to be used in this work.

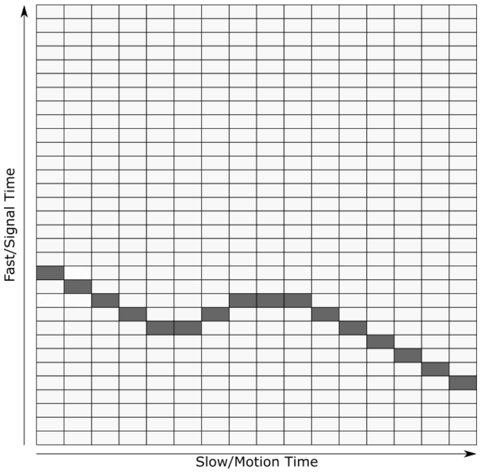


Figure 2.6: Applying a Fourier transform along slow-time values extracts Doppler frequencies of the object.

To then extract the velocity, or Doppler, data from this, the data is Fourier transformed along slow-time indexes, see Fig. 2.6. This is where the importance of an analytic signal lies. Taking this transform using a real-valued signal would result in a two-sided Doppler-spectrum, and therefore which way the object is moving can not be determined. Using an analytic signal would however resolve this issue. By introducing a phase, any real value will be accompanied by an imaginary part and therefore be distinguishable from other real-part equivalents. Applying a Fourier transform to this analytic signal would then give unambiguous values, so if the object were to move away from the radar, that image would be distinguishable compared to if the object were to move towards the radar.

Chapter 3

Method

3.1 Simulating Hand Gestures

To simulate the hand, two different approaches are compared. Both rely on simulating frequency domain coefficients, multiplied with parts which are general for the system, and then finally transform into time domain. The process of going from a generated signal to a calculated response can be seen in Fig. 3.1.

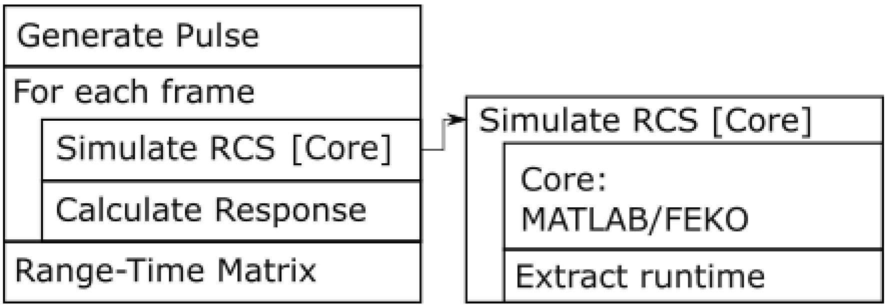


Figure 3.1: The steps taken to simulate a Time-Domain signal.

What is done first is to define the transmitted signal in Time-Domain, which is then transformed into Frequency-Domain to simplify the following calculations. This step can be seen in Fig. 3.2. For each frame of the motion, the RCS is then simulated in Frequency-Domain for each pose of the gesture or motion.

To simulate the RCS, one approach is to use analytic expressions in MATLAB, and the other would be to use a PO solver, such as FEKO. In MATLAB, calculations would be made matrix-by-matrix, and the time to calculate would be proportional to the resolution in Fast-Time and Slow-Time. Using FEKO, the run-time would be proportional to many more variables. Besides that there is a difference in how a model is implemented and how easy it is to define any given gesture or motion.

For each frame, the response is then calculated in Frequency-Domain and the values are added to a matrix at an index corresponding to the frame of the motion. As a last step this matrix is transformed into Time-Domain to be able to compare with the measured values.

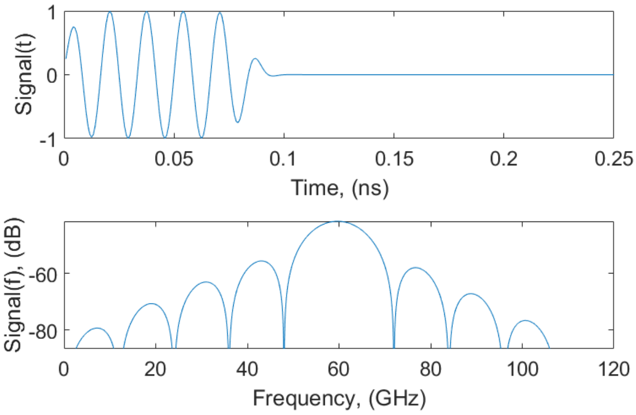


Figure 3.2: The defined Time-Domain signal, and the corresponding frequency spectrum. The images have been zoomed in on the relevant intervals.

3.2 Generating Radar Response

In MATLAB, the objects to be simulated will be given all values which are relevant. For the RCS based on MATLAB calculations that means that the direction of the objects, the distance between the transmitter/receiver and the objects, and the size of the objects will be defined. The relevant expression will then be used based on which object is being calculated.

Using FEKO, the object is described based on size and shape. It is then meshed based on the frequency/frequencies to be solved for. After solving for the RCS, the data is imported into MATLAB.

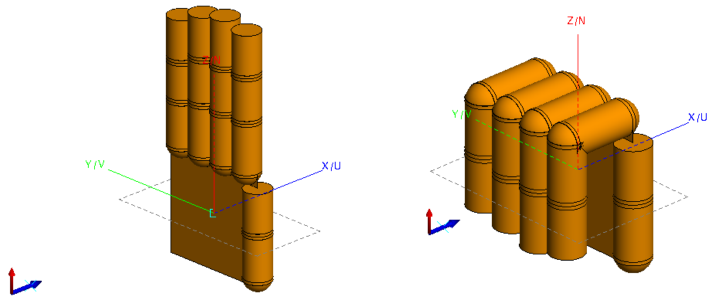
3.3 Implementation of Basic Objects

As a first step, the basic objects are implemented into MATLAB and FEKO. The RCS is then calculated in both solvers and compared. Since the final comparison is between the analytic expressions and the measured data, it is important to check that the transfer function, C, properly delays the signal in time. A secondary test is thereby performed. This test is to control that the distance to the object is properly described.

Here, the run-time will be compared between the two simulators, to give further values to compare when choosing simulator.

3.4 Analytic Model vs. Physical Optics Approximation

Secondly, two hand poses are modelled and simulated. This test is to see if there is any difference in value when comparing the MATLAB value to FEKO. The gestures simulated can be seen in Fig. 3.3:



(a) Open Hand **(b)** Clenched Hand

Figure 3.3: The two gestures simulated as a test.

These two models, which have been modelled in FEKO, has also been implemented in MATLAB. The blue arrow which is parallel to the "X/U" axis correspond to the direction of the incident field and the red arrow correspond to the polarization of said field. To be noted is that scattering direction is the opposite of the incident direction. In this stage, two variants for the PO approximation have been tested. The main one being the standard PO method, and the second one being a Large Element PO (LEPO). The LEPO, with its bigger and therefore fewer elements, should run faster for hopefully the same results.

A comparison between the three methods is shown later for the run-time for the basic objects, but the more important comparison is the run-time for the hand as a whole. Seeing as the simulation of gestures is the end goal, comparing the run-time frame-by-frame is far more important.

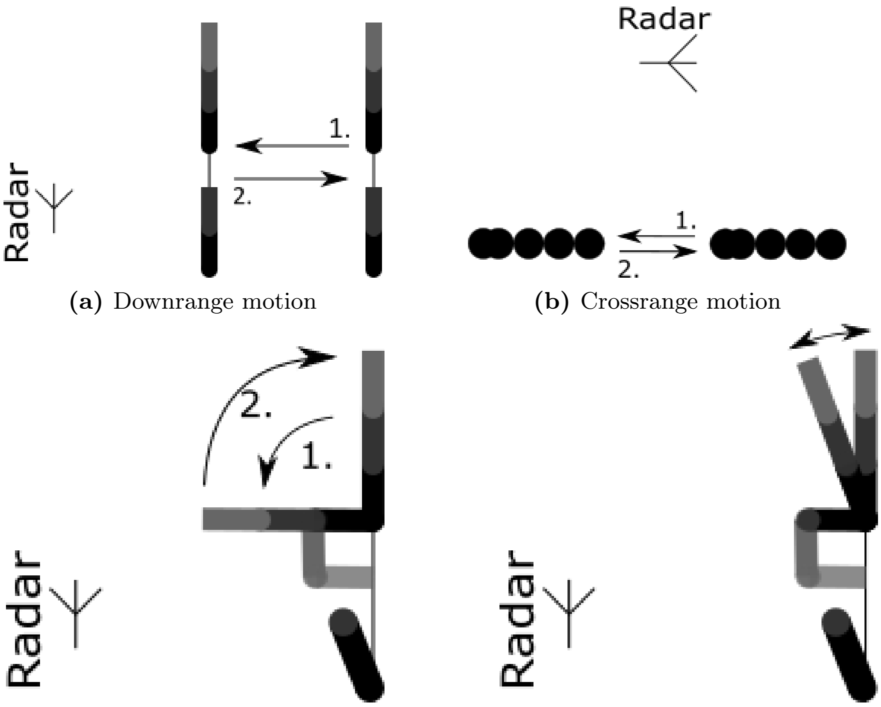
3.5 Simulated Radar Response vs. Measured Data

Finally, radar response will be simulated in MATLAB. In the end MATLAB is chosen as the simulator for the hand gestures, which means that the responses which have been simulated are based on the analytic expressions. Four different motions have been implemented as a test of the model based on analytic expressions, see Fig. 3.4.

There are then four gestures and motions that are going to be simulated, illustrated in Fig. 3.4. The first one is a simple downrange motion. This means that the hand is going to move in a straight line towards the radar. Starting at distance of 0.5 m, and then ending at a distance of 0.3 m over a time of 1.8 s.

The second motion is a cross range motion. The minimum distance between the hand and the radar will be 0.45 m, and the cross range motion will be from -0.15 m to 0.15 m in the horizontal plane.

The third motion is a clenched hand at a distance of 0.45 m from the radar. The index finger will then go from a completely straightened state, to a point where the first joint is bent 90 degrees, and then back again.



(a) Downrange motion

(b) Crossrange motion

(c) Index bending

(d) See-sawing fingers

Figure 3.4: The four motions that have been simulated, images not to scale.

The fourth and final motion is similar to the third one. The difference being that the index finger will only bend to about 20 degrees and then back. At the same time, the middle finger will start at 20 degrees, go to zero degrees, and then back.

Chapter 4

Results and Discussions

4.1 RCS of the Basic Objects

The first step of comparing the two simulators was to compare the basic objects, with equal size, over a frequency interval, which can be seen in Fig. 4.1.

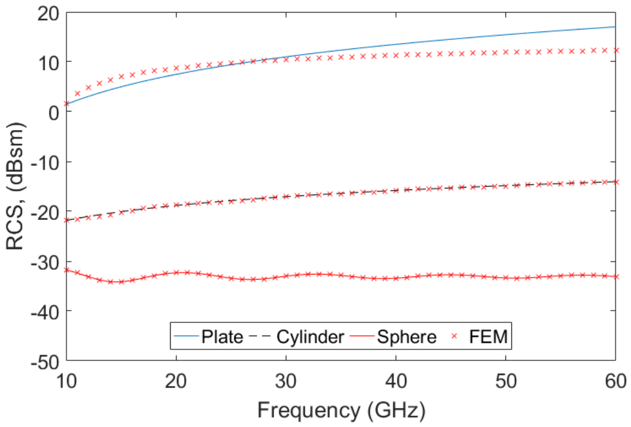


Figure 4.1: RCS comparison of the basic objects.

As can also be seen in Fig. 4.1, the RCS of the basic objects in MATLAB mostly follow the values from FEKO. The values from FEKO seem to swing around the values from MATLAB, with the analytic values seeming to be a trend. This is probably due to some behaviour which is not represented in the analytic expressions which have been used in this model.

Further on, the run-time for the plate, cylinder, and sphere can be seen in Table 4.1.

Run-time (s)	Plate	Cylinder	Sphere
FEKO	82.431	60.995	63.807
MATLAB	0.004	0.004	0.002

Table 4.1: The run-time for the three basic objects in the two simulators.

In Tbl. 4.1 it is shown that MATLAB is the far faster alternative, at least for simulating basic objects.

4.2 Impulse Response of Basic Objects in MATLAB

With the RCS seeming to have been implemented properly for each of the basic objects, it is time to see if the inherent phase of the values properly propagates the object in time-domain. The total frequency-domain coefficient, as defined in (2.23), was transformed into time-domain, which resulted in the impulse response of each of the basic objects, see Fig. 4.2.

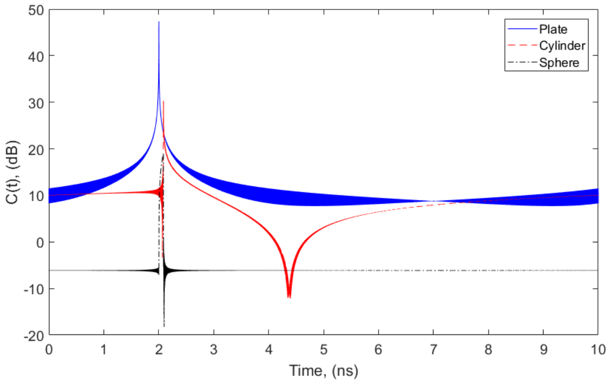


Figure 4.2: The impulse response for each of the basic objects, with the objects being placed at a distance of 0.3 m away from the radar. The figure zooms in on the relevant interval and the values are given in dB.

As can be seen in Fig. 4.2, the objects all seem to be properly delayed. The plate does have some non-casual behaviour which can seen as it is centered on 2 ns, but has a symmetrical behaviour around that center. This is due to that a finite set of frequencies have been simulated. Increasing the number of sampled frequencies, and the largest sampled frequency, solves this problem as the pulse narrows. But, there is a drop of about 30 dB down to the nearest side lobe, so the peak value is separated from the rest of the signal. To make sure that the delay will propagate properly, the group delay of C has also been extracted, see Fig. 4.3.

The plate is given the proper delay, but the other two objects deviates from the wanted value. For the cylinder, this is probably due to that it has phasors

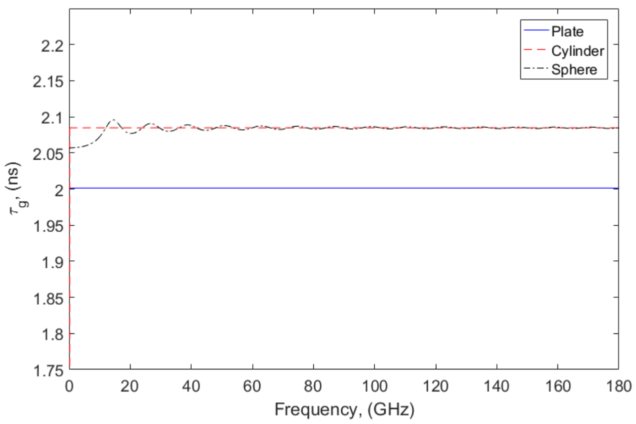
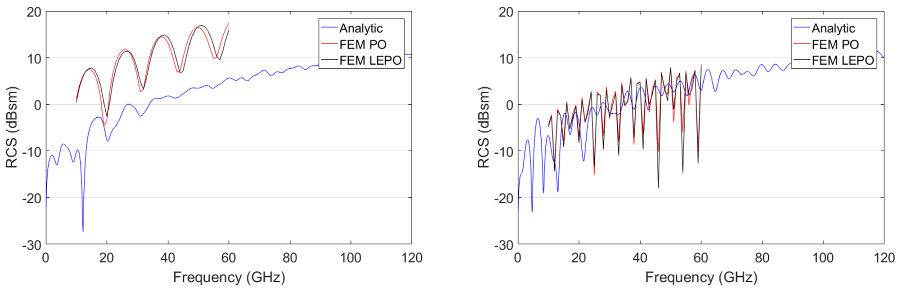


Figure 4.3: The group delay for the transfer function C.

other than the one based on the distance from the radar to the object. As can also be seen, the group delay for the sphere seems to oscillate. This is probably due to how it is defined, as there is a phasor attached to a decreasing imaginary value. What is less fortunate is that the stable value that it eventually reaches differs from the group delay it should exhibit. In the end, all objects display a group delay close enough to the expected value to assume that they should behave properly if implemented in a larger structure.

4.3 RCS of Basic Hand Gestures

With the RCS of the basic objects at least following the same trend in both simulators, moving on to the basic hand poses given in Fig. 3.3 is possible. As in the test of the basic objects, the simulations made in FEKO are made for a shorter and sparser interval, which can be seen in Fig. 4.4.



(a) Open Hand

(b) Clenched hand

Figure 4.4: The simulated RCS for the two hand poses for the two simulators.

What is made clear about the analytic model is that there isn't any separation in value between an open or a closed hand. There might be some phase difference in the model, but it is clear that some form of object interaction need to be implemented. This interaction would be to either implement ray-tracing or by trying to artificially shadow or obscure objects; If the palm is completely covered by fingers, it should not contribute to the received signal. It is also interesting that the value of the analytic model relates better with the closed pose than the open. If anything, the analytic model should relate better with the open hand pose, as all object always give some contribution.

As can also be seen in Fig. 4.4, there are also two lines for FEKO. These two are for two different methods, one being standard element size physical optics, and the other being a large element physical optics. This comparison has been made to see if it is possible to get a proper solution from FEKO, but for a shorter run-time.

Simulator	Run-time (s)
FEKO PO	265
FEKO LEPO	47
MATLAB	0.1

(a) Open Hand

Simulator	Run-time (s)
FEKO PO	237
FEKO LEPO	46
MATLAB	0.1

(b) Clenched Hand

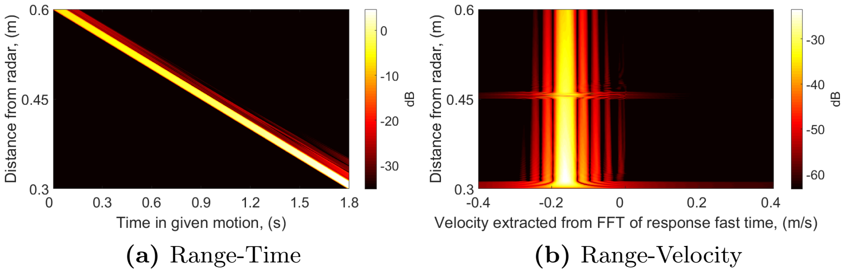
Table 4.2: The run-time comparison for the hand model. This is the time it would take to simulate one out of hundreds of frames per motion.

As can be seen in Tbl. 4.2, even though the two methods used in FEKO being rather close in their values, there is a major difference in run-time. But, they are still far slower than the analytic model which has been implemented in MATLAB.

Due to there being such a major difference in run-time for the two simulators, MATLAB has been chosen as the simulator to use for simulating the motions.

4.4 Simulations of Motions and Gestures

As a first test to see if the velocity was properly extracted, one motion has been simulated. The hand is simulated to be moving straight towards the radar starting at a distance of 0.6 m and then stopping at 0.3 m, at a speed of roughly 0.167 m/s. As can be seen in Fig. 4.5, this is properly represented in both the Range-Time image and in the Range-Doppler. This image looks as expected in regards to supposed distances to the objects.



(a) Range-Time **(b)** Range-Velocity

Figure 4.5: The simulated signal data for the open hand moving towards the radar. Distance decreasing from 0.6 m at the start, to 0.3 m at then end.

One of the first frames out of Fig. 4.5 has been extracted to show the difference between the transmitted signal and the one that has been received, and this can be seen in Fig. 4.6.

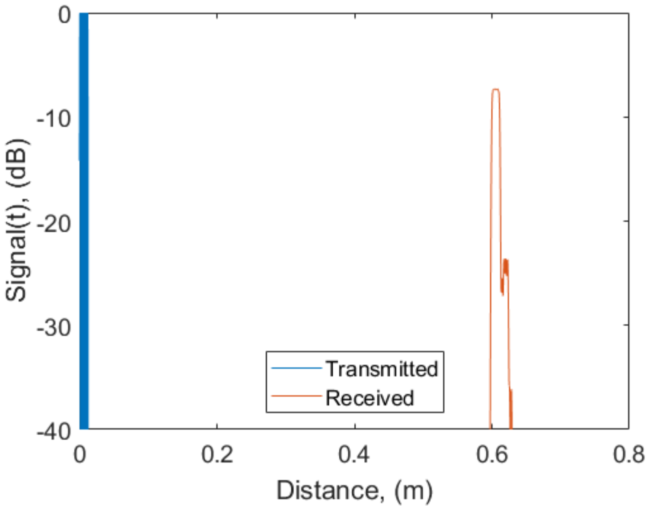


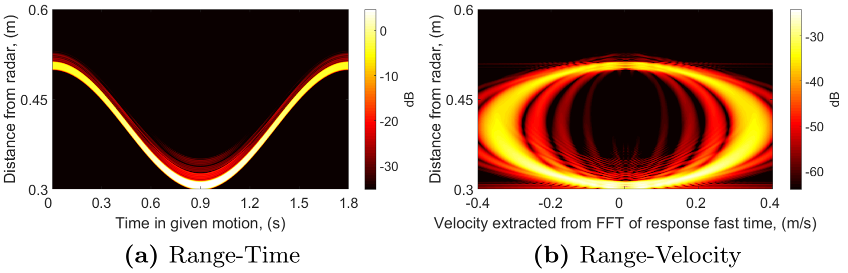
Figure 4.6: A comparison of the transmitted signal in Time-Domain and the received signal of frame 3 in Fig. 4.5a.

The thick line to the left is the transmitted signal and the signal at the right is the received signal. Here it is made clear that the signal is definitively affected by its trip to and from the hand, as it is reduced in strength and as it has been widened.

There are four motions that have been simulated. The first one is an open hand moving in a cosine manner towards and then back from the radar, see Fig. 4.7. The second one is a hand moving in cross range at a constant x-, and z-value, only changing its y-position at a cosine speed maxing out at roughly 0.79 m/s, see

Fig. 3.4b. The third one is a stationary hand folding all the fingers but the thumb, see Fig. 3.4d, and the fourth one is a stationary hand folding only the index, with the other fingers following slightly, see Fig. 3.4c.

For all simulations, the data has been cropped to focus the image around where there are signal values.



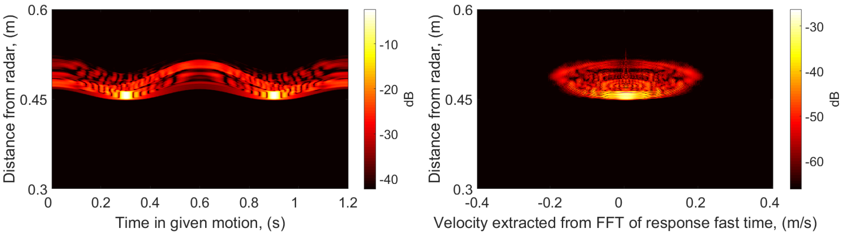
(a) Range-Time

(b) Range-Velocity

Figure 4.7: The simulated signal data for the open hand moving towards and away from the radar. Distance decreasing from 0.5 m at the start, to 0.3 m in the middle, and then back to 0.5 m.

For just a simple open hand moving towards the radar, it seems possible to simulate the response. The broadening of Range-Time, RT, signal as the hand moves closer to the radar is due to that the minimum distance to the radar for all objects other than the palm is that objects distance to the center of the palm. Since these objects move in a straight line which is not aimed at the radar, the radial velocity for these objects are cosine proportional to the angle between their line to the radar and the line from the palm to the radar. As the hand does not reach a point where these velocities differ drastically from the velocity of the palm, it instead broadens the velocity data in Fig. 4.5b.

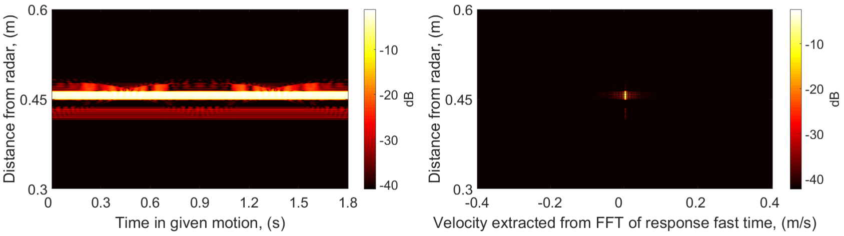
In Fig. 4.8 there is a first example of a more blurry motion. The cross range data goes from the thumb being further away than the palm, to the palm being further away than the thumb. The corresponding velocities in Fig. 4.8a are mostly comprised of a zero velocity at the distance closest to the radar, but there are traces of motion to be found at a slightly farther distance. That the strongest signal is from the zero-velocity point is probably due to that the palm is located at its closest to the radar, with an angle that coincides with the specular reflection. Therefore, the strongest component being transformed is the stationary one.



(a) Range-Time **(b)** Range-Velocity

Figure 4.8: The simulated signal data for the open hand moving in cross-range at a minimum distance of 0.3 m from the radar.

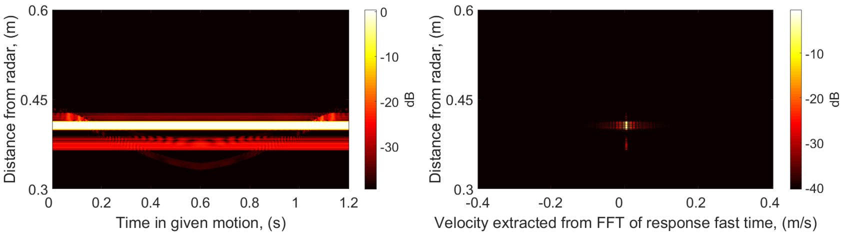
With the two previous motions being on a rather large scale, the next two are on a smaller scale. Fig. 4.13 is the response for a stationary hand with the ring finger, the pinky, and the thumb being folded in. The index finger and the middle finger are then moving in accordance to Fig. 3.4d. Even though fingers are bent in front of the palm, there is still a rather strong signal from the palm blocking whatever motion is being made.



(a) Range-Time **(b)** Range-Velocity

Figure 4.9: The simulated signal data for a hand at a distance of 0.45 m from the radar wiggling the index and the middle finger.

This last motion is the smallest motion that has been simulated. Smallest in the sense that it has the least amount of moving objects. Fig. 4.10 is the image set for the stationary hand, with only the index folding the whole way down. The other fingers are constantly bent down as seen in Fig. 3.4c.



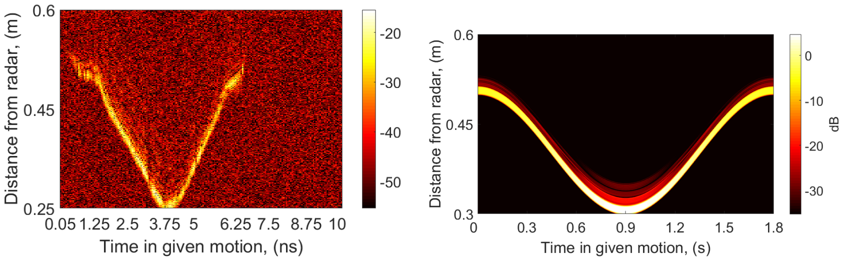
(a) Range-Time **(b)** Range-Velocity

Figure 4.10: The simulated signal data for a hand at a distance of 0.4 m from the radar folding the index finger.

4.5 Measured Motions and Gestures

The amount of measured data provided by H. Dahlberg and A. Evertsson was too much to be able to show it all, but a selection of the most average, visually, of the data have been selected. These data sets are also what the simulated data has been tried to replicate. The measured data is also shown in dB-scale, but the value-range has been cut-off to try and show the actual data, instead of the noise.

Starting with the down-range motion, there is a resemblance in the major feature of the motion. There is not much to compare between Fig. 4.11a and 4.11b, but it is clear that at least motions of this size can be compared.



(a) Measured **(b)** Simulated

Figure 4.11: The (a) measured and (b) simulated signal data for an open hand moving towards the radar, and then moving back. Range varying in the interval of 0.3 to 0.5 m.

Then comparing the cross-range motion it seems like there is not much to compare, but what is interesting is that the peak values in Fig. 4.11b seem to correspond to the shape of the peak values in Fig. 4.12a. This would mean that it might not be far off from the shape of the simulated signal being close to real value. There is clearly a difference in signal strength between the simulated and the measured data, but at least the shape seem to be comparable for the two larger motions.

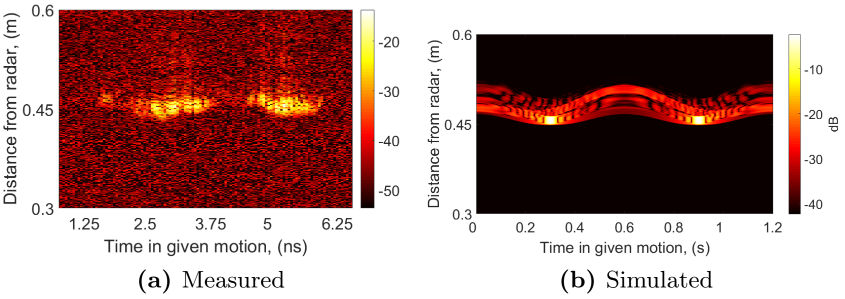


Figure 4.12: The (a) measured and (b) simulated signal data for a hand moving cross-range at a distance of roughly 0.45 m.

Comparing the measured data of bending a finger in the direction of the radar does not really give anything. There is not much in Fig. 4.14b which can be compared to Fig. 4.14a, and this is also the case for the two wiggling fingers, see Fig. 4.13b and 4.13a. There not being much to compare between the measured and the simulated data for the smaller motions is probably due to that the model is not very detailed. The palm is visible all throughout the simulated data in Fig. 4.14b, but it is not clear whether it is the palm or the other fingers which are visible in Fig. 4.14a.

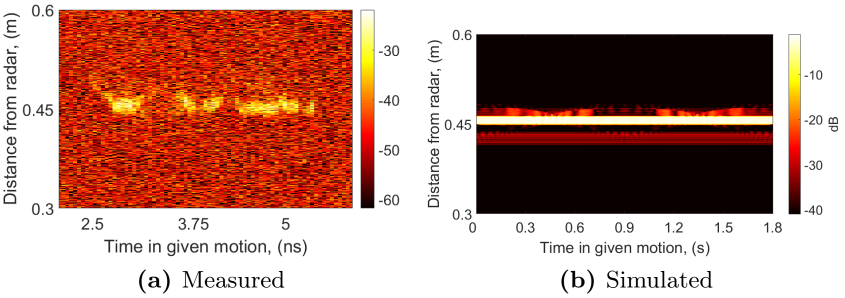
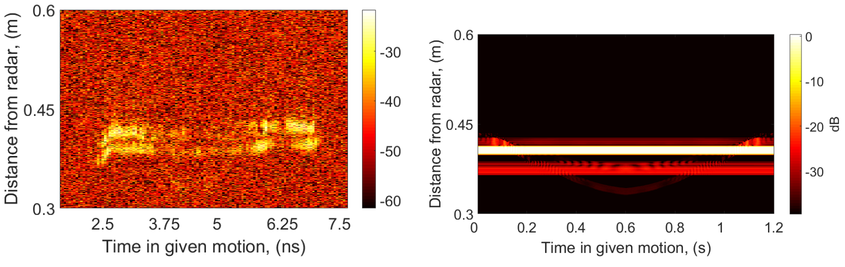


Figure 4.13: The (a) measured and (b) simulated signal data for a clenched hand stationary at a distance of about 0.45 m, index- and middle-finger see-sawing.

Something to consider is that all four motions have been differing in signal strength between the simulated and the measured signals. This is most probably due to that the signal which has been simulated has been assumed to be of unit value. In reality, the strength of the transmitted signal has to be taken into consideration to properly represent the system which the simulation is supposed to simulate.



(a) Measured **(b)** Simulated

Figure 4.14: The (a) measured and (b) simulated signal data for a clenched hand stationary at a distance of about 0.4 m, bending the index finger toward the radar, and then straighten it out.

Chapter 5

Conclusion and Outlook

5.1 Conclusion

Analytic expressions for the RCS of three basic objects have been implemented, and these objects have been assembled into a model of a hand. As can be seen in Fig. 4.4, the analytic expressions used deviates from the values given by the PO solution from FEKO. What is also shown in Fig. 4.4 is that the analytical model fits better to the clenched hand, than to the open hand.

Granted, the FEKO solution is not necessarily true to real values but when the simulated response is compared to the measured response, there is a difference in max signal value. This difference can either be due to the analytical model giving values which are too small, or that the setup used for measuring the response having less amplification and gain than assumed. Performing further measurements or using different setups would help discern which is the more likely cause.

Moving on to comparing the images. For the two motions that are on the larger scale the simulation is comparable to the measured values. Further trying to simulate the down-range and cross-range motions, so that the motions fit better to the motions measured should resolve most of the differences. The simulated response for the two smaller motions are however hard to compare to the measured data. The wiggling of the index and the middle finger show some resemblance in three bumps, but other than that it is hard to compare the simulated response to measured data, as what is seen the most is the response from the palm. There is some behaviour that might compare to the measured data, but the only way to know is to implement some form of object interaction.

This simulation does however not take into consideration how some parts of the hand might obscure others. There is also no ray-tracing which might redirect some part of the signal which would otherwise get lost. If these components were to be implemented the analytic model may compare to measured values for even the smaller motions. As it is currently, the analytic model can be used to represent larger motions, but it has problems with properly representing smaller gestures.

5.2 Outlook

With some further work on the interaction between the objects of which the hand consists, the analytic model should prove to be accurate to measured data. To properly represent reality, some random variance could then be introduced to the models parameters, and this would then allow for the model to be used as training data for neural networks, reducing the load of generating such data. Applying variance in the length of the fingers, some erratic behaviour in the motions, and then some noise floor. These changes are what is probably required to represent measured data, with the last one having to be dependent on the system used.

References

[1] M. Billinghurst, T. Piumsomboon, and H. Bai. Hands in space: Gesture interaction with augmented-reality interfaces. *IEEE Computer Graphics and Applications*, 34(1):77–80, Jan 2014.

[2] G. Li, H. Wu, G. Jiang, S. Xu, and H. Liu. Dynamic gesture recognition in the internet of things. *IEEE Access*, 7:23713–23724, 2019.

[3] Xinghao Chen, Guijin Wang, Hengkai Guo, Cairong Zhang, Hang Wang, and Li Zhang. Mfa-net: Motion feature augmented network for dynamic hand gesture recognition from skeletal data. *Sensors*, 19(2), 2019.

[4] Q. Wan, Y. Li, C. Li, and R. Pal. Gesture recognition for smart home applications using portable radar sensors. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6414–6417, Aug 2014.

[5] T. Sakamoto, X. Gao, E. Yavari, A. Rahman, O. Boric-Lubecke, and V. M. Lubecke. Hand gesture recognition using a radar echo i–q plot and a convolutional neural network. *IEEE Sensors Letters*, 2(3):1–4, Sep. 2018.

[6] Y. Kim and B. Toomajian. Hand gesture recognition using micro-doppler signatures with convolutional neural network. *IEEE Access*, 4:7125–7130, 2016.

[7] Z. Peng, C. Li, J. Muñoz-Ferreras, and R. Gómez-García. An fmcw radar sensor for human gesture recognition in the presence of multiple targets. In *2017 First IEEE MTT-S International Microwave Bio Conference (IMBIOC)*, pages 1–3, May 2017.

[8] B. Dekker, S. Jacobs, A. S. Kossen, M. C. Kruithof, A. G. Huizing, and M. Geurts. Gesture recognition with a low power fmcw radar and a deep convolutional neural network. In *2017 European Radar Conference (EURAD)*, pages 163–166, Oct 2017.

[9] X. zhang, Q. Wu, and D. Zhao. Dynamic hand gesture recognition using fmcw radar sensor for driving assistance. In *2018 10th International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 1–6, Oct 2018.

[10] S. Hazra and A. Santra. Robust gesture recognition using millimetric-wave radar system. *IEEE Sensors Letters*, 2(4):1–4, Dec 2018.

[11] Z. Zhang, Z. Tian, and M. Zhou. Latern: Dynamic continuous hand gesture recognition using fmcw radar sensor. *IEEE Sensors Journal*, 18(8):3278–3289, April 2018.

[12] Zhi Zhou, Zongjie Cao, and Yiming Pi. Dynamic gesture recognition with a terahertz radar based on range profile sequences and doppler signatures. *Sensors*, 18(1), 2018.

[13] Faheem Khan, Seong Kyu Leem, and Sung Ho Cho. Hand-based gesture recognition for vehicular applications using ir-uwb radar. *Sensors*, 17(4), 2017.

[14] Shahzad Ahmed, Faheem Khan, Asim Ghaffar, Farhan Hussain, and Sung Ho Cho. Finger-counting-based gesture recognition within cars using impulse radar with convolutional neural network. *Sensors*, 19(6), 2019.

[15] S. A. Akhmanov and S. Yu Nikitin. *Physical Optics.* Clarendon Press, 1997.

[16] P. van Dorp and F. C. A. Groen. Human walking estimation with radar. *IEE Proceedings - Radar, Sonar and Navigation*, 150(5):356–365, Oct 2003.

[17] V. C. Chen. Detection and analysis of human motion by radar. In *2008 IEEE Radar Conference*, pages 1–4, May 2008.

[18] Eugene F. Knott, John F. Shaeffer, and Michael T. Tuley. *Radar Cross Section, 2nd edition.* Scitech Publishing, 2004.

[19] Caner Özdemir. *Inverse Synthetic Aperture Radar Imaging With Matlab Algorithms.* Wiley, 2012.

[20] Merrill I. Skolnik. *Radar Handbook.* McGraw-Hill, Inc., 1970. p.1-6.

[21] Julius A. Stratton. *Electromagnetic Theory.* John Wiley Sons, Inc., 2007.

[22] George T. Ruck, Donald E. Barrick, William D. Stuart, and Clarence K. Krichbaum. *Radar Cross Section Handbook - Volume 1.* Plenum Press, 1970. p.50-59.

[23] Mark A. Richards. *Principles of Modern Radar, Volume I - Basic Principles.* Institution of Engineering and Technology, 2010.

Printed by Tryckeriet i E-huset, Lund 2019

LUND
UNIVERSITY

Series of Master's theses
Department of Electrical and Information Technology
LU/LTH-EIT 2019-713
http://www.eit.lth.se