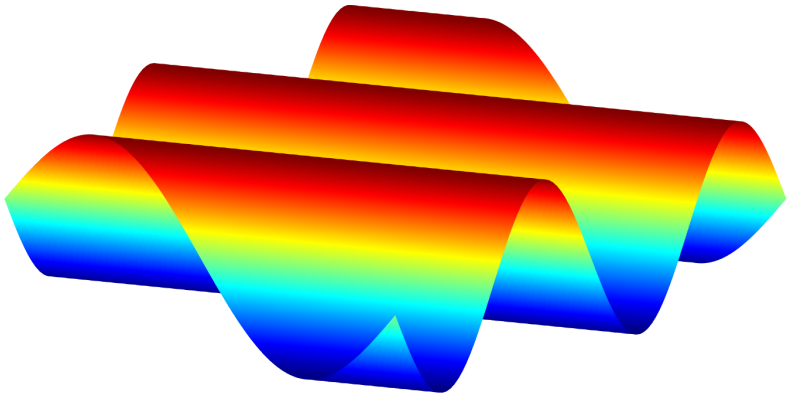

Sensing and Communication Using FMCW Radar Technology

Master Thesis

Information and Communication Technology (MPICT)

Leon Janetzko & Alexander Rex

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Sensing and Communication using FMCW Radar Technology

Leon Janetzko & Alexander Rex

Department of Electrical Engineering
Division of Communications, Antennas, and Optical Networks
Chalmers University of Technology
Gothenburg, Sweden 2024

Sensing and Communication Using FMCW Radar Technology

Leon Janetzko & Alexander Rex

© Leon Janetzko & Alexander Rex, 2024.

Supervisor: Navid Amani, José Miguel Mateos Ramos
Examiner: Henk Wymeersch, Chalmers University of Technology, Department of Electrical Engineering

Master's Thesis 2024
Department of Electrical Engineering
Division of Communications, Antennas, and Optical Networks
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Using FMCW Radar Technology to Perform Communication and Sensing.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2024

Sensing and Communication using FMCW radar technology
Leon Janetzko & Alexander Rex
Department of Electrical Engineering
Chalmers University of Technology

Abstract

This thesis involves the development of algorithms to perform simultaneous communication and sensing using FMCW radar. Information is encoded in the vibration pattern of a vibra-motor, which can be seen in the phase of the received chirps at the radar side. The developed algorithms should be able to decode the received phase signal into data, while still doing the sensing part. To achieve this, a set of modules were developed, including spatial object detection, vibration detection, object tracking, trajectory management, clutter elimination, motion suppression, trajectory stitching, and data extraction. The final system is able to track the movement, and decode the data of one vibra-motor under certain conditions, which were evaluated through measurements under different scenarios. For the case of a stationary vibra-motor without scattering objects, with a symbol time of 90 ms, bit error rates less than 1 % were measured all the way up to 2.5 m. When scattering objects were present, or when the vibra-motor was moving, the signal quality degraded significantly, resulting in either larger bit error rates or an inability to decode the data. When smaller movements were introduced on the motor side, the data has potential to be decoded according to one measurement with a bit error rate of 1.67 %, when mounting the motor on a moving person, the data could not be decoded but the algorithm is successful at tracking the motor in both cases. Despite mixed results, the thesis showed simultaneous communication and sensing to be possible by only extending the signal processing on the radar side without integrating additional communication equipment to the system.

Keywords: Frequency Modulated Continuous Wave (FMCW) Radar, Joint Communication and Sensing (JCAS), Digital Signal Processing (DSP).

Acknowledgements

The participants in this thesis would like to thank the company RadChat for providing us with guidance as well as the hardware for the project. We are extra grateful to our supervisors Navid Amani from RadChat, as well as José Miguel Mateos Ramos for providing weekly advice and guidance on the project. Finally, a thanks to Henk Wymeersch for being the examiner and evaluating the thesis.

Contents

List of Figures viii

List of Tables ix

List of Acronyms 1

1 Introduction 3
1.1 Background . . . 3
1.2 Aims and Goals . . . 4
1.3 Research Questions . . . 4
1.4 Literature Overview . . . 4

2 Theory 6
2.1 FMCW Radar . . . 6
2.1.1 Radar Cross Section . . . 7
2.1.2 Data Grouping and Maps . . . 8
2.2 Target Detection and CFAR . . . 9
2.2.1 Cell-Averaging CFAR . . . 10
2.2.2 Angle Detection . . . 10
2.3 Data Decoding . . . 11
2.3.1 Down-Conversion . . . 11
2.3.2 Matched Filtering . . . 12
2.3.3 Packet Synchronization . . . 12
2.3.4 Threshold Setting . . . 13
2.4 Tracking and Motion . . . 14
2.4.1 Clustering and Closeness . . . 14
2.4.2 Prediction . . . 15
2.4.3 Association . . . 17
2.4.4 Creation and Deletion . . . 17
2.5 Clutter Elimination and Circle-Fitting . . . 18
2.6 Motion Suppression . . . 20
2.7 Trajectory Stitching . . . 21
2.8 BER and Statistical Significance . . . 22
2.9 Summary . . . 23

3 Method 24
3.1 Aim and Limitations . . . 24

3.2 Vibration motor 24
3.3 Radar 25
3.4 Measurement Setup (Stationary Case) 25
3.4.1 DSP Chain 27
3.5 Moving Vibra-Motor Algorithm 29
3.5.1 DSP chain 29
3.6 Summary 31

4 Results 32
4.1 Structure and Performance Metrics 32
4.2 Stationary Algorithm Showcase 32
4.2.1 Sensing 32
4.2.2 Vibration Detection 34
4.2.3 Communication Data Extraction 34
4.3 Stationary Case Evaluation 38
4.3.1 Impact of Scattering Object 39
4.4 Moving Algorithm Showcase 41
4.4.1 Test 1: Slowly Moving Vibra-Motor on Tripod 41
4.4.2 Test 2: Pseudo-Stationary Vibra-Motor on Tripod 42
4.4.3 Test 3: Quickly Moving Vibra-Motor on Vest 43
4.4.4 Test 4: Pseudo-Stationary Vibra-Motor on Vest 44
4.5 Summary 45
4.5.1 Stationary Algorithm 45
4.5.2 Moving Case Algorithm 45

5 Conclusion 46
5.1 Stationary Algorithm 46
5.1.1 Bit Error Rate 46
5.1.2 Metric for Deciding Optimal Signal 46
5.1.3 Scattering Objects 46
5.1.4 Overall Performance 47
5.2 Moving Case Algorithm 47
5.2.1 Tracking and Trajectories 47
5.2.2 Impact of Motion 47

List of Figures

2.1 Frames and Range-Velocity Processing 8
2.2 CFAR: CUT, Guard and Training Cells. 10
2.3 Example of a constellation diagram according to equation (2.27), assuming = 1 13
2.4 Prediction and Association Visualization 17
2.5 Clutter Illustration 18
2.6 Circle-Fitting Algorithm Steps 19
2.7 Phase in Motion 21
2.8 Trajectory Stitching 22

3.1 Measurement Setup Stationary Case 26
3.2 Overview of Vibra-Motor Casings 27
3.3 Vibra-Motor Close-up 27
3.4 DSP Block Diagram for the Stationary Case. 28
3.5 Phase extraction of one stationary object extracted over multiple received chirps. 28
3.6 DSP Block Diagram for the Moving Case Part 1 29
3.7 DSP Block Diagram for the Tracking and Trajectory Management Step 30
3.8 DSP Block Diagram for the Moving Case Part 2 30

4.1 Range Doppler Map from the First Frame. 33
4.2 Range Doppler Map after Applying 2D-CFAR on the First Frame. . . 33
4.4 FFT of Received Signal from Range Bin 29 35
4.5 FFT of Down-Converted Signal from Range Bin 29 36
4.6 Correlation Plot with Preamble 36
4.7 Constellation Plot of Samples Signal 37
4.8 Eye Diagram 37
4.9 Threshold for Sampling (Red Line) 38
4.10 Measurement Setup for Scattering Object near Vibra-Motor 40
4.11 Eye Diagram for Scattering Object near Radar. 41
4.12 Trajectory Test 1 41
4.13 Eye Diagram for Test 1 42
4.14 Trajectory Test 2 42
4.15 BER vs Time and Range for Test 2 43
4.16 Vest 44
4.17 Trajectory Test 3 44
4.18 Trajectory Test 3 45

List of Tables

2.1 Table of barker codes 12

3.1 Explanation for Con guration Parameters for FMCW Radar 25

4.1 Vibration Range Indices 34
4.2 BER for a symbol time $T = 0.09$ s for two separate measurements, the one to the left transmitted approximately 570 bits and the one to the right 310 bits 38
4.3 BER for a symbol time $T = 0.06$ s, bits transmitted 480 38
4.4 BER for a symbol time $T = 0.03$ s, bits transmitted 1000 39
4.5 BER for an angle = 15 degrees, $T = 0{:}09$ s, bits transmitted 310 . 39
4.6 Ranges and BERs for Test 2 43

List of Acronyms

AoA	Angle of arrival
BER	Bit error rate
BERUB	Bit error rate upper bound
CA-CFAR	Cell averaging CFAR
CFAR	Constant false alarm rate
CUT	Cells under test
DBSCAN	Density-based spatial clustering
DFT	Discrete Fourier Transform
DSP	Digital signal processing
EVM	Error vector magnitude
FFT	Fast Fourier Transform
FMCW	Frequency Modulated Continuous Wave
JCAS	Joint Communication and Sensing
RCS	Radar-cross section
RDM	Range-Doppler map
RVA	Range, velocity, AoA
SNR	Signal to noise ratio

1
Introduction

1.1 Background

The automotive industry has in recent years been investing heavily in technology that enables higher reliability, driver safety, and driving automation [1]. A crucial component in most of these systems is the use of frequency-modulated continuous wave (FMCW) radars commonly operating in the 77 GHz band [2]. The advantage of using radar-integrated systems is that they improve driver safety and make it more redundant by, for instance, providing the driver with information about obstacles or other vehicles even with obscure weather conditions [3]. These technologies have helped to improve the aforementioned aspects of driving and now, there is a push to further expand them with communication capabilities. Societies are becoming more interconnected in all areas of life, for instance, industry 4.0, machine-to-machine communications, smart grids, and 5G networks. So, it is no surprise that there is a trend in the automotive industry to integrate communication technologies within their systems as well. This could enable things like car-to-car communication [1] by, for instance, collecting information about other vehicles' turn signals or brake lights, or communicating with road infrastructure like tra c lights [4].

One of the proposed methods for enabling the car to communicate with its environment is the use of back-scatter communication through the use of tags. A tag has the property that it modulates an incoming signal with additional information that can be extracted at the radar side [4]. This can be useful for sending additional information to other vehicles on the road and could, as an example, help with vehicle identi cation since a radar is capable of detecting an object but not identifying it. An example of this could be a car that has an individual id it wants to transmit. The thesis is done in collaboration with the company Radchat, which also has an interest in exploring the possibility of using this technology to identify workers that have unique ids within a mining environment. The design of such a system would involve designing the tags and performing additional signal processing on the radar side, which also has the advantage of not needing the integration of additional communication systems [1].

The thesis aims to develop a signal processing algorithm to develop a joint communication and sensing system (JCAS) based on FMCW radar measurements and evaluate its performance. For the tag, an o -the-shelf vibration motor (vibra-motor) will be used that encodes information in a vibration pattern for the radar to detect.

The thesis builds upon existing work on the topic to evaluate the feasibility of such a system on the radar side without integrating new communication systems.

1.2 Aims and Goals

The aim is to develop an algorithm implemented on the radar side that is capable of performing simultaneous sensing, tracking, and communication of one moving vibra-motor using FMCW radar hardware. The final demonstration should be able to show the radar being able to sense and track the vibra-motor and extract its communication data, thus, creating a JCAS system.

1.3 Research Questions

The following points are of great importance to the thesis and highlight some questions it aims to answer.

- Is it possible to perform both sensing and communication at the same time? Is the radar able to detect the vibration pattern?
- What will be the impact of having scattering objects in the presence of targets?
- Will a Kalman filter be needed to gain satisfactory results with regard to tracking?
- Are there certain angles or ranges that impact the performance of the system?
- What are the limits on data rates?

1.4 Literature Overview

Researchers, especially in the automotive industry, have a great interest in JCAS systems involving automotive radar. For instance, [1] studies the feasibility of developing tags for cars that can modulate incoming FMCW radar waves by switching the power supply of the tag amplifier. Similarly, [4] proposes the integration of a self-packaged active radio tag with road infrastructure to transmit important data to cars. The tags used in such systems usually deploy amplifiers, micro-controllers and other electronics components to modulate the incoming wave, which is a key difference from this thesis.

A different approach is studied in [3], which develops and evaluates the performance of another FMCW radar JCAS system called mmRipple, this time using smartphones as tags. The data in this case is being transmitted by mechanical vibrations from the phone, therefore modulating an incoming radar wave with information. This paper is of particular interest to this thesis, since it also aims at using mechanical vibrations for communication. mmRipple utilizes pattern recognition as a metric for evaluating their JCAS system which they define as the rate that their system correctly matches recovered vibration patterns to their corresponding programmed patterns. They achieve a pattern recognition accuracy of 97.67 % for

handheld smartphones, 78.22 % when said phones are placed inside of pockets, and 72.89 % when phones are being held by people with some arm movement present. These results are interesting, since they suggest that it might be possible to detect the vibra-motors messages, even when placed on a person that is moving. In short, mmRipple provides a solid foundation for investigating the feasibility for JCAS with vibrations, and this thesis aims at recreating some of the results presented in their paper, this time using a vibra-motor instead of smartphones. It also aims at expanding on the capabilities of mmRipple by further evaluating the impact of mounting the vibra-motor on a person that is moving in the environment.

2

Theory

In this chapter, the basics of FMCW radar necessary to understand the thesis will be covered.

2.1 FMCW Radar

An FMCW radar transmits pulses (chirps) consisting of a sinusoid of a linearly increasing frequency described by

$$s(t) = \exp[\ j(2\ f_c t + \ Kt^2)];\ 0 < t < T_c \tag{2.1}$$

where T_c is the chirp ramp time, f_c is the carrier frequency and K is the frequency slope i.e the chirp bandwidth divided by the ramp time. These chirps are continuously transmitted by the radar, the time between each one being the chirp idle time. Furthermore, a prede ned number of chirps can be grouped together into a frame, so transmitting one frame corresponds to transmitting a certain amount of chirps. If there is a point target at range d from the radar, these chirps are re ected back to the radar, attenuated, and time-shifted according to

$$r(t) = \quad s\ \ t\ \ \frac{2d}{c}\ \ ! \ : \tag{2.2}$$

where is the path loss and c is the speed of light. To simplify the calculations, let $t_d = 2d/c$. At the receiver, a beat signal is produced through mixing of the received chirp with the transmitted one, we have

$$m(t) = s(t)r(t) \tag{2.3}$$

$$= \quad \exp[\ j(2\ Kt\ _d t + 2\ f\ _c t_d\ _{!\#}\ \ Kt\ _d^2)] \tag{2.4}$$

$$\exp\ \ j\ \ 2\ \frac{2Kd}{c}t + \frac{4\ d}{c} \tag{2.5}$$

$$= \quad \exp[\ j\ (2\ f\ _{IF}\ t + \ )]; \tag{2.6}$$

which is then sampled and output by the receiver. The approximation is justi ed partly because the round-trip time delay t_d typically is very small compared to the chirp ramp time T_c, and partly because the carrier frequency f_c is larger than the

chirp bandwidth B. The reasoning is described in the equations below.

$$2f_c t_d \quad Kt_d^2 \quad ! \tag{2.7}$$

$$= 2 \quad 1 \quad \frac{Kt_d}{2f_c} \quad f_c t_d \quad ! \tag{2.8}$$

$$= 2 \quad 1 \quad \frac{Bt_d}{2T_c f_c} \quad f_c t_d \tag{2.9}$$

$$t_d << T_c \ \& \ B < f_c \quad ! \tag{2.10}$$

$$=) \quad 2 \quad 1 \quad \frac{Bt_d}{2T_c f_c} \quad f_c t_d \quad 2f_c t_d \tag{2.11}$$

So, a point object at a certain range correspond to a single frequency component in the beat signal. Multiple objects results in a superposition of frequency components of varying magnitude and phase due to di erent ranges and/or path losses described by

$$m(t) \quad \sum_i \quad _i \quad \exp \quad " \quad j \quad 2 \quad \frac{2Kd_i}{c} t + \frac{4 d_i}{c} \quad !\# \quad : \tag{2.12}$$

This means objects at di erent ranges can be resolved by taking the discrete Fourier transform (DFT) of the beat signal. In the DFT, frequency translate to a range through

$$f_{IF} = \frac{2Kd}{c} \quad () \quad d = \frac{cf_{IF}}{2K}: \tag{2.13}$$

To derive the resolution of the radar, two objects spaced d apart can be resolved if

$$f_{IF} = \frac{2K \quad d}{c} > \frac{1}{T_c} \quad =) \quad d > \frac{c}{2KT_c} = \frac{c}{2B} \tag{2.14}$$

where B is the chirp bandwidth. The bandwidth remains constant at 4 GHz in our setting, meaning the range resolution is 3.75 cm. Thus, by looking at equation (2.6), a small change in range ($<< 3{:}75$ cm) will not be signi cant in terms of beat frequency f_{IF} but detectable with respect to phase . As a consequence, a frequency component's phase over consecutive chirps can be used to measure small range changes like a tiny vibration or velocity.

The idea is to exploit these detectable phase variations to transmit data from an object to the radar using vibrations. More speci cally, an arbitrary bit sequence is encoded in the vibration pattern of a vibra-motor, through on-o -keying (OOK), which is observed in the phase of the re ected chirps. Furthermore, the elements in the DFT of the beat signal are referred to as range bins. If a vibra-motor is located at range bin k the signal from the vibra-motor can be extracted by looking at the phase of range bin k.

2.1.1 Radar Cross Section

When working with radar systems, it is of bene t to understand the concept of radar cross section (RCS). The RCS for an object can be described as its visibility

to the radar, meaning it is a value for an object that describes its re ectivity to electromagnetic waves. It is de ned as

$$= \frac{\text{power re ected toward source/unit solid angle}}{\text{incident power density}=4} = \lim_{R!1} \; 4 R^{2} \frac{jE_r j^2}{jE_i j^2}$$

where E_r is the re ected electric eld from the object, E_i is the incident electric eld, and R the distance to the object, the de nition is taken from Skolniks book on radar systems [5]. The amount of re ected power is dependent on the characteristics of the object the wave is re ected upon, for instance the objects shape and material.

2.1.2 Data Grouping and Maps

In this thesis, the samples of a chirp are referred to as fast-time or chirp samples, while consecutive chirps are called slow-time samples. A group of chirps are bundled together in a matrix (frame), displayed Figure 2.1a, with fast-time samples along the columns and slow-time samples along the rows. Performing the DFT along the columns (range-DFT) separates objects in terms of range. Similarly, performing the DFT along the rows separates objects in terms of phase changes over time (velocity). The resulting matrix, see Figure 2.1b, is called the Range-Doppler map (RDM).

If the map has size $N_R \; N_V$, then element $(k_R; k_V)$ correspond to signal components from objects at a certain range R and radial velocity V_R, speci cally

$$k_R \; =) \quad R = \frac{k_R \quad 1}{N_R} \frac{c}{2K T_s} \tag{2.15}$$

$$k_V \; =) \quad V_R = 2 \; \frac{k_V \quad 1 \quad 0{:}5N_V}{N_V} \; ! \; \frac{}{4T_{C2C}} \tag{2.16}$$

where T_s is the fast-time sampling time and T_{C2C} is the slow-time sampling time.

(a) Frames (b) Range-Velocity Map Processing

Figure 2.1: Frames and Range-Velocity Processing

Moreover, the used radar has multiple receiver (Rx) antennas, organized in a linear array, each producing unique beat signals. Assume an object is located at angle in the same plane as all Rx antennas where = 0 is directly in front of the radar.

As a consequence, the beat frequency signal of all Rx antennas should be similar, di ering only in a phase shift according to

$$m_k(t) = m_1(t) \exp\left(-j\frac{2}{\lambda}(k-1)d_A \sin(\theta)\right) \tag{2.17}$$

where k is the Rx antenna number, and d_A is the distance between Rx antenna k and $k+1$. Since the DFT is a linear operator, the phase shift in equation (2.17), remains in each unique RDM produced by each Rx antenna. Thus, objects can also be separated in angle by performing the DFT over all Rx antennas (angle-DFT) for any given range and velocity in the RDM. An element k_A in the N_A-point angle-DFT is called an angle bin and correspond to the angle θ through

$$k_A \Rightarrow \theta = \arcsin\left(2\frac{k_A - 0.5N_A}{N_A}\right) \tag{2.18}$$

This can be used to estimate the angle-of-arrival (AoA) of an object and to coherently add the signals of the Rx antennas to improve signal-to-noise ratio (SNR).

2.2 Target Detection and CFAR

Constant false alarm rate (CFAR) is a set of algorithms which can be used to detect power level peaks due to a target in the presence of noise, clutter (unwanted re ections from the environment) and interference using adaptive thresholding. In this thesis, detection in the range-velocity map is of interest to nd vibrating objects while avoiding false positives i.e detections due to noise or clutter. For the relevant approaches each element in the RDM, one by one, acts as the cell under test (CUT). In a window around the CUT, training cells are used to estimate the clutter and noise power level to set an appropriate power detection threshold. In Figure 2.2, the window is visualized with the CUT in yellow.

Figure 2.2: CFAR: CUT, Guard and Training Cells.

2.2.1 Cell-Averaging CFAR

Cell-averaging CFAR (CA-CFAR) works by declaring a detection if the power level of the CUT exceeds the average power level of the training cells by some relative amount. Let the average training cell power level be P_{av} and the CUT power level be P_{CUT}. Then, CA-CFAR declares a detection if

$$P_{CUT} > T_{CA} \quad P_{av} \tag{2.19}$$

where T_{CA} is a scalar set by

$$T_{CA} = N(P_{FA}^{1=N} \quad 1) \tag{2.20}$$

in which P_{FA} is the desired false positive rate and N is the number of training cells. The desired false positive rate holds assuming no clutter, no interference from other targets and that all noise is additive Gaussian with zero mean [6].

2.2.2 Angle Detection

After performing CA-CFAR on the RDM, angle-DFTs are performed on the range-velocity pixels where detections were declared. Then, the goal is to find the AoA of the objects corresponding to the detections. In order to do this, the magnitude of a given angle-DFT is normalized such that the maximum peak is 1. A detection is declared at a peak in the angle-DFT if the magnitude exceeds a certain threshold between 0 and 1.

2.3 Data Decoding

This section addresses the procedure for data decoding in a scenario where a bit sequence has been modulated using on-o keying (OOK). Given that a vibra-motor can either vibrate or remain silent, OOK modulation is a natural choice in this context.

We assume that the phase of the signal remains constant during one symbol-time but can change randomly between symbols. This assumption arises because each time the vibra-motor starts vibrating, its phase is not necessarily aligned with the phase of the previous vibration, leading to phase inconsistency in the transmitted signal. Once a vibration starts, it is reasonable to assume that the phase remains stable, as any phase shift would imply a physical lag or time o set created by the vibra-motor, which does not occur. Once the signal is received, it goes through a down-conversion process to get the signal back to base-band. A time synchronization method is utilized to extract the start of the data sequence, followed by a matched lter to increases the SNR. In the nal stage, the signal is sampled and decoded according to an adaptive threshold.

2.3.1 Down-Conversion

We assume that the received signal is of the form:

$$R(t) = L(t) + s(t) \cos(2f_{vib}t + (s(t))) + n(t) \tag{2.21}$$

where $L(t)$ is a low frequency disturbance, $s(t)$ is a modulated train of square pulses according to

$$s(t) = \sum_{k=0}^{N} b_k \, h(t - k \, T) \tag{2.22}$$

where $b_k \in \{0, 1\}$ is the sequence of transmitted bits, N is the length of the sequence, T is the symbol time and $h(t)$ is a square pulse according to

$$h(t) = \begin{cases} 1, & 0 \leq t \leq T \\ 0, & T < t \\ 0, & t < 0. \end{cases} \tag{2.23}$$

The channel attenuation is described by , f_{vib} is the carrier frequency (vibration motor frequency), $n(t)$ is additive white Gaussian noise and $(s(t))$ is a phase shift that depends on $s(t)$. For instance, if $s(t)$ has three square pulses in a row, this means that $(s(t))$ will have the same phase for all three symbol times, but if $s(t)$ starts a new sequence of pulses after having been turned o , then $(s(t))$ will be di erent every time a new sequence of pulses starts transmitting and is constant over one symbol time.

After receiving the signal, it is down-converted by multiplying it with an exponential of the same frequency as the signal, we have

$$R_{down}(t) = R(t)\ e^{j\,2\,f_{vib}\,t} \tag{2.24}$$

$$= L(t)\ e^{j\,2\,f_{vib}\,t} + s(t)\ \left(\frac{e^{j\,(4\ f_{vib}\,t + \ (s(t))} + e^{\ j\ \ (s(t))}}{2}\right) + n(t)\ e^{j\,2\ f_{vib}\,t} \tag{2.25}$$

Note that the low-frequency signal $L(t)$ has been up-converted to a higher frequency, similarly, $s(t)$ now also has a high frequency component. Applying a low-pass filter with a cut-off frequency lower than f_{vib} and multiplying by 2, we have

$$s(t)e^{\ j\ \ (s(t))} + n(t) \tag{2.26}$$

where $n(t)$ represents the remaining noise after low-pass filtering.

2.3.2 Matched Filtering

Given that the phase offset $(s(t))$ for the signal is constant over one symbol time, a matched filter can be applied on equation (2.26). The filter will be the same as $s(t)$, the used pulse in the transmitted signal. This filter is applied through convolution with $h(t)$ and equation (2.26), we have

$$R_{matched}(t) = s(t)e^{\ j\ \ (s(t))}\ h(t) + n(t)\ h(t) \tag{2.27}$$

2.3.3 Packet Synchronization

To decode the bits from the received signal, it is necessary to know the start of the sequence. One common method for time synchronizing signals is by using barker codes, which are binary sequences with the property that they minimize side lobes after auto-correlation and are commonly used in communication and radar systems. They are widely covered in many engineering books that cover wireless communication, see for instance George W. Stimsons book on airborne radar and the use of barker codes [7] for more details. The currently known barker codes can be seen in table 2.1.

length of code	barker code
2	+1, -1
3	+1, +1, -1
4	+1, +1, -1, +1
5	+1, +1, +1, -1, +1
7	+1,+1,+1,-1,-1,+1,-1
11	+1,+1,+1,-1,-1,-1,+1,-1,-1,+1,-1
13	+1,+1,+1,+1,+1,-1,-1,+1,+1,-1,+1,-1,+1

Table 2.1: Table of barker codes

For this thesis, one of these known sequences will be embedded in the transmitted signal. This enables the use of a correlation operation between the signal and barker sequence to nd the index at which it is maximized, which will be the beginning of the data sequence. With this information, it is then possible to sample the signal for decoding. Let the optimal samples for decoding be denoted as $S(k)$, assuming $R_{matched}$ has been perfectly synchronized after the correlation with barker code, we have

$$S(k) = jR_{matched}(k \quad T)j; \; k = 0,1,2... \tag{2.28}$$

where T is the symbol time. Note that since $R_{matched}$ has a non-constant phase rotation, the constellation diagram after sampling the symbols will look like what is shown in gure 2.3. The gure shows a simulation of equation (2.28).

Figure 2.3: Example of a constellation diagram according to equation (2.27), assuming = 1

2.3.4 Threshold Setting

To derive an approach to set the optimal threshold, we start by examining the signal after applying a low-pass lter, the form of the signal is then as in equation (2.26). Due to the random phase shift (t), distinguishing between the symbols requires taking the magnitude of the symbols after matched ltering. As a consequence, the noise distributions for the 1's will now di er from the 0's which should be re ected in the decoding policy. After matched ltering and down-sampling, the magnitude

of the symbols is of the form

$$r(k) = jAe^{j\ (k)}s(k) + z(k)j \tag{2.29}$$

$$s(k)\ 2\ f\ 0; 1g \tag{2.30}$$

$$k = 1; 2; 3::: \tag{2.31}$$

$$z(k)\quad CN(0; 2\ ^2) \tag{2.32}$$

where $CN(m; 2v)$ denotes a complex random variable of mean m and independent real and imaginary part, each Gaussian with variance v. Also, A is some scalar solely dependent on the physical channel, for instance accounting for path loss. Thus,

$$rj(s = 0)\quad Rayleigh(\ _s) \tag{2.33}$$

$$rj(s = 1)\quad Rician(jAj;\ _s) \tag{2.34}$$

Then for a received symbol r, classify r as a 0 if

$$P[r < T\ js = 0] > P[r > T\ js = 1] \tag{2.35}$$

where T is a threshold parameter ful lling the condition. By using the approximation, $Rician(A;\ _s)\quad N(A;\ ^2_s)$, which is valid for large A, the following can be derived

$$P[r < T\ js = 0] = P[r > T\ js = 1] \tag{2.36}$$

$$()\quad \exp\ \frac{T^2}{2\ ^2_s}\ ! = Q\ \frac{jAj\quad T}{\ _s}\ ! : \tag{2.37}$$

where A and $_s$ are estimated from the received samples in the known preamble at the start of the data sequence when implementing this approach and $Q(x)$ is the complement to the normal cumulative density function.

2.4 Tracking and Motion

When a vibra-motor is in motion it may traverse multiple range and angle bins over time. Thus, in order to extract communication data from the vibra-motor, the radar needs to track objects and then extract phase variations at the correct range and angle at the correct time. We propose a tracking algorithm which acts in an iterative manner frame-to-frame and returns trajectories. A trajectory consists of the range and AoA of one object over time. Broadly, the algorithm performs trajectory-to-measurement association, trajectory creation and deletion, and prediction.

2.4.1 Clustering and Closeness

At every frame, a set of points are generated, each containing measured range, velocity and AoA (RVA-points). Due to non-zero resolution in terms of range, velocity and AoA, one object will generate many points relatively close to each other. Thus, to reduce computation and increase accuracy, it is appropriate to cluster points together corresponding to unique objects and then track the clusters.

Although the number of clusters is not known, it is possible to provide a cost function which can determine which points can be considered close to each other. However, merely taking the Euclidean distance between the RVA-points is not appropriate since what is considered close in terms of range is not necessarily close in terms of AoA since they are different quantities with different units and resolutions. Instead we suggest the following cost function between RVA-points i and j

$$L(i,j) = \sqrt{\left(\frac{R_i - R_j}{R_L}\right)^2 + \left(\frac{V_i - V_j}{V_L}\right)^2 + \left(\frac{A_i - A_j}{A_L}\right)^2} \tag{2.38}$$

where R_L, V_L and A_L are tuning parameters. If $L(i,j) \leq 1$ then point i and j are close. As a consequence of the loss function, i and j are not close if the difference in range, velocity or AoA exceed R_L, V_L or A_L respectively which is used to set the parameters.

With the proposed cost function, the algorithm makes use of an altered version of the density-based spatial clustering (DBSCAN) algorithm which is a common unsupervised clustering algorithm. The clustering algorithm is described below where N_i is the set of points close to point i and K_{min} is an input parameter.

1. Initialize $K_{clusters} = 0$, $i = 0$, and all labels as undefined.
2. $i + 1 \rightarrow i$.
3. If label(i) == undefined: proceed. Otherwise: go to step (2).
4. If $|N_i| \geq K_{min}$: proceed. Otherwise: noise $\rightarrow$ label(i), and go step (2).
5. $K_{clusters} + 1 \rightarrow K_{clusters}$.
6. $K_{clusters} \rightarrow$ label(i)
7. Initialize set $S = N_i$.
8. $\forall p$ s.t $p \in N_i$ and label(p) == undefined: If $|N_p| \geq K_{min}$: $S \cup N_p \rightarrow S$.
9. $\forall p$ s.t $p \in S$ and label(p) == noise: $K_{clusters} \rightarrow$ label(p).
10. $\forall p$ s.t $p \in S$ and label(p) == undefined: $K_{clusters} \rightarrow$ label(p).
11. Go to step (2) until all points are labeled.

This algorithm differs from regular DBSCAN mostly due to a slight change in step 8, where the set S at most grows to consider the neighbors of the neighbors of the initial point i. The purpose of this is to reduce the growth of clusters and to simplify the tuning of the parameters in the loss function.

2.4.2 Prediction

To form a trajectory of an object, measurements of the object's state need to be observed over time. However, which measured RVA-cluster belongs to which object is not known. Thus, to perform tracking, predicting where an object in one frame will be in the next frame is an essential step. An extended (non-linear) Kalman filter is used to do this and to perform smoothing on the trajectories. The state variables are radial range R, radial velocity V_R, angle-of-arrival A and azimuthal velocity V_A, where R, V_R and A are measurements.

The state transition model assumes constant velocity and small changes in AoA. From the assumptions, the radial velocity at time $(k+1)$ is modeled, through a small angle approximation $\cos(A_{k+1} - A_k) \approx 1$, to be the same as the radial velocity at time k. The same applies for the azimuthal velocity. Using the same small angle approximation, the range at time $(k+1)$ is modeled to only be dependent on the range and radial velocity at time k. Finally, using the small angle approximation $\sin(A_{k+1} - A_k) \approx A_{k+1} - A_k$, the AoA at time $(k+1)$ is modeled to only be dependent on the range, AoA and azimuthal velocity at time k. After all approximations and assumptions, the state at time $(k+1)$, $x^{(k+1)}$, is dependent on the state at time k, $x^{(k)}$, modeled through

$$x^{(k)} = \begin{bmatrix} R \\ V_R \\ A \\ V_A \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}; \quad x^{(k+1)} \quad f(x^{(k)}) = \begin{bmatrix} x_1 + T x_2 \\ x_2 \\ x_3 + T\frac{x_4}{x_1} \\ x_4 \end{bmatrix} \tag{2.39}$$

where T is the time difference between time $k+1$ and k (frame period time in our case). Moreover, the Jacobian of the state transition function evaluated at state $x^{(k)}$ is

$$F_k = \frac{\partial f}{\partial x}_{x^{(k)}} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ T\frac{x_4}{x_1^2} & 0 & 1 & T\frac{1}{x_1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.40}$$

and the state-to-measurement matrix is

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{2.41}$$

Lastly, using a measurement noise matrix R and process noise matrix Q, the filtering equations can be expressed in two parts. Firstly, the prediction step

$$x_p = f(x^{(k-1)}) \tag{2.42}$$

$$P_p = F_{(k-1)} P_{(k-1)} F_{(k-1)}^T + Q \tag{2.43}$$

and secondly, the update step

$$\text{measured RVA: } z = [R; V_R; A]^T \tag{2.44}$$

$$\text{Residual error: } y = z - H x_p \tag{2.45}$$

$$S = H P_p H^T + R; \quad K = P_p H^T S^{-1} \tag{2.46}$$

$$x^{(k)} = x_p + K y \tag{2.47}$$

$$P_k = P_p - K H P_p. \tag{2.48}$$

These equations allow for smoothing of the trajectories given previous points in the trajectory. To make predictions for every trajectory another prediction step, equation (2.42) and (2.43), is performed using the current filtered state $x^{(k)}$.

2.4.3 Association

One step of the tracking algorithm is to associate measured RVA-clusters in the current frame to existing trajectories. Before this step, all existing trajectories generate RVA-predictions to which the measured RVA-points are compared. In terms of the loss function in equation (2.38), the closest measured RVA-cluster is claimed by a trajectory if it is deemed close i.e less than a certain value L_{max}.

Trajectories are indexed by i and measured RVA-clusters are indexed by j

1. For any trajectory prediction i calculate $L(i, j)\ \forall j$.
2. If $\arg\min_j L(i, j) < L_{max}$. Claim j as a new measurement in trajectory i.

Figure 2.4 exists to further illustrate the prediction and association steps (in two dimensions instead of three). Measured clusters are shown as red dots. The predictions of 3 trajectories are shown as blue stars and the prediction neighborhoods are shown using blue dotted circles (all measured clusters within a prediction neighborhood are deemed closed). In this example all three measured clusters are uniquely associated to all three trajectories since the conditions are met.

Figure 2.4: Prediction and Association Visualization

2.4.4 Creation and Deletion

The next step is to create new or delete existing trajectories. First, all trajectories which did not receive a new measurement in the association step are looped over and deleted. To not remove trajectories unnecessarily, trajectories over a certain length (number of measurements) are allowed to make blind predictions i.e use the prediction made at time k as the measurement at time $(k + 1)$. If the number of

consecutive blind predictions exceed a certain number the trajectory is deleted.

To create new trajectories, all RVA-clusters which were not associated to an existing trajectory are looped over instead. The measured RVA-cluster is set as the first filtered value $x^{(0)}$ and the covariance matrix P_0 is initialized.

2.5 Clutter Elimination and Circle-Fitting

From equation (2.6), if an object in free space is in range bin i while moving or vibrating, the complex samples of the range-DFT in a short time interval will approximately be on a circle in the complex plane. However if stationary scatterers is near range bin i the complex samples $g_i(t)$, instead can be described by

$$g_i(t) \quad {}_v \exp\left[j \left(\frac{2 d_v(t)}{c} \right. \right. \quad {}_v \left. \left. \right] + {}_c \exp[j \quad {}_c] \right. \tag{2.49}$$

where ${}_v$; ${}_c$; ${}_v$ and ${}_c$ are real constants and $d_v(t)$ is time-varying range of the moving or vibrating object. This corresponds to the samples being placed on a circle not centered on the origin in the complex domain which is illustrated in Figure 2.5. Thus, if information is encoded in the phase of the vibrating object, the measured signal will be distorted by the stationary scatterers. To alleviate this problem, it is possible to fit a circle to the measured complex data and then remove the center of the fitted circle which should restore the signal.

(a) No Scatterer (b) Scatterer

Figure 2.5: Clutter Illustration

To fit a circle to a set of complex data points we have developed a new algorithm inspired from an iterative process specified in [8] which uses initial guesses for the circle center and radius. The proposed changes were made to mainly to improve the

reliability of the algorithm through a more thorough method of setting the initial guesses for the center and radius. Figure 2.6a to 2.6d illustrates the algorithm in 4 steps.

(a) Step 1

(b) Step 2

(c) Step 3

(d) Step 4

Figure 2.6: Circle-Fitting Algorithm Steps

Step 4 in Figure 2.6d, is performed by using the pair of update equations, displayed in equation (2.50) and (2.52), which iteratively ts a circle to the translated data.

To understand the equations, some notation and initial settings is shown below.

Radius after iteration k: r_k
Center after iteration k: $T_k e^{j a_{best}}$
$c_x = \cos(a_{best})$
$c_y = \sin(a_{best})$
Complex mean of original data: $_D$
Translated data point i: $\tilde{d}_i = \tilde{x}_i + j\tilde{y}_i$
Regularization Term:

$$r_k = \frac{1}{N}\sum_{i=1}^{N} |\tilde{d}_i - T_{k-1} e^{j a_{best}}| = f_r(T_{k-1}) \tag{2.50}$$

$$T_k = \frac{1}{N + } \sum_{i=1}^{N} \left[c_x \tilde{x}_i + c_y \tilde{y}_i - r_k \sqrt{\frac{c_x(\tilde{x}_i - T_{k-1}c_x) + c_y(\tilde{y}_i - T_{k-1}c_y)}{(\tilde{x}_i - T_{k-1}c_x)^2 + (\tilde{y}_i - T_{k-1}c_y)^2}} \right] \tag{2.51}$$

$$= f_T(r_k; T_{k-1}) \tag{2.52}$$

The fourth step of the circle- tting algorithm can then be described with the instructions below wheres and i_{max} are input parameters controlling the stopping criterion and c_D is the circle center of the original untranslated data.

1. Initialize r and T: $0.6R \to r$, $0.5R \to T$
2. Initialize $0 \to i$
3. $i + 1 \to i$
4. $f_R(T) \to r_{new}$
5. $f_T(r_{new}; T) \to T$
6. If $|r_{new} - r| < s\ r$ OR $i \geq i_{max}$: proceed. Otherwise $r_{new} \to r$ and go to step (3).
7. Calculate the circle center of the original data: $_D + (R + T)e^{j a_{best}} \to c_D$

2.6 Motion Suppression

As mentioned previously in equation (2.5), in a given range bin, any small range change in said bin will result in a phase change. Thus, if the vibra-motor is moving, phase changes will be due to both the movement and the vibrations according to

$$(t) \quad _0 + \frac{4}{}v(t)\ t + A\cos(!_{vib}t) \tag{2.53}$$

where $_0$ and A is are constants and $v(t)$ is the time varying velocity. One issue is that the phase changes due to velocity can be signi cantly larger than the phase changes due to the vibrations. This e ect is illustrated in Figure 2.7a, where the vibrations are hardly visible. Consequently, the goal is to remove the phase changes caused by the velocity and keep the phase changes caused by the vibrations. By

modelling the velocity as locally constant, this can be achieved by taking the difference between consecutive elements, approximating the first order derivative, like

$$\Delta\phi(t) \approx \frac{4}{\lambda} v \, A\omega_{vib} \sin(\omega_{vib} t) \tag{2.54}$$

and then apply a finite impulse response digital band-pass filter. The result is shown in Figure 2.7b.

(a) No Motion Suppression (b) Motion Suppression

Figure 2.7: Phase in Motion

2.7 Trajectory Stitching

The phase progression data is collected frame by frame and there may be large phase jumps between frames especially after clutter elimination. To counteract this, phase signals are stitched together frame by frame as in Figure 2.8. Let $p_i(k)$ be the kth phase sample from frame i, and let N be the number of samples per frame. The stitching process is described in the instructions below where K is an input parameter.

1. Initialize $0 \rightarrow c_2$
2. Initialize $0 \rightarrow i$
3. $i + 1 \rightarrow i$
4. Find s_1 and c_1 which minimizes $\sum_{k=1}^{K} [p_i(k) - (s_1(k-1) + c_1)]^2$
5. $\forall k \in \{1, 2, \ldots, N\}$: $p_i(k) - c_1 + c_2 \rightarrow p_i(k)$
6. Find s_2 and c_2 which minimizes $\sum_{k=N-K+1}^{N} [p_i(k) - (s_2(k-(N-K+1)) + c_2)]^2$
7. $p_i(N) + s_2 \rightarrow c_2$
8. Repeat from step (3) until all frames have been stitched.

(a) No Stitching (b) Stitching

Figure 2.8: Trajectory Stitching

2.8 BER and Statistical Signi cance

Due to slow bit rates, relatively few bits are used to measure the bit error rate (BER) in the results, making the empirical results unreliable. However, through some probability calculations it is possible to obtain an upper bound of the BER given the measurement with an arbitrary level of certainty. By viewing the BER as a random variable$x$ it is possible to construct said upper bound through solving

$$P[x \quad P_b | N_c; N_e] = C \tag{2.55}$$

where N_e is the number of erroneous bits, N_c is the number of correctly decoded bits, P_b is the BER upper bound and C is the con dence level. To construct it, the posterior distribution of the BER is calculated from Bayes' theorem through

$$p(x|N_e; N_c) = \frac{p(N_e; N_c|x)p(x)}{p(N_e; N_c)}: \tag{2.56}$$

If the likelyhood, $p(N_e; N_c|x)$, is assumed to be binomial (independent bit ips) and and the prior distribution $p(x)$ is assumed to be at (a prior belief that all possible values of$x$ are equally likely), then the posterior has a closed form solution in the form of

$$p(x|N_e; N_c) = \frac{(N_e + N_c + 2)}{(N_e + 1)(N_c + 1)} x^{N_e}(1 \quad x)^{N_c} \tag{2.57}$$

where $(\ )$ is the gamma function. Consequently, to construct an upper bound with 95 % certainty, we computationally solve forP_b the following:

$$\int_{x=0}^{P_b} p(x|N_e; N_c)dx = C = 0:95 \tag{2.58}$$

The calculatedP_b will be presented along the results as 95 % BERUB.

2.9 Summary

The theory section has covered the necessary background knowledge needed to understand the thesis, ranging from FMCW-radar basics and sensing, to communication processing and tracking algorithms. Through equation (2.6), we showed that a single object with su ciently small vibrations will not be visible in the range map but can indeed show up in the phase of the received chirp signal, hinting at the possibility of using the received phase for data encoding. Common sensing algorithms like cell-averaging CFAR were covered, which can be applied on the received range data from the radar to detect objects relative to surrounding clutter. The communication-decoding chain for a signal of the form (2.21) has been covered, which will later be used to decode the received phase data from the vibration motor. Finally, a tracking algorithm was proposed and covered for the case when the vibration motor is moving.

3
Method

3.1 Aim and Limitations

This section aims to provide an understanding of how the theory presented in the previous section is implemented with a real system, including which hardware components are used. We start by examining the building blocks of the system such as the vibration motor and the radar, we then split the section into two parts, one for the purely stationary case, and one for the moving case. The idea is that it is easier to extend the algorithm when it works for a simpler case, therefore solving one issue at a time. This involves going through the measurement setup and digital signal processing (DSP) chain for both cases.

This thesis will be using the following hardware components:

- Texas Instrument AWR1642BOOST radar chip.
- One vibration motor, in this case the vibra motor 4 click from microe.
- The Arduino Uno Rev 3 SMD

3.2 Vibration motor

To create a JCAS system, a way to transmit data is necessary, for this thesis, a vibration motor is used to perform this role. The motor starts vibrating once it gets an electrical start signal which can be controlled using software. An Arduino Uno board sends a signal to the motor to vibrate according to a prede ned pattern encoded in software. The modulation scheme used is on-o keying meaning that the transmitter transmits a bit value of 1 when it vibrates and a 0 when silent. The frequency of the vibra-motor is around 230 Hz.

The bit sequence used as a demonstration is a 13 bit barker code from Table 2.1, followed by 7 data bits. Once this 20 bit sequence is transmitted, it immediately starts over, creating a continuous vibration pattern that constantly transmits data.

3.3 Radar

The radar used in this thesis is an FMCW radar from Texas instruments operating in the 77 GHz region. This component has two purposes in the JCAS system, sensing, and receiving data. The crucial detail with regards to communication is that the radar needs to be capable of detecting the vibrations of the motor. The FMCW radar from Texas Instruments can be con gured through the software mmWave Studio to set the value of parameters such as chirp rise time, bandwidth of the transmitted signal etc. A table of the important parameters and what they mean is listed in Table 3.1. The table also includes the values used throughout the thesis which are kept the same during every measurement unless otherwise speci ed to introduce consistency in the results.

Parameter	Explanation	Value
Ramp time	Time it takes from that the chirp starts sending out a signal until it ends.	300 s
Idle time	Cool down period before each new ramp time.	300 s
Chirp time	Ramp time + idle time.	600 s
Frame periodicity	Time between each frame.	153.6 ms
Chirp slope	Derivative of the chirp signal with respect to time(MHz/ s).	13.325 Mhz/s

Table 3.1: Explanation for Con guration Parameters for FMCW Radar

3.4 Measurement Setup (Stationary Case)

Figure 3.1 shows the measurement setup for the stationary case, the radar is in the bottom of the picture and the vibration motor is enclosed in a white case, placed at the same height as the radar. The casing ensures that the vibration motor is kept steady and protected, and the tri-pod makes sure that the radar and vibra motor are kept at the same height. Figure 3.2a and 3.2b show a closer look at the schematic for the case and the vibration motor while Figure 3.3 shows the real case with motor.

Figure 3.1: Measurement Setup Stationary Case