Multiple-input multiple-output radar for drone detection and localisation

A project-oriented study at Nordic Wing

Aske Barnaby Rasborg Best

AALBORG UNIVERSITY

Abstract

The growing use of drones, particularly small and cost-effective suicide drones, in modern warfare poses substantial threats to military assets. The purpose of this project is to contribute to further developing the design of the Archangel system: a drone detection and localisation system. This project applies a radar and a camera to acquire spherical measurements of a drone. The multiple-input multiple-output radar principle is applied to obtain azimuth angle measurements in addition to range and velocity measurements. Four different frameworks: angle FFT, Bartlett and Capon beamformers, and the MUSIC algorithm are investigated for direction-of-arrival estimation. Multiple data collections have been conducted in environmental conditions to evaluate the performance of the radar target detection. A data fusion scheme has been developed using a Kalman filter and spherical-to-Cartesian transformed radar and camera measurements to improve the drone location estimations. The Kalman filter effectively combines the multi-sensor measurements from the radar and camera, reducing measurement noise, and successfully models the linear movement of the drone. The method produces reliable estimations of the location and velocity of the drone. The fundamental advantages and limitations of the methods have been identified. This study further lays the groundwork for designing an effective drone detection and localisation system.

Titlepage

Title:
Multiple-input multiple-output radar for drone detection and localisation

Theme:
Radar, MIMO radar, target detection, CA-CFAR, data fusion, Kalman filter, tracking

Project Period:
2nd of September 2024 to 6th of January 2025

Project Group:
ES9-921a

Author:
Aske Barnaby Rasborg Best

abest23@student.aau.dk

Supervisors:
Anders Malthe Westerkam
Troels Pedersen

Date of Completion:
6th of January 2025

Electronics and Information Technology
Institute of Electronic Systems
http://www.aau.dk

Preface

The mathematical notation used in this report is described here. Generally, scalar values will be denoted by lower-case letters such as x. Vectors are denoted by bold lower-case letters such as $\mathbf{x}$, and all vectors are assumed to be column vectors. A superscript T denotes the transpose of a matrix or vector so that $\mathbf{x}^T$ will be a row vector. The Hermitian transpose (conjugate transpose) is denoted as H. Upper-case bold letters, such as $\mathbf{M}$, denote matrices. Variables will throughout the report be denoted in italic font. The units employed in this report are based on the SI, except angle measurements, which are expressed in degrees and denoted by the symbol [°]. Some notes on additional comments on the text formatting follow. Specific components relating to sensing, software, and programming functions will be denoted as: `arbitrary text`.

Relating to the data processing and presentation thereof, a data point having zero value should be regarded as a 'no measurement'. The term UAV is commonly used as a scientific notation for fixed-wing aircraft, quadcopters, helicopters, and other similar devices. In this report, the term *drone* is used to refer to such devices.

All code relating to the project and to producing the obtained results is appended to the project report. The obtained video illustrating the results is also appended. For those interested in accessing the data, code, and video material, please feel free to reach out. To obtain these materials, contact at `bestaske@gmail.com`

Abbreviations

FFT fast fourier transform

MIMO multiple-input multiple-output

ULA uniform linear array

CA-CFAR cell-averaging constant false alarm rate

MUSIC multiple signal classification

DOA direction-of-arrival

TX transmitter

RX receiver

VX virtual

SIMO single-input multiple-output

YOLO you only look once

FOV field of view

LNA low-noise amplifier

ADC analogue-to-digital converter

TDM time division multiplexing

FMCW frequency-modulated continuous-wave

PCB printed circuit board

RMSE root mean squared error

Acknowledgements
A huge thank-you to Nordic Wing and Jonas Dyhr for providing the opportunity for a project-oriented study with them and facilitating all the equipment necessary for the execution of the project. Special thanks to supervisors Anders Malthe Westerkam and Troels Pedersen for their support and direction throughout the project.

Contents

1 Introduction

In the Russo-Ukrainian war, cheap drones have become a key weapon, used for both surveillance and attacks. The war has highlighted a new threat on the modern battlefield, as retrofitted hobby drones, or suicide drones, can destroy high-value targets like tanks, multiple launch rocket systems, and radio stations at a low cost. Detecting, localising and countering these drones is an urgent matter in this scenario. Nordic Wing is pursuing a solution capable of effectively detecting and neutralising such drone threats. Currently available technologies are challenged by the harsh milieu. Additionally, they themselves are expensive and become high-value targets. The cost factor underscores the importance of deploying a cost-effective solution to mitigate the drone threats effectively. The solution: the Archangel system.

The rest of this chapter includes a description of Nordic wing, earlier project work providing the foundations for the current project, the project boundaries, and the problem proposal. Chapter 2 introduces the MIMO radar principle. Chapter 3 presents the array signal model along with the four methods applied for angle estimation: the angle FFT, the Bartlett and Capon beamformers, and the MUSIC algorithm. Chapter 4 establishes the radar target detection composed of a presentation of the physical radar applied in the project, the implementation, a description of the data collection, and the corresponding results. Chapter 5 presents the Kalman filter used as a data fusion scheme and the final results of this project. The results are discussed in Chapter 6, and conclusions are drawn in Chapter 7.

1.1 Description of Nordic Wing

Nordic Wing [1] is a Danish drone manufacturer of fixed-wing drones for mapping and surveying, defence and security, and search and rescue among other purposes. The company is located at the old military air force base in Værløse, Denmark. Nordic Wing produces the Astero drone, displayed in figure 1.1.

Figure 1.1: The Astero drone

The company is a growing organisation which currently employs around 40 people across various roles, competencies, and skills. The company departments include R&D division, production and service, and management. The R&D division is composed of three areas: mechanics, hardware, and software. In addition to the Astero drone, the company is exploring other potential spin-off products.

1.2 Previous work and foundations for current project

This project is a continuation of the project: *Detection and localisation of drone for use in anti-suicide drone system* [2] carried out in the spring of 2024. The previous work investigated the feasibility of various sensor technologies for detection and localisation of drones. The combination of radar and camera technology was evaluated as the most suitable sensors to be applied in the Archangel system based on the specification and requirements, seen in appendix C. The project utilised the mmwave `AWR1843` radar and a `GoPro MAX`

360 to obtain the spherical state estimates. Different target detection schemes were employed: the CA-CFAR algorithm was applied to the radar to obtain range and velocity estimates of the drone. The YOLOv8 was evaluated as the most effective object detection model, providing angle estimates in both the horizontal and elevation planes. The state estimates were produced by the data fusion scheme implemented as a Kalman filter. The Archangel system is conceptualised in figure 1.2 with the spherical and Cartesian coordinate systems employed in the project.

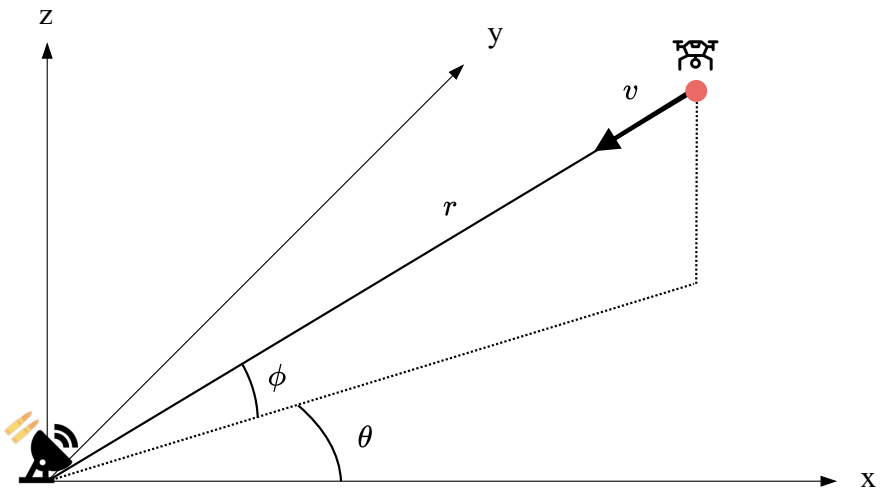


Figure 1.2: Proposed Archangel system illustration displaying the coordinate system configuration employed in the project. The sensor boresight follows the x-axis.

The sensor boresight follows the x-axis and the azimuth (horizontal) and elevation angle are denoted as θ and ϕ, respectively. The previous project was preliminary and served as an initial investigation into the feasibility of the Archangel system. Several improvements were discussed for further development. These were related to all aspects: radar, camera, and data fusion. They included the following, which will be the focus points of this project: applying the MIMO principle enabling angle estimates from the radar in addition to the camera, yielding multi-sensor measurements greatly improving the state estimates of the data fusion method. Additionally, the camera drone detection was applied on a wide-angle video format, severely affecting the effectiveness of the drone identification using the camera. Furthermore, the measurements of the motion of the drone were spherical,

yielding deviation in the resulting estimates when applied to the linear Kalman filter. This project continues the work with FMCW radar employing the same radar. A revised approach is encouraged.

1.3 Problem boundaries and problem proposal

Certain constraints are introduced to simplify the problem of drone detection, creating a realisable baseline to study the effectiveness of the applied sensors and methods. The constraints are with respect to drone, environment and detection parameters. They are presented as follows:

- Drone type and number: the sole focus will be on detecting and localising a single quad-copter drone.
- Static/dynamic: this project will focus exclusively on the Archangel system operating in a stationary mode, thereby disregarding a dynamic scenario.
- Detection parameters: this project will disregard the requirements for detection range, velocity and FOV. The focus is on evaluating the performance with the available equipment in the context of the Archangel system.

The project is a collaboration between Aalborg University and Nordic Wing. The project work is conducted at the facilities at Nordic Wing as a project-oriented study and with supervision from Aalborg University. Aalborg University and Nordic Wing intend to address the detection and localisation problem by data fusion, utilising a radar and a camera. Consequently, a problem proposal has been formulated to address this specific challenge:

The object of this study is to investigate the implementation of a data fusion strategy for detection and localisation of a drone utilising a combination of radar and camera technologies.

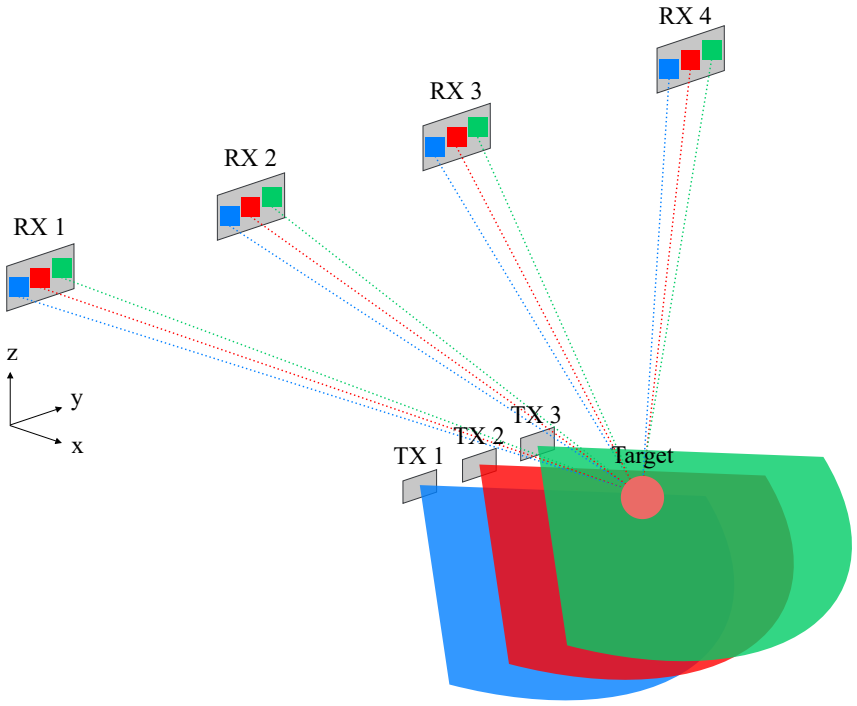
2 | The MIMO radar principle

This chapter introduces the multiple-input multiple-output (MIMO) radar principle utilised to obtain angle estimates of a target location from a radar. Following this, a time division multiplexing as a multiplexing strategy for separation of channels is presented.

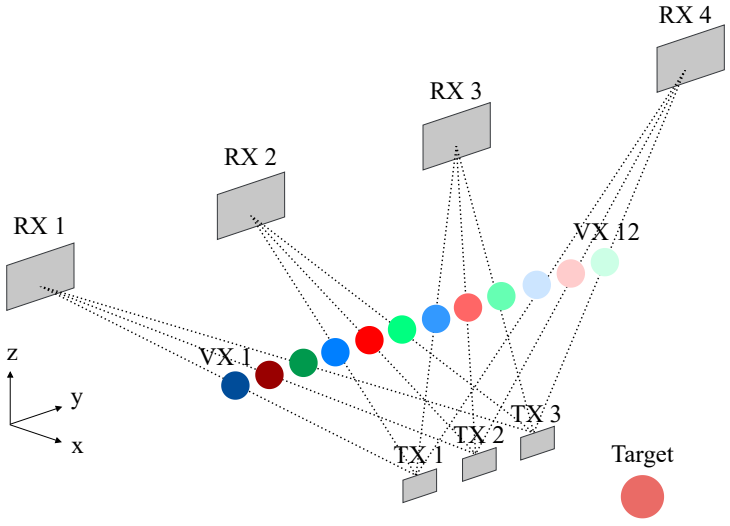
MIMO radar technology is an advanced radar approach that leverages multiple transmitter (TX) and receiver (RX) antenna elements to enhance radar performance across various dimensions. Unlike traditional radar systems, which typically use a single antenna or a phased array to transmit and receive signals, MIMO radar employs multiple independent antennas for both transmission and reception. This configuration enables unique signal processing techniques that can significantly improve spatial resolution, target detection, and accuracy. An MIMO radar with N_{TX} transmitter elements and N_{RX} receiver elements, has the equivalent angle resolution of a SIMO radar with $N_{\mathrm{TX}}N_{\mathrm{RX}}$ receiver antennas. Therefore the MIMO radar, with fewer physical antenna elements requiring individual processing chains, each including an LNA, mixer, filter, and ADC, provides a cost-effective solution for enhancing radar angle resolution. This technology plays a crucial role in improving spatial resolution [3].

The MIMO radar principle utilises the multiple TX and RX elements to create a virtual array. This principle is illustrated in figure 2.1. The same set of RX elements process signals from transmission from multiple TX elements, i.e. each RX antenna receives the backscattered signal from all TX elements. Provided that the RX elements are able to separate the signals corresponding to different TX elements, e.g. by having the TX elements transmit on orthogonal channels, a virtual array is created. Each virtual (VX) element is related to the unique pathways between each TX-RX element pair. The number of VX elements is equal to $N_{\mathrm{VX}} = N_{\mathrm{TX}}N_{\mathrm{RX}}$. The target introduces a phase change between the virtual antenna array elements related to the angle of the target and the physical placement of the antenna elements. This setup allows for superior angular resolution and increased clutter suppression [3].

Different multiplexing strategies can be applied to provide the separation of the TX signals at each RX element. One method is time division multiplexing which is described in the following section in the context of FMCW radars.



(a)



(b)

Figure 2.1: Figure (a) illustrates the MIMO radar principle for antenna array of 3×4 TX and RX antenna elements. Each RX element receives the backscattered signal from each of theTX elements. Figure (b) illustrates the synthesis of the virtual array from the formed pathways. Contrast indicating unique pathways formed.

2.1 Time division multiplexing

Time division multiplexing (TDM) is the simplest strategy to provide separation between the signals and is easily implemented. Each frame comprises several blocks, with each block containing N_{TX} time slots, each corresponding to the transmission by one of the N_{TX} TX antennas. The signal is transmitted from one TX antenna at a time, alternating between the elements. This scheme is illustrated in figure 2.2.

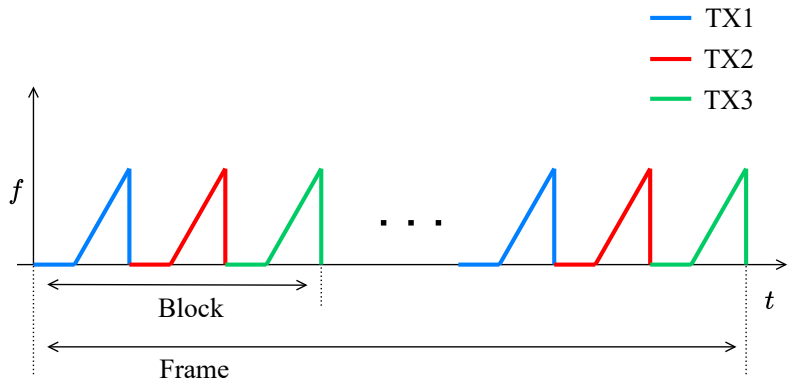


Figure 2.2: TDM illustration using three TX antenna elements.

The figure illustrates TDM applied for an FMCW radar with $N_{\text{TX}} = 3$. This approach ensures no interference between signals at the receivers from different transmitters. Thereby, achieving orthogonality in time [3].

3 | Direction-of-arrival estimation

MIMO radar enables high-resolution direction-of-arrival (DOA) estimation by synthesizing a virtual array, allowing for much higher resolution than that created by the physical antenna elements alone. This chapter presents the signal model for a uniform linear antenna array. Following this the four techniques for DOA estimation applied in the project are presented.

3.1 Array signal model

There are different configurations of a virtual antenna array. This section will present the configuration of a uniform linear array (ULA). The array of M antenna elements are uniformly spaced on a line, depicted in figure 3.1. The elements are spaced distance d apart. Making a far-field assumption, the signal from the target is impinging as planar waves on the array at an angle θ, measured (counter-clockwise) with respect to the normal of the antenna array [4].

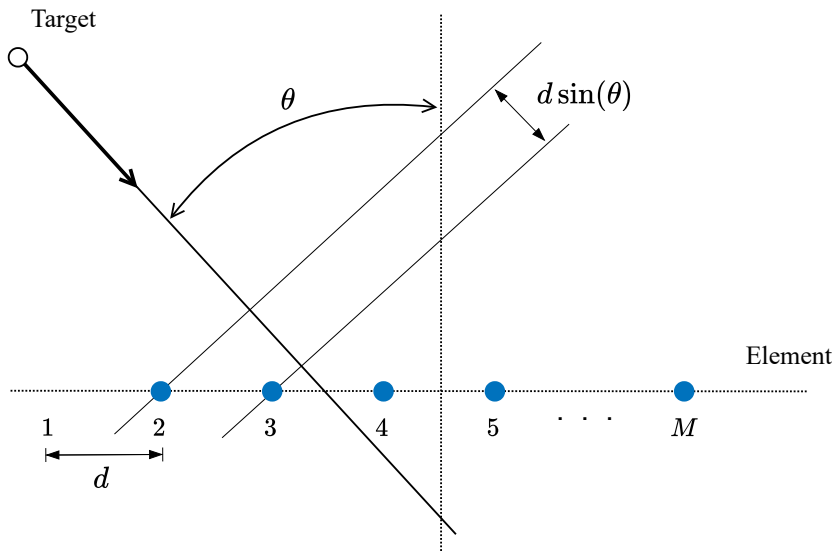


Figure 3.1: The uniform linear array

The antenna array signal model is based off the relative time-delay τ_m across the array

elements, and is written as [4]

$$\tau_m = (i-m)\frac{d\sin(\theta)}{c}, \qquad \text{for } \theta \in [-90°, 90°] \tag{3.1}$$

where $m \in \{1, 2, \ldots, M\}$ and c denotes the propagation velocity of electromagnetic waves, i.e. the speed of light. The phase shift due to τ_m across each antenna element for a target at θ can be represented by the steering vector $\mathbf{a}(\theta)$, i.e. the steering vector models the expected phase shifts across the array elements for a plane arriving at θ. The steering vector is given as [4]

$$\begin{aligned}\mathbf{a}(\theta) &= \begin{bmatrix}1 & e^{-j\omega_c\tau_2} & \ldots & e^{-j\omega_c\tau_m}\end{bmatrix}^T \\ &= \begin{bmatrix}1 & e^{-j\omega_{s_2}} & \ldots & e^{-j\omega_{s_m}}\end{bmatrix}^T,\end{aligned} \tag{3.2}$$

where $\omega_c = 2\pi f_c$ denotes the angular carrier frequency and ω_{s_m} represents the spatial frequency at element m, defined as [5]

$$\omega_{s_m} = (m-1)\frac{2\pi d\sin(\theta)}{\lambda}. \tag{3.3}$$

The relationship between ω_s and θ is non-linear; ω_s is most sensitive to change at $\theta = 0$ and the sensitivity decreases as θ increases. This means that the estimation of the DOA is more error prone as θ increases. Intuitively, the increase of θ makes the apparent area of a patch antenna seem larger, reducing the effective received power from a target, thereby reducing the ability of the radar to detect efficiently [5].

Spatial aliasing, analogous to temporal aliasing in signal processing, can lead to ambiguities in DOA estimation, resulting in multiple indistinguishable directions. To avoid this consideration should be given to the spacing between elements in the ULA. The steering vector $\mathbf{a}(\theta)$ is uniquely defined if $|\omega_s| \leq \pi$. This, combined with the desire for the requirement of d to hold for any θ, leads to the condition [4]

$$d \leq \frac{\pi}{2}, \tag{3.4}$$

meaning that the spatial sampling (or antenna spacing) should not exceed $\pi/2$. Two targets at locations symmetric to the array line will produce the same phase shift. The restriction of θ to be within the interval $[-90°, 90°]$ is a limitation by the ULA. In a practical scenario this ambiguity is eliminated for patch antenna arrays as they only pass signals that lie within this interval.

The angle resolution is the minimum angle separation for two objects to appear as separate peaks. The resolution is given as [4]

$$\Delta\theta \simeq \lambda/\text{array length}. \tag{3.5}$$

The following model equation is presented to describe the system behaviour, capturing the key dynamics. Denoting the signal corresponding to the target at time index t as $s(t)$. Collecting multiple samples of the target signal $s_k(t)$ at the same time index, i.e. producing multiple snapshots, constructs the signal vector given as

$$\mathbf{s}(t) = \begin{bmatrix} s_1(t) & s_2(t) & \dots & s_K(t) \end{bmatrix}, \tag{3.6}$$

where $k \in \{1, 2, \dots, K\}$ denotes the snapshots. The superposition principle can be applied, such that the output signal of the array can be written as [4, 6]

$$\mathbf{X} = \begin{bmatrix} \mathbf{a}(\theta_1) & \mathbf{a}(\theta_2) & \dots & \mathbf{a}(\theta_N) \end{bmatrix} \begin{bmatrix} \mathbf{s}_1(t) \\ \mathbf{s}_2(t) \\ \vdots \\ \mathbf{s}_N(t) \end{bmatrix} + \mathbf{N} \tag{3.7}$$

$$= \underset{M\times N}{\mathbf{A}} \cdot \underset{N\times K}{\mathbf{S}} + \underset{M\times K}{\mathbf{N}}, \tag{3.8}$$

where $n \in \{1, 2, \dots, N\}$ is the number of targets (or sources) and $\mathbf{A}$ is the Vandermonde matrix of steering vectors. $\mathbf{N}$ is the noise present, which may be considered as Gaussian white noise.

3.2 DOA methods

Four different DOA estimation methods are considered in this project. The methods are the angle FFT method, the Bartlett and Capon beamformers, and the MUSIC algorithm. These methods constitute some of the most popular methods for DOA estimation. An illustrative example of using the estimators is displayed in figure 3.2 supporting the theoretical foundation of the methods presented in the following.

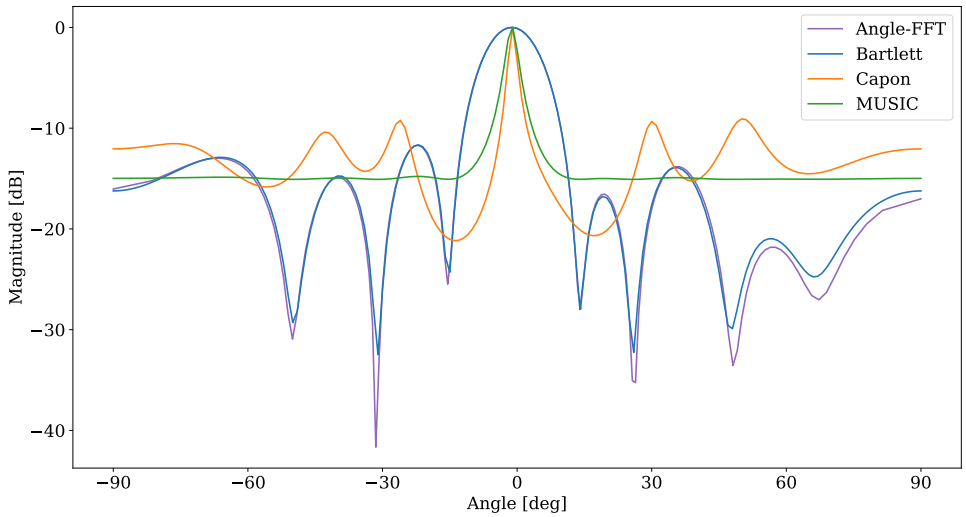


Figure 3.2: Normalised spectrum of the DOA estimation methods for a target at approximately $\theta = 0°$ in the `radar_angle_0` described in appendix B.

Angle FFT

The angle FFT is an extension to the range FFT and Doppler FFT as introduced in [5]. For each frame a 2D FFT is obtained from each antenna element in the ULA. In each 2D FFT there will be a peak for each target at the same location. An FFT on the sequence of phasors corresponding to the 2D FFT peaks resolve the target. This is called an angle FFT. The angle FFT will have a peak at a frequency ω relating to the DOA of the target. The frequency peak correspond to the spatial frequency defined in (3.3) and the angle is determined in relation herefrom as [5]

$$\theta = \sin^{-1}\left(\frac{\lambda\omega}{2\pi d}\right). \tag{3.9}$$

Bartlett beamformer

Common for the Bartlett and Capon beamformers and the MUSIC algorithm is that they apply the covariance matrix $\mathbf{R}$. In practice the covariance matrix is estimated by computing the sample covariance matrix $\hat{\mathbf{R}}$ from the received signal as [4]

$$\hat{\mathbf{R}} = \frac{1}{K}\mathbf{Y}\mathbf{Y}^H \tag{3.10}$$

and captures the spatial correlation across the antenna array. The Bartlett beamformer is the conventional beamformer and is a non-parametric method, since is does not make any assumptions of the covariance structure of the data. For a general spatial filter design problem the objective is that the filter

(i) passes undistorted the signals with a given DOA θ; and

(ii) it attenuates all the other DOAs different from θ as much as possible.

The power of the beamformer is calculated as [4]

$$P(\theta) = \mathbb{E}\left[|\mathbf{y}|^2\right] = \mathbf{w}^H\hat{\mathbf{R}}\mathbf{w}, \tag{3.11}$$

where $\mathbf{w}$ is the filter coefficients. Hence, $\mathbf{w}^H\hat{\mathbf{R}}\mathbf{w}$ should peak at the DOAs of the sources located in the viewing field of the array. The objectives (i) and (ii) leads to a mathematical formulation of constraints in the design problem. Utilising this, it can be shown that $P(\theta)$ maximises when $\mathbf{w} = \mathbf{a}(\theta_s)$ for a source located at θ_s. Therefore, the Bartlett beamforming DOA estimates are given by the location of the n highest peaks of the function [4]

$$P_{\text{Bartlett}}(\theta) = \mathbf{a}^H(\theta)\hat{\mathbf{R}}\mathbf{a}(\theta). \tag{3.12}$$

The Bartlett beamforming method is a direct spatial extension of the periodogram. The Bartlett beamformer is simple to implement and provides a reasonable performance. However, its performance degrades quickly for correlated signals and a few number of snapshots when estimating the covariance matrix. Furthermore, it has a poor angular resolution.

The method serves as a foundational method for more advanced methods like the Capon beamformer and MUSIC algorithm, which will be presented in the following.

Capon beamformer

For the Capon beamformer the objective is accomplished in a more sound way; the filter actively tries to minimise the output power when fed with the actual array data $\mathbf{y}$. In the Capon approach the objective (ii) is designed in a data dependent way alternatively to the Bartlett beamformer where this was done independently of the data. As a consequence the goal of the Capon filter, when steered toward a specific direction θ, is to suppress signals arriving from any DOA $\neq \theta$. In contrast the Bartlett beamformer gives equal attention to all DOAs $\neq \theta$, even if no signals are actually incoming from many of the DOAs. The Capon DOA estimates are obtained as the locations of the n largest peaks of the function [4]

$$P_{\text{Capon}}(\theta) = \frac{1}{\mathbf{a}^H(\theta)\hat{\mathbf{R}}^{-1}\mathbf{a}(\theta)}. \tag{3.13}$$

The Capon beamformer actively suppresses the $\theta \neq$ DOA. By effectively suppressing the interference and noise from other directions the Capon beamformer provides a better resolution. However, the method requires the inversion of the sample covariance matrix. Therefore, $\hat{\mathbf{R}}$ needs to be invertible. Furthermore, the inversion can be computationally expensive for large arrays or poorly conditioned matrices. The performance is also highly dependent on accuracy of the covariance matrix estimate, requiring a sufficient number of snapshots for the estimation. Additionally, the method struggles with correlated signals.

MUSIC

The multiple signal classification (MUSIC) algorithm is a parametric method and is a popular technique used for estimating the DOA. The MUSIC method is derived from the covariance model with $M > N$ [4]. It is based on the eigenstructure of the covariance matrix of the signal. The eigenstructure of $\mathbf{R}$ contains complete information on the DOAs. Given the estimate of the covariance matrix $\hat{\mathbf{R}}$ the MUSIC algorithm estimates the DOA using an eigenspace method. Considering the eigendecomposition of $\hat{\mathbf{R}}$ as [4]

$$\hat{\mathbf{R}}\mathbf{U} = \mathbf{U}\boldsymbol{\Lambda}, \tag{3.14}$$

where $\boldsymbol{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_M)$ which is arranged in a non-increasing order and $\mathbf{U}$ contains the corresponding orthonormal set of eigenvectors. The set of eigenvectors of $\hat{\mathbf{R}}$ can be split into two subsets: one related to the source (target) $\mathbf{U}_s$ and one related to the noise $\mathbf{U}_n$. The associated eigenvectors poses some interesting properties that can be used for angle estimation. Expressing the sample covariance matrix in terms of the subsets as

$$\hat{\mathbf{R}} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^H = \begin{bmatrix}\mathbf{U}_s & \mathbf{U}_n\end{bmatrix}\begin{bmatrix}\boldsymbol{\Lambda}_s & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda}_n\end{bmatrix}\begin{bmatrix}\mathbf{U}_s^H \\ \mathbf{U}_n^H\end{bmatrix} \tag{3.15}$$

$$= \mathbf{U}_s\boldsymbol{\Lambda}_s\mathbf{U}_s^H + \mathbf{U}_n\boldsymbol{\Lambda}_n\mathbf{U}_n^H, \tag{3.16}$$

where the suffixes s and n relates to the N sources and $M - N$ noise components, respectively. The eigenvalues are conditioned as

$$\begin{cases}\lambda_k > \sigma^2 & \text{for } k = 1, \ldots, N \\ \lambda_k = \sigma^2 & \text{for } k = N+1, \ldots, M\end{cases}$$

where σ^2 denotes the variance of the noise present in the signal. The eigenvalues correspond to the variability along different directions, with the largest eigenvalue indicating the direction of maximum variability. The subset $\mathbf{U}_s$, associated with the N largest eigenvalues, spans the signal subspace $\mathcal{U}_s$, which represents the directions of greatest variability. The remaining $M - N$ eigenvectors span the noise subspace $\mathcal{U}_n$. The two subspaces are orthogonal to each other $\mathcal{U}_s \perp \mathcal{U}_n$ [4, 6].

The MUSIC algorithm exploits the originality between these. The key idea is that for the true signal directions θ the steering vectors $\mathbf{a}(\theta)$ reside in the signal subspace $\mathbf{a} \in \mathcal{U}_s$ and are orthogonal to the noise subspace. This results is

$$\mathbf{a}^{\mathrm{H}}(\theta)\mathbf{U}_n\mathbf{U}_n^H\mathbf{a}(\theta) = 0, \tag{3.17}$$

leading to the formulation of the the MUSIC pseudo spectrum defined as [4]

$$P_{\text{MUSIC}}(\theta) = \frac{1}{\mathbf{a}^H(\theta)\mathbf{U}_n\mathbf{U}_n^H\mathbf{a}(\theta)}. \tag{3.18}$$

The expression evaluates how orthogonal the steering vector $\mathbf{a}(\theta)$ is with the noise subspace. The primary advantages is that the method is very robust to noise and can resolve closely spaced targets. The disadvantages are that the method requires prior knowledge of the number of sources present in the data. Additionally, there is an increase in computation complexity due to the eigendecomposition.

4 | Radar target detection

This chapter focuses on radar target detection, detailing the specific radar system used in the project and describing its implementation. Furthermore, the drone data collection process is presented along with an an evaluation of the performance of the radar target detection.

4.1 Radar system overview

The radar utilised in the project is the `TI AWR1843` radar with its compatible development board `AWR1843BOOST` [7]. This is coupled with a data handling board, specifically the `DCA1000EVM` [8] as displayed in figure 4.1. This enables direct integration and data handling.

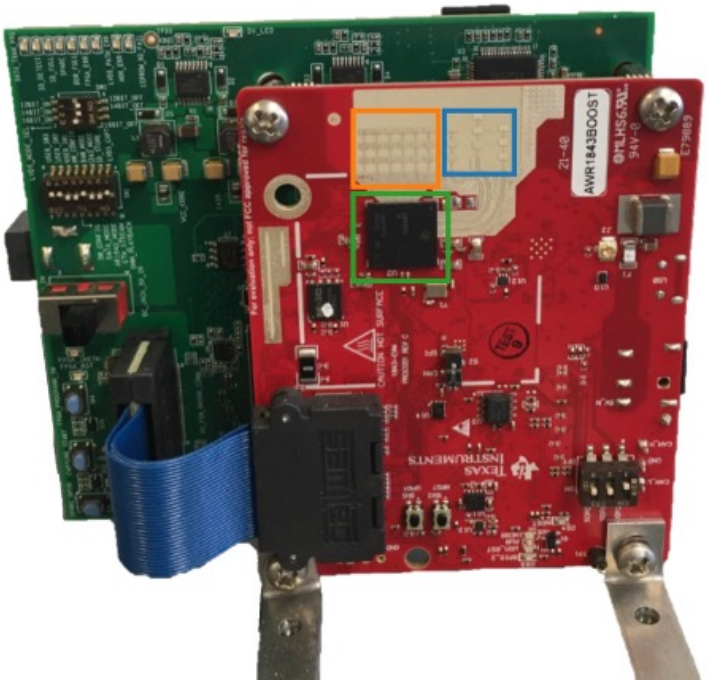


Figure 4.1: The used radar setup: the `TI AWR1843` is highlighted by the green box, the three transmitters by the blue box and the four receivers by the orange box. The handling board is the green board mounted on the back.

The radar chip is a FMCW radar with an operating frequency between 76GHz and 81GHz. The setup of the development and handling board will be referred to as the radar moving forward. The setup supports three $N_{\text{TX}} = 3$ transmitter elements and $N_{\text{RX}} = 4$ receiver elements arranged as a multidimensional array. Each TX/RX element has three physical patch antennas associated with it. Furthermore, one column of dummy patch antennas on each side of the receiver array are placed to reduce coupling between the elements. Additional radar specifications are given at [9].

From the physical antenna array a virtual antenna array is constructed as introduced in chapter 2. The physical and virtual antenna array is illustrated in the top and bottom of figure 4.2, respectively. Each VX array element corresponds to a TX-RX pair and the location is determined from the coordinates of the TX and RX elements [3]. The spacing between the RX and TX elements are $\lambda/2$ and λ, respectively. The TX elements are represented by blue circles, the RX elements by orange circles, and the VX elements by green circles.

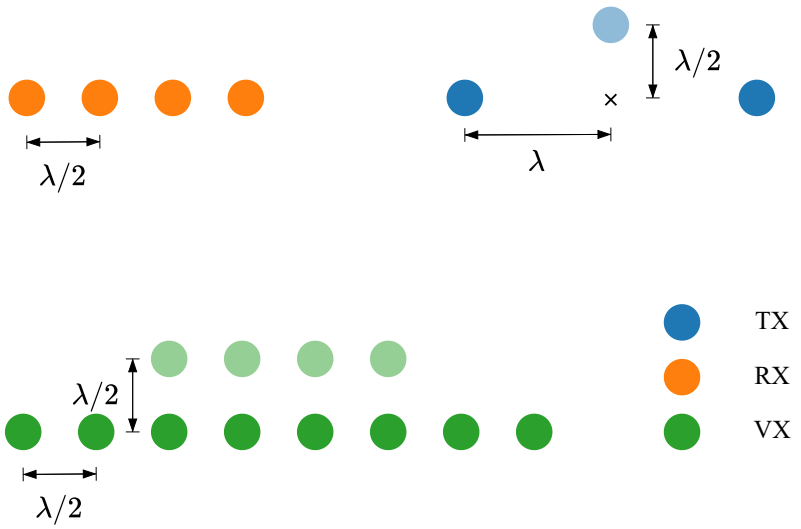


Figure 4.2: The MIMO array of the radar system illustrating the physical antenna array and the corresponding virtual antenna array.

Due to the few number of elements in the elevation plane, the project will disregard obtaining angle estimates in the elevation plane from the radar, but focus exclusively on the horizontal plane, i.e. the azimuth angle θ. The second TX element is only relevant

for angle estimation in the elevation plane and it will therefore be excluded, along with the corresponding VX elements, indicated by the transparent circles. Since the virtual elements only provide a double representation of the phase difference, yielding unnecessary additional computations. Thereby the structure of the virtual array is that of an $M = 8$ ULA.

The settings related to the range, velocity, and angle parameters, such as the data handling configurations, can be adjusted using the `MMWAVE-STUDIO` software [10]. The radar settings applied during data collection in this project are shown in figure B.3. These settings enable range, velocity, and angle parameters suitable for detecting the drone. The theoretical radar parameters are calculated and displayed in table 4.1 [5].

Parameter	**Value**
$r_{\max}$	49.995 m
Δr	0.1953 m
$v_{\max}$	±6.49 m/s
Δv	0.101 m/s
$\theta_{\max}$	±90°
$\Delta\theta$	14.3°

Table 4.1: Theoretical parameters of the radar.

4.2 Implementation

The implementation is a process with multiple steps covered here. To utilise the data received from the radar, it is first processed. The data from the `AWR1843BOOST` development board is structured as $\mathbb{R}^{128\times256}$ for each VX element for each frame, the values determined from the number of chirps and ADC samples applied in the FMCW radar. The data is initially subjected to a range FFT with a Chebyshev window function applied dividing the data into 256 range bins. The range FFT data is then further processed using a Doppler FFT with a Chebyshev window resulting in 128 Doppler bins. The FFTs are performed using [11]. This process produces a 2D FFT for each VX element for each frame, which can then be utilised for target detection. An example of such 2D FFT is illustrated in figure 4.3. Each cell in the resulting 2D map corresponds to a range and velocity interval of 0.195 m and 0.101 m/s, respectively, as specified by the resolution in table 4.1.

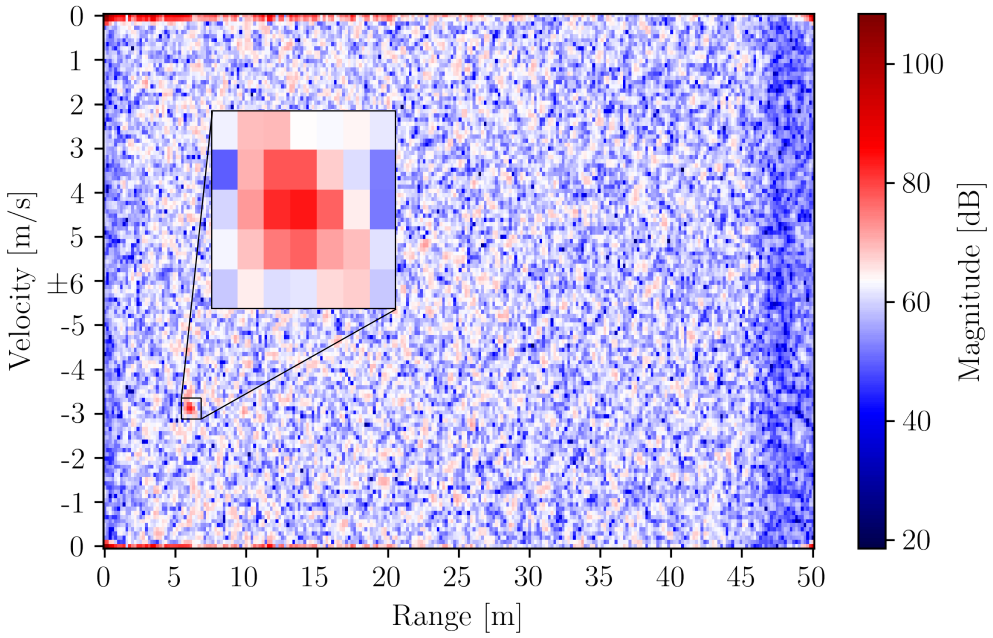


Figure 4.3: Illustration of 2D FFT constituted of range and velocity cells. The target is made up of approximately 5×3 cells.

Applying the Doppler FFT produces a moving target indicator [12] thereby filtering all the power from the stationary environment around the zero velocity axis. This provides a better estimate of the target approaching.

CA-CFAR

The CA-CFAR algorithm [13] is in this project applied for detecting the target from the radar data. The algorithm is implemented with (2, 1) guard cells on each side of the cell under test, organised as (range, Doppler), thereby covering the approximate size of the of drone, ensuring that minimal power from the target enters the reference cells. From tuning (2, 2) reference cells where chosen. The tuning was based on obtaining fewest false detections, while enabling the longest detection range. This configuration provides 48 reference cells enabling accurate noise level estimation and reducing the CA-CFAR loss. A constant false alarm rate of 1×10^{-3} is applied.

The CA-CFAR computes a threshold from the applied window. The threshold output from the CA-CFAR algorithm applied to the 2D FFT displayed in figure 4.3 is presented in figure 4.4.

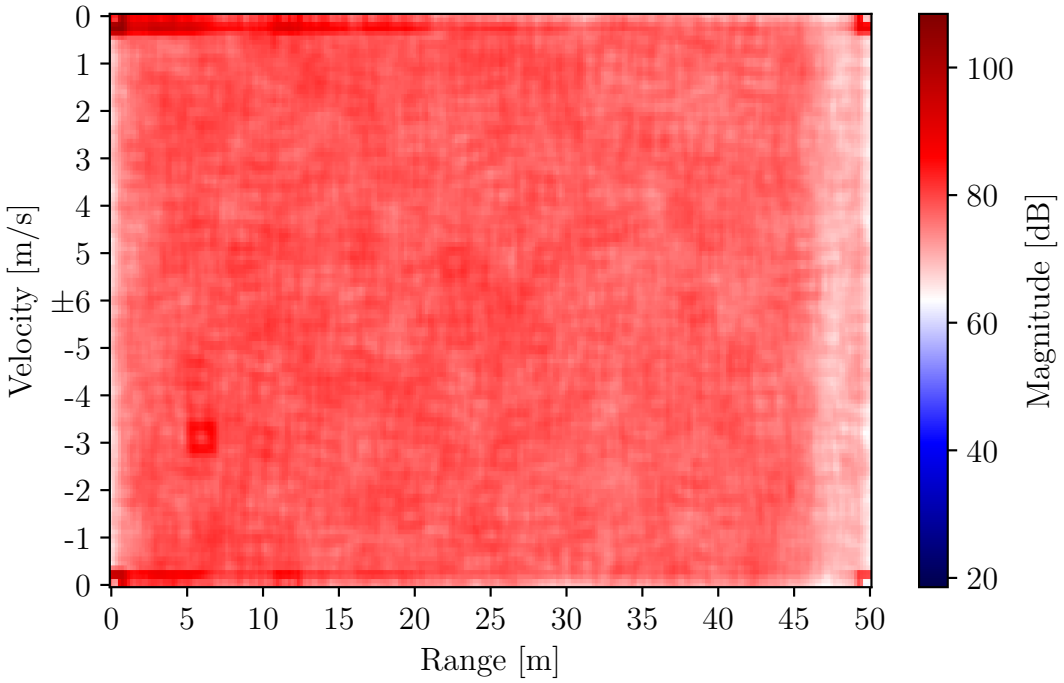


Figure 4.4: Threshold of the applied CA-CFAR algorithm.

Subsequently, the amplitude value of each bin is held up against the threshold. The maximum bin amplitude higher than the threshold is considered as the target. Thereby, a range and velocity bin index of the target is passed from the algorithm. Next, the DOA is estimated.

DOA methods

The processing scheme for the DOA estimation is: the 2D FFT for each VX element is non-coherently summed to create a predetection matrix [3]. The non-coherent summation is performed by adding the magnitude of the individual 2D FFTs together, thereby avoiding constructive and destructive interference due to the relative phase between the elements. Subsequently, the CA-CFAR algorithm is applied on the predetection matrix obtaining one common index for the peak corresponding to the target.

The angle FFT method utilises the phasor values of the peak index as touched upon in section 3.2. The common peak determined from the predetection matrix is applied thereby obtaining $M = 8$ phasors from the 2D FFTs of the VX elements. The sequence of phasors are padded with zeros, before computing the FFT, to enhance and smooth the spectrum without changing the underlying frequency content. The output is an angle spectrum displaying the received power across the interval $[-90°, 90°]$.

For the Bartlett and Capon beamformers and the MUSIC algorithm the covariance matrix is estimated. The number og snapshots is chosen to $K = 5$, i.e. the peak index ± 2 range bins, thereby representing the drone in the estimated covariance matrix. The measurement vector is therefore structured as $\mathbf{Y} \in \mathbb{R}^{8\times5}$ and the sample covariance is computed from (3.10). This structure also ensures enough noise in the channels of the estimated covariance such that the matrix is invertible as required in the Capon beamformer.

4.3 Data collection

A several number of data collections have been carried out to verify and evaluate the performance of the implemented methods for DOA estimation. The objective is to collect data for verification of the range, velocity, and angle estimates produced by the radar detection processing scheme. The target is a drone representative of a that which the Archangel system should detect and localise. The test equipment is summarised in table 4.2.

Equipment	**Manufacturer**	**Model**	**Specifications**
Drone	DJI	Phantom 4	[14]
Radar	TI	AWR1843Boost, DCA1000EVM	[7, 8]
PC	Lenovo	Lenovo Legion 17"	||
Power station	Goal Zero	Yeti 500X	[15]
Camera	GoPro	Hero 11 Black	[16]

Table 4.2: Test equipment used for the data collection.

The radar is operated using the `MMWAVE-STUDIO` software version: 02.01.01.00 [10]. The radar settings applied during data collection in this project are displayed in figure B.3. The vision aspect of the target detection is not the focus in this project, however a `GoPro` camera was applied to obtain additional position estimates of the drone to utilise in the data fusion scheme.

Setup

The setup of the data collections is seen in figure 4.5. The drone was positioned at a location measured with kevlar string with pre-measured labels indicating the desired distances. The distances were also calculated to obtain specific flight paths of the drone. Five different data collections were performed to properly evaluate the performance. The format of the test designation is primarily: `drone_`θ`_`ϕ`_`r`_`$additional\ information$, where r is the distance from the radar to the drone at start position. The five datasets and a description of their individual setups and procedures are:

- `Drone_0_0_30` ●
  Flying the drone from $r = 30\,\text{m}$ towards the radar at boresight $\theta = 0°$ and $\phi = 0°$.
- `Drone_0_15_41` ●
  Flying the drone from $r = 41\,\text{m}$ towards the radar at $\theta = 0°$ and $\phi = 15°$. This approach simulates the behaviour of an attacking drone.
- `Drone_30_0_30` ●
  Flying the drone from $r = 30\,\text{m}$ towards the radar at $\theta = 30°$ and $\phi = 0°$.
- `Drone_perpendicular_towards` ●
  Flying the drone from $r = 14\text{m}$ at an angle towards the drone and past it to the right. Thereby obtaining varying angle estimates with the target velocity being different from zero at all frames.
- `Drone_0_0_30_stopping` ●
  Flying the drone from $r = 30\,\text{m}$ towards the radar at boresight $\theta = 0°$ and $\phi = 0°$. The drone is stopped at $r = 15\,\text{m}$ and subsequently accelerated again towards the radar.

An illustration of the collective flight paths of the drone for the five tests is presented in figure 4.5. The respective flight paths are highlighted by the colours corresponding to the descriptions above.

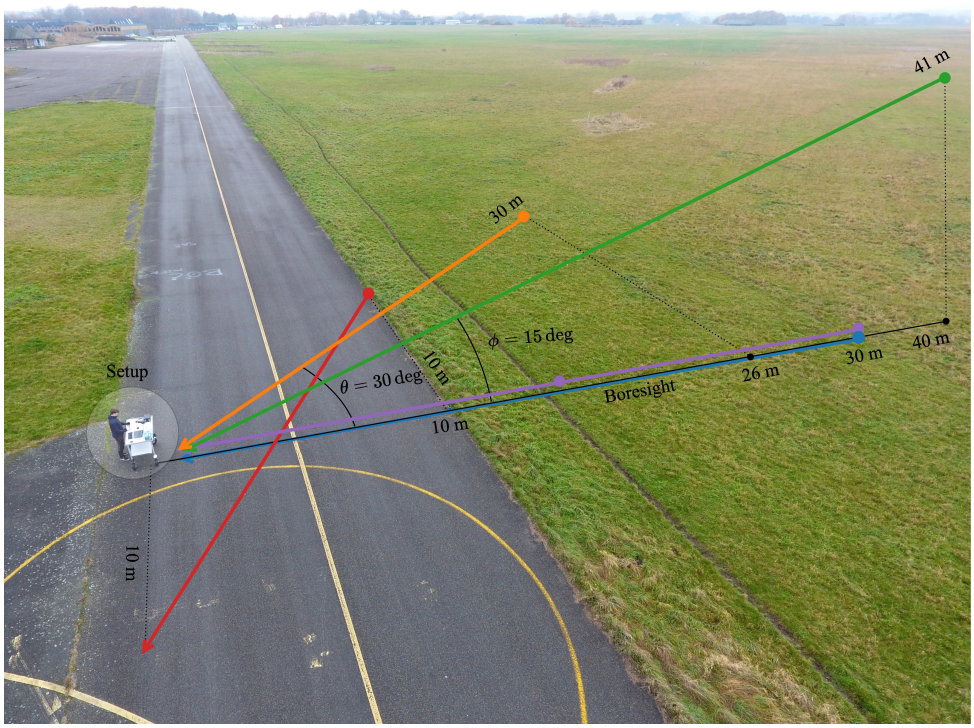


Figure 4.5: Illustration of collective flight paths of the five tests. The colours representing the respective tests.

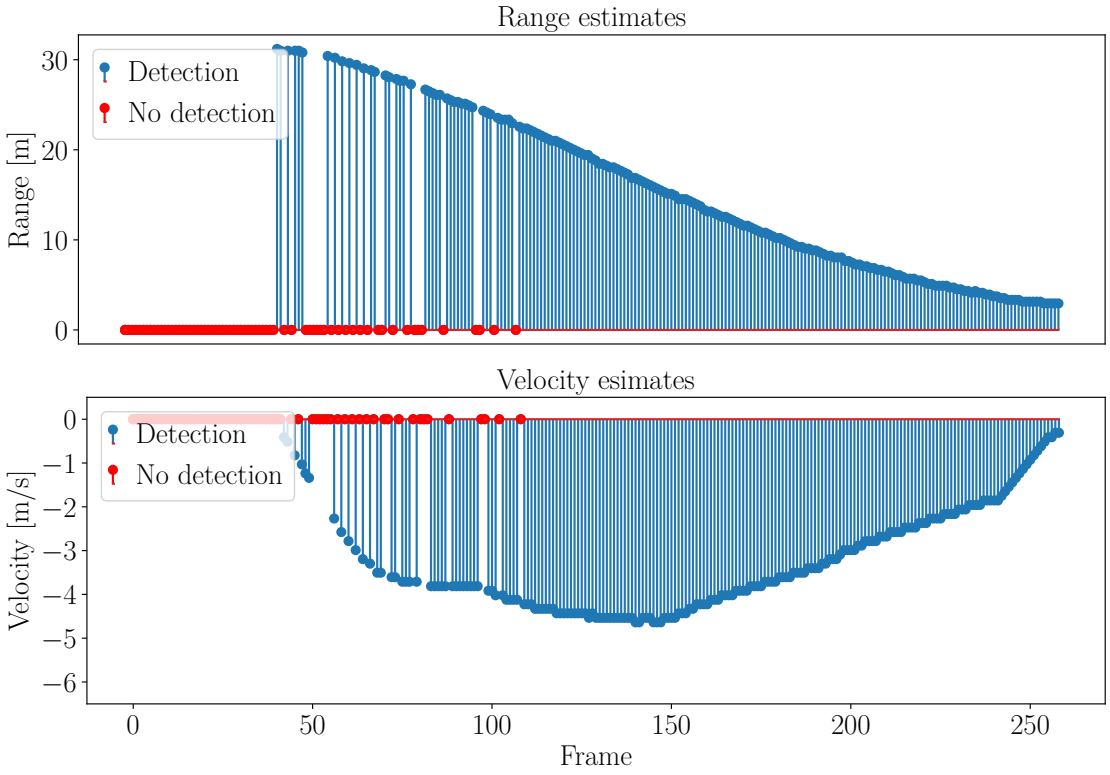
The flight paths displayed are the theoretical/ideal flight paths. The true flight path of the drone differs from this, as the real world case introduced variations in the flight path, e.g. due to wind conditions and since the drone was manoeuvred by hand introducing human error.

4.4 Results

The outcomes of the radar target detection scheme introduced in this chapter 4, are presented and evaluated based on the introduced data collected in section 4.3. The performance of the radar is assessed based on the estimates of the range, velocity, and the DOA. The resolution denoted by the Δ in the radar parameters displayed in table 4.1 is the minimum distance to be able to distinguish between targets. However, for the range and velocity, this also denotes the spacing between data points. For the angle the interval between points is one degree, although not for the angle FFT method due to the nonlinearity of the angle estimates described in section 3.2. The sample time is $\Delta t = 40\,\text{ms}$, i.e. 25 samples are collected each second. The results for the data collection are displayed in the presented order.

`Drone_0_0_30`

The results for the `drone_0_0_30` test are displayed in figure 4.7 The estimated states: the range, velocity, and angles are be presented in this sequence with frame number (time index) along the first axis and the state value along the second axis.



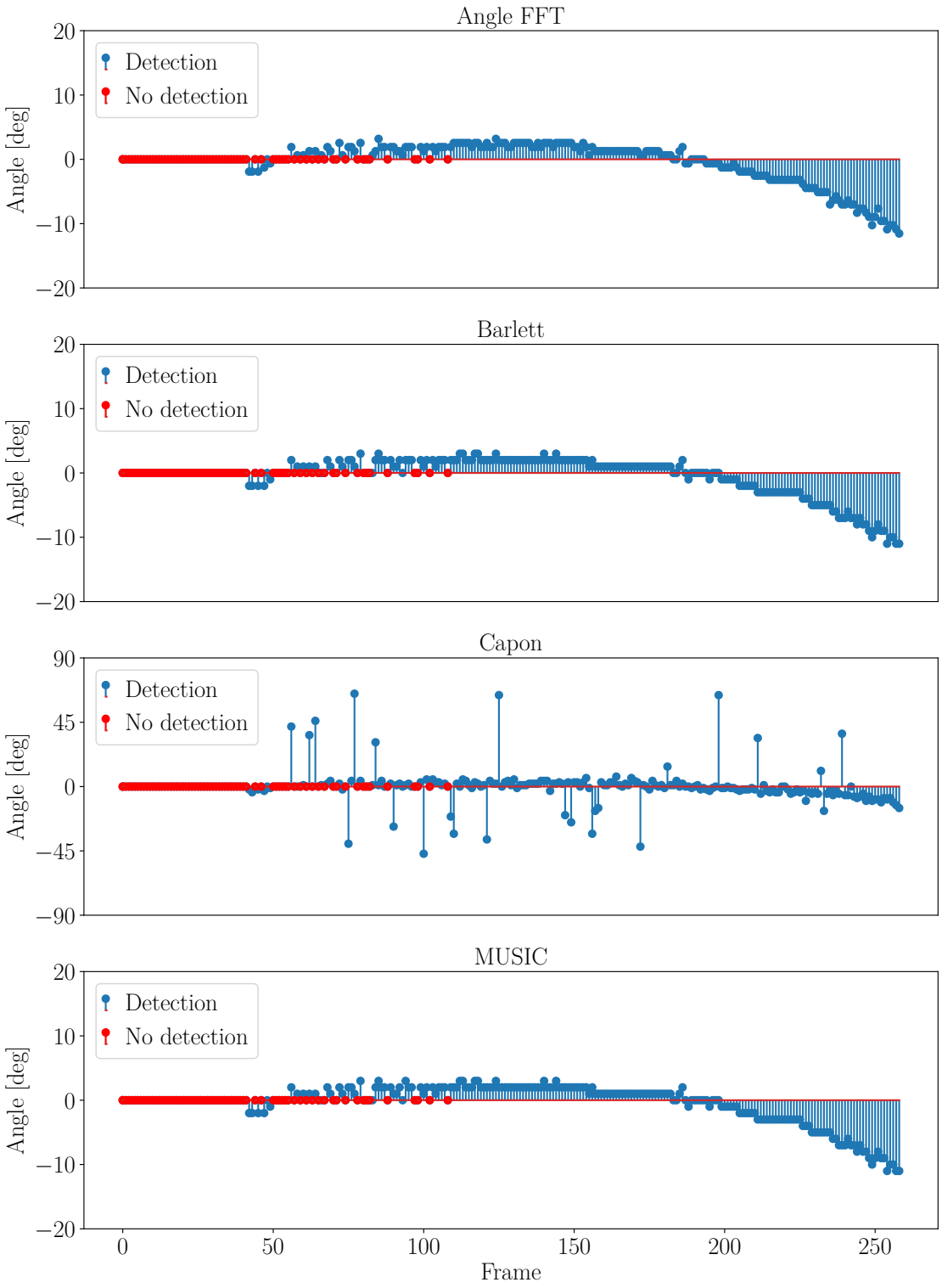


Figure 4.7: Results for the state estimates of the range, velocity, and angles for the `drone_0_0_30` test.

The range and velocity estimates highlight the capabilities of the radar. The maximum range measurement aligns well with the observed data, supported by velocity estimates that start near zero and display an acceleration from that point. The detection range is good, with no apparent false detections, i.e. no incorrectly identified targets in the 2D FFT processed by the CA-CFAR algorithm. There appear gaps between the measurements from a longer range, but from around 20 m the measurements are consistent. The angle estimates also appear reliable. The angle FFT, Bartlett beamformer, and MUSIC algorithm yield similar DOA estimates. However, noticeable variations arise with the Capon estimates, which underperform compared to the other three methods. This may be attributed to inaccuracies in the covariance matrix estimation and an insufficient number of snapshots.

To compare the results of the four DOA estimation methods the RMSE is computed and displayed in table 4.3. The RMSE is calculated with respect to the theoretical flight path value of 0°. Considering, that the value of 0° does not necessarily represent the physical true value of the drone (this is rather unlikely) the RMSE value should not be interpreted as an evaluation of the DOA estimation method in its ability to accurately measure the true DOA. However, it showcases the similar performance between angle FFT, the Bartlett beamformer, and the MUSIC algorithm in DOA estimation, and the larger variation of the Capon beamformer.

DOA method	**RMSE**
Angle FFT	3.13°
Bartlett	3.05°
Capon	11.84°
MUSIC	3.05°

Table 4.3: RMSE value of the four DOA estimation methods.

Additionally, the average 3 dB drop-off angle interval is compared between the four methods and displayed in table 4.4. The table clearly highlights the performance differences of the methods in terms of the sharpness of peaks in estimating the DOA.

DOA method	−3 dB interval
Angle FFT	20.10°
Bartlett	12.82°
Capon	3.84°
MUSIC	3.77°

Table 4.4: The average 3 dB drop-off interval of the four DOA estimation methods.

Following this, the results for the remaining four data collections are presented. These results will be presented in the transformed coordinate system, i.e. polar to Cartesian, to compare against the theoretical flight path of the drone. To obtain this transformation the angle estimate from the MUSIC algorithm is used.

`Drone_0_15_41`

The results for the `drone_0_15_41` test are displayed in figure 4.8.

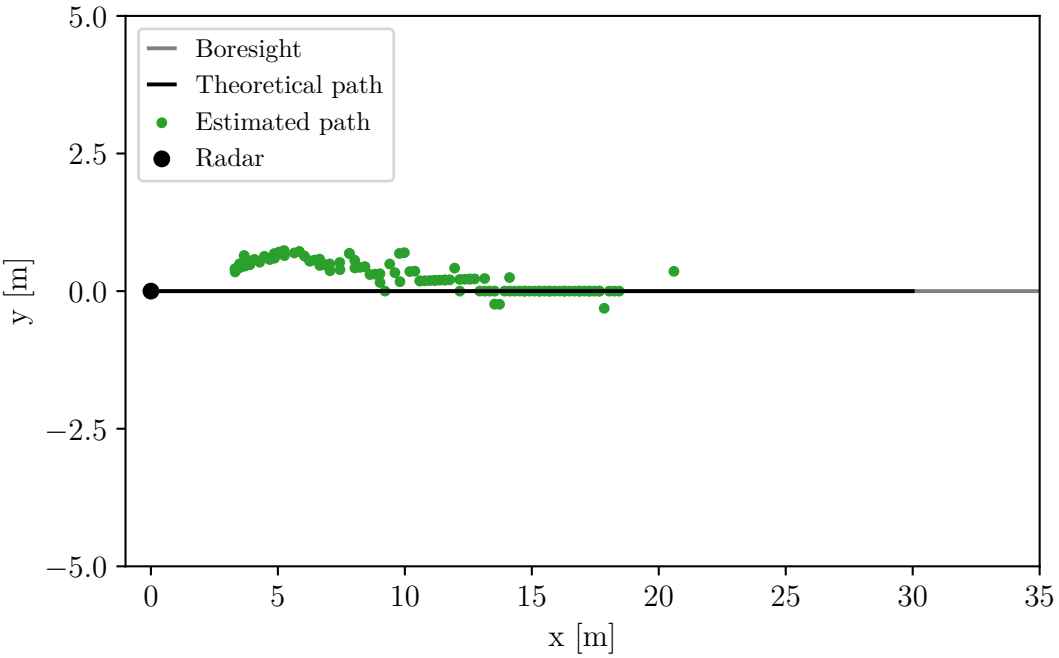


Figure 4.8: Estimated flight path of the drone for the `drone_0_15_41` test.

The results highlights the reduced detection range of the drone approaching at an elevation angle of $\phi = 15°$. This substantially reduce the performance of the radar target detection.

This is supported by radiation pattern of the radar in the elevation plane (E-plane) presented in [7]. The data point resolution, i.e. the interval between data points, on one degree is quite, yielding noisy position estimates especially at higher range. There are no false detections.

`Drone_30_0_30`

The results for the `drone_30_0_30` test are displayed in figure 4.9. This test introduces a constant approach azimuth angle of $\theta = 30°$. The measured path approximates the theoretical path and appear to correspond well to the true location. Especially considering that the true flight path likely differs from the theoretical.

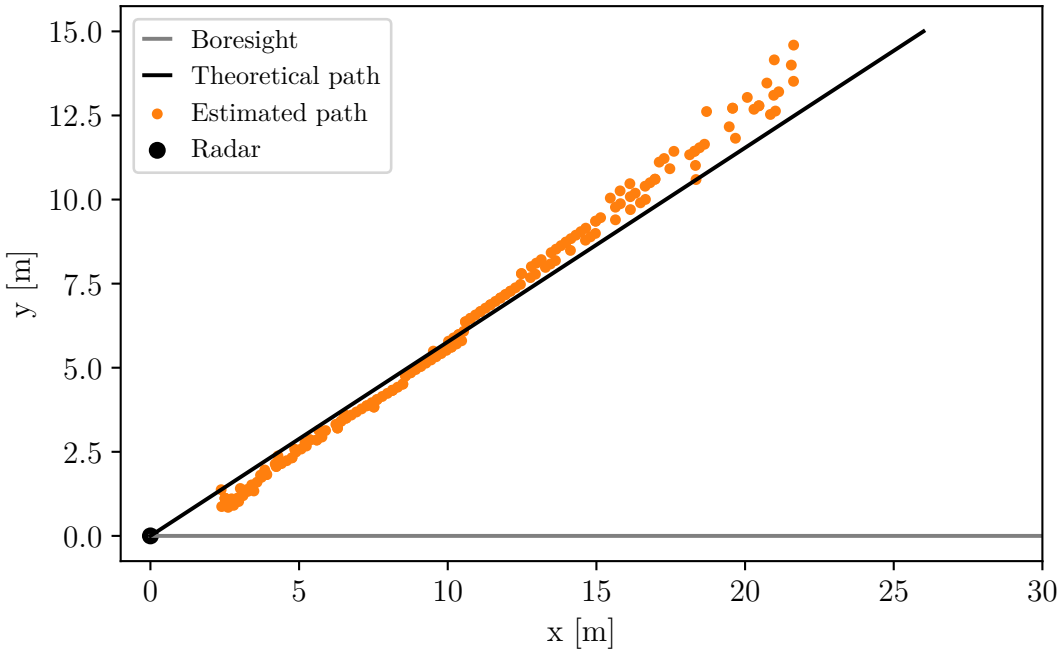


Figure 4.9: Estimated flight path of the drone for the `drone_30_0_30` test.

Noticeably, the detection range is still high in spite of the DOA being off boresight. This is in accordance with the radiation pattern of the radar in the horizontal plane, having a wider 3 dB and 6 dB beamwidth (approximately twice) as that of the E-plane.

`Drone_perpendicular_towards`

The results for the Radar `drone_perpendicular_towards` test are displayed in figure 4.10. This test introduces a varying azimuth angle in a wide interval. The obtained estimates seem to correspond well to the theoretical flight path of the drone.

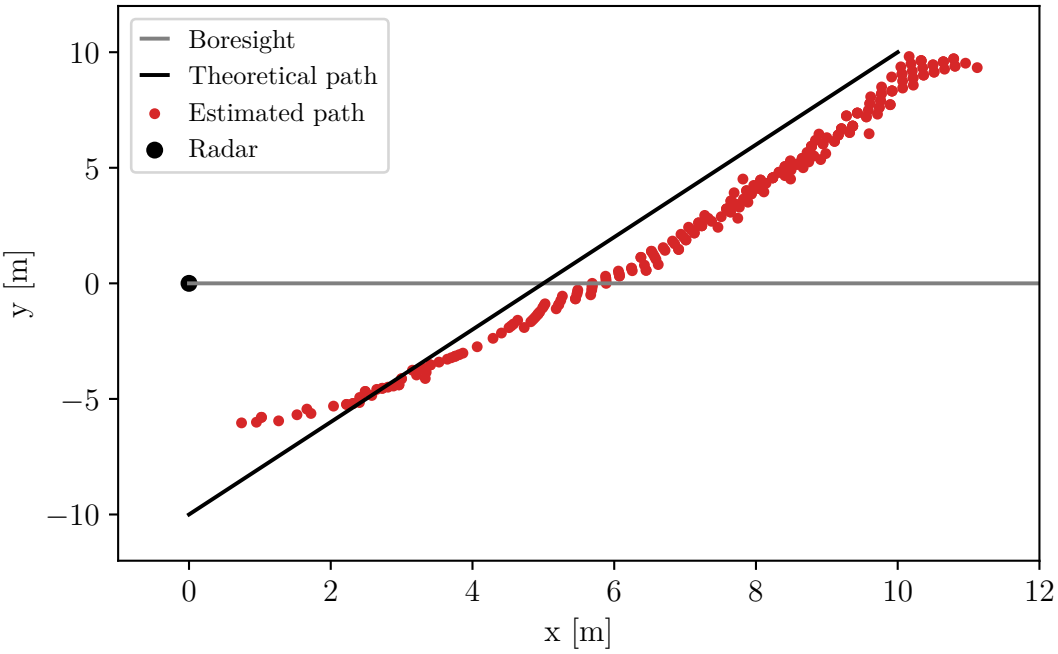


Figure 4.10: Estimated flight path of the drone for the `drone_perpendicular_towards` test.

Noticeably, the effectiveness of the target detection decreases as the angle to the radar increases, evident from the fewer data points present. The radiation patterns of the radar contributes to the effectiveness of the target detection, and needs to be carefully considered when designing the radar system.

`Drone_0_0_30_stopping`

The results for the Radar `drone_0_0_30_stopping` test are displayed in figure 4.11. This test investigates how the radar system handles the scenario of an attack drone stopping on approach and subsequently reinitialising the attack.

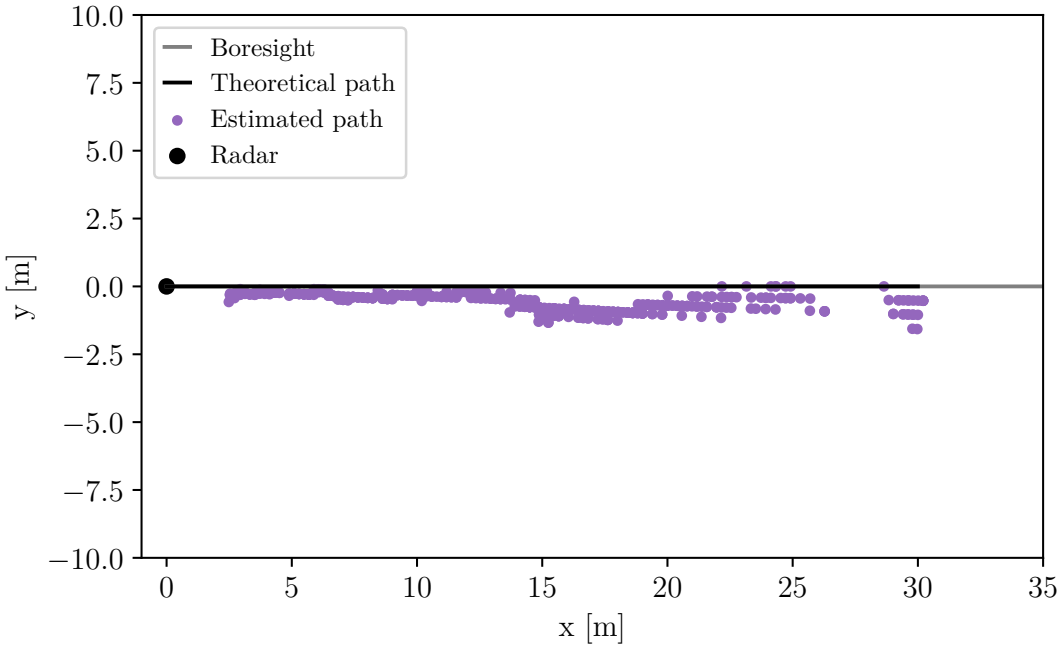


Figure 4.11: Estimated flight path of the drone for the `drone_0_0_30_stopping` test.

There are no introduced false detections or gaps in the position estimates of the drone. However, the drone disappears in the radar target detection and a strategy should be instigated to handle a drone stopping and hovering in place. The influence of the one degree data point resolution in the angle estimates on the position estimates at a high range is noticeable.

5 | Data fusion

To increase the accuracy and reliability of the position estimates of the drone a data fusion scheme is introduced. In this project, a real-time application requires handling measurement data sequentially rather than as a single batch. Implementing a recursive estimation method to process multiple sequential measurements is essential. This is accomplished using a Kalman filter. The theoretical operational principle of the Kalman filter will not be presented in this report, but is found at [17, 18]. The Kalman filter implementation within the Archangel project is introduced followed by the results from the data fusion scheme.

The GoPro camera was utilised to obtain azimuth and elevation angle estimates in addition to the measurements from the radar. The `YOLO11` [19] object detection algorithm was applied using the weights learned from training the model in [2]. The angles were obtained from a direct linear relation between the camera FOV and pixel coordinate and no advanced lens optics was considered in the conversion.

5.1 Kalman filter dynamics

A Kalman filter is a recursive method for linear minimum mean square error estimation. It operates iteratively, using a dynamic system model, a set of equations, and sequential data inputs to estimate the true values of system states, such as the position of a drone. The Kalman filter is implemented as a data fusion algorithm between the position estimates from the radar and a camera.

Measurements are obtained from the radar: the range, velocity and azimuth angle denoted as r, v, and θ_r, respectively, and from the camera: the azimuth and elevation angle denoted as θ_c and ϕ_c, respectively. The sensor measurement vector μ is defined as

$$\mu = \begin{bmatrix} r & \theta_r & v & \theta_c & \phi_c \end{bmatrix}^T. \tag{5.1}$$

The Kalman filter is an ideal framework for sequential updating with linear model operators and Gaussian error distributions [20]. Therefore, applying a linear Kalman filter requires consideration regarding the state definitions. In the physical environment the

movement of the drone is linear in the Cartesian coordinate system and is non-linear in spherical coordinates. To apply the linear Kalman filter for drone localisation the input measurements need to be in Cartesian coordinates therefore requiring a transformation of the measurement vector. The appropriate state vector $\mathbf{x}$ of the Kalman filter is defined as

$$\mathbf{x} = \begin{bmatrix} x & y & z & v_x & v_y & v_z \end{bmatrix}^{\mathrm{T}}, \tag{5.2}$$

where x, y, and z are the positional coordinates and v_x, v_y, and v_z are the respective velocity components.

Model matrices

To apply the Kalman filter, a state model that describes the system's dynamics must be established. In this project, the state model and process noise covariance are constant and will be represented as $\mathbf{\Phi}$ and $\mathbf{Q}$, respectively. Additionally, no control input is included, so it will be omitted from the model. The state model is defined as

$$\mathbf{\Phi} = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{5.3}$$

where Δt denotes the time between samples. A measurement model is set up to relate the measurements to the state model. The process noise covariance is determined applying Newton's laws of motion, the property of normal distributions, and the physical acceleration characteristics of the `Phantom 4` drone [14, 21]. The process noise covariance is defined as [2]

$$\mathbf{Q} = \mathbf{G}\mathbf{\Sigma}_a\mathbf{G}^{\mathrm{T}}, \tag{5.4}$$

where $\mathbf{\Sigma}_a$ is the effect of an unknown input and $\mathbf{G}$ applies this effect to the state vector. The unknown input is modelled as $\mathbf{\Sigma}_a = \sigma_a^2\mathbf{I}$, where σ_a^2 is acceleration variance determined from the three standard deviation value of the acceleration $3\sigma_a = 6\,\mathrm{m/s^2}$. And $\mathbf{G}$ is defined as

$$\mathbf{G} = \begin{bmatrix} \frac{\Delta t^2}{2} & 0 & 0 \\ 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & 0 & \frac{\Delta t^2}{2} \\ \Delta t & 0 & 0 \\ 0 & \Delta t & 0 \\ 0 & 0 & \Delta t \end{bmatrix}. \tag{5.5}$$

The input measurement vector applied to the Kalman filter is defined as

$$\mathbf{y} = [\underbrace{x \quad y \quad v_x \quad v_y}_{\text{radar}} \quad \underbrace{x \quad y \quad z}_{\text{camera}}]^{\mathrm{T}}, \tag{5.6}$$

obtaining positional coordinates from both the radar and camera, and measurements of the velocity components only from the radar. No measurements are obtained for v_z which is only inferred by the Kalman filter. In accordance with the employed coordinate systems presented in figure 1.2 the mapping function $f(\cdot)$ from the spherical sensor measurements to Cartesian input measurements is performed as

$$\mathbf{y} = f(\mu) = \begin{bmatrix} r\cos(\theta_r) \\ r\sin(\theta_r) \\ v\cos(\theta_r) \\ v\sin(\theta_r) \\ r\cos(\theta_c)\cos(\phi_c) \\ r\sin(\theta_c)\cos(\phi_c) \\ r\cos(\phi_c) \end{bmatrix}. \tag{5.7}$$

Here the transformation to the velocity components is only valid for a drone flight path

straight towards the sensor setup, which is assumed in the scope of this project. The range measurement from the radar is also used in the camera conversion. The observation model is defined as

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{5.8}$$

For the Kalman filter to operate optimally the inputs should be Gaussian. The measurement noise is a random variable which is Gaussian in nature. To define the transformed measurement noise covariance and ensuring that the distribution of the transformed measurement noise is Gaussian the delta method is applied. Utilising the delta method to approximate the distribution of the mapping function of the measurement noise using a first order Taylor expansion [22]. The covariance of the transformed measurement noise $\mathbf{R}_n$ is given as

$$\mathbf{R}_n = J_f(\mu)\mathbf{\Sigma}_n J_f(\mu)^T, \tag{5.9}$$

where $J_f(\mu)$ denotes the Jacobian of the mapping function (5.7) and $\mathbf{\Sigma}_n$ the covariance matrix in the spherical coordinate system. This is made up of variance σ^2 values of the measurements from the radar and camera. To achieve an invertible (non-singular) measurement noise covariance matrix (5.9) was implemented using block matrices to force independent columns [23]. Depending on whether the radar or the camera provides measurements $\mathbf{\Sigma}$ is changed at each iteration n of the Kalman filter. The covariance matrix indices corresponding to sensors that provide no measurements are assigned a large variance value of 10^6, effectively ignoring the 'no measurement' value of zero and relying on the model instead. If measurements are received from both sensors the it is defined as

$$\mathbf{\Sigma}_s = \begin{bmatrix} \sigma_r^2 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{\theta_r}^2 & 0 & 0 & 0 \\ 0 & 0 & \sigma_v^2 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\theta_c}^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\phi_c}^2 \end{bmatrix}. \tag{5.10}$$

The variance values of the radar is selected based on the number of bins of the drone introduced in 4.2. The 3σ values for the range and velocity are five and three bins, respectively. The variance values are calculated and equal

$$\sigma_r^2 = 0.106\,\text{m}$$
$$\sigma_v^2 = 0.010\,\text{m/s}$$

The variance $\sigma_{\theta_r}^2$ is determined from the 3 dB drop-off value from the peak in the spectrum and therefore varies significantly depending on the applied method for DOA estimation as as presented in section 4.4 and indicated in figure 3.2. The variance values for the camera are established from the size of the boundary box obtained from the object detection method as [2]

$$\sigma_{\theta_c}^2 = (1/3)^2 = 0.11°$$
$$\sigma_{\phi_c}^2 = (1/5)^2 = 0.04°$$

Initialisation

Prior to the first iteration of the prediction and update steps of the Kalman filter, it is initialised with an initial state vector $\mathbf{x}_0$ and an initial covariance matrix $\mathbf{P}_0$. Given the uncertainty in the radar and camera measurements, a straightforward initialisation scheme is employed. The Kalman filter is initialised when reliable measurements are obtained from both sensors. The time index for measurements received preceding the Kalman initialisation will be denoted as k. Measurements are deemed reliable if at time k there are five measurements within the previous $k - 15$ measurements. Additionally,

measurements from both sensors are required. The initial state vector is then defined as

$$\mathbf{x}_0 = [\underbrace{x_k \quad y_k \quad z_k}_{\text{camera}} \quad \underbrace{v_{x_k} \quad v_{y_k}}_{\text{radar}} \quad -1]^T, \tag{5.11}$$

using the measurements from each sensor and setting $v_z = -1$, since a small order negative velocity is expected. The initial estimate covariance matrix $\mathbf{P}_0$ is given as

$$\mathbf{P}_0 = \text{diag}\left(\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 10 \end{bmatrix}\right). \tag{5.12}$$

The initial states are based upon reliable measurements and the corresponding indices are given a small variance. However, the index related to v_z is higher as this is set somewhat arbitrarily. This concludes the fundamental setup of the Kalman filter in the Archangel project. Implementing the defined quantities the Kalman filter is executed.

5.2 Results

In order to asses the performance of the developed data fusion scheme it is applied on the `drone_0_15_41` test described in section 4.3 which imitates the approach of an attacking drone. The Kalman filter position and veolcity states are displayed in figures 5.1 and 5.2, respectively.

An initial analysis of the results indicate that the Kalman filter effectively reduces measurement noise as expected. This is apparent for the state y where especially the radar measurements fluctuates a lot, though within a small physical distance of one meter. There are only two 'no measurements' from the camera after the Kalman filter is initialised, which has a marginal effect and is handled well. This test does not demonstrate how effectively the Kalman filter manages a high number of missing measurements. However, it is expected to perform well in such cases since it relies on the model when measurements are unavailable. It is important to note that the y state estimate is attenuated. This occurs because the corresponding velocity component is negative, even though the movement of the drone is in the positive y direction. The reason lies in the mapping function (5.7) for the velocity components, which is not valid for the flight path of the drone.

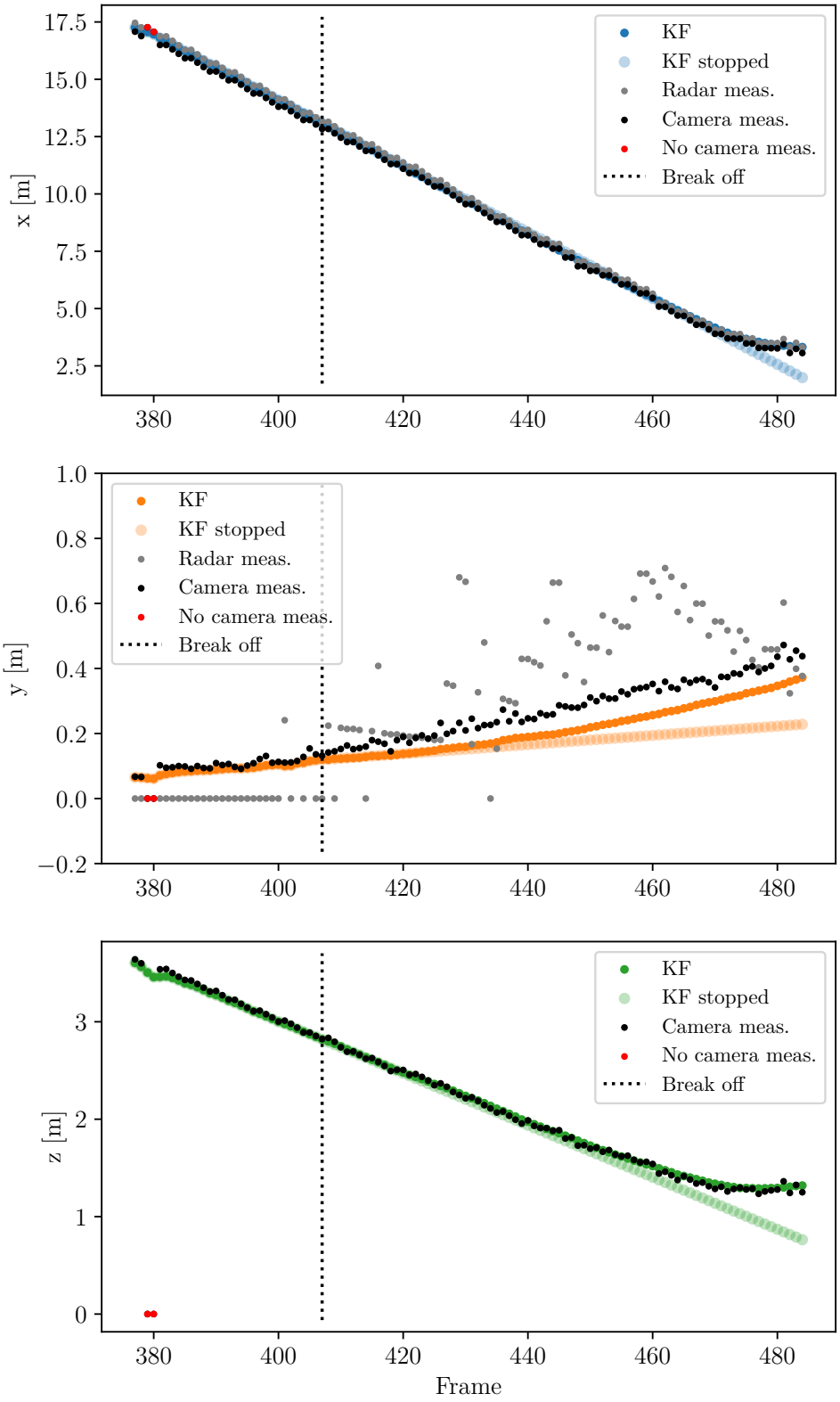


Figure 5.1: Position state estimates from Kalman filter from frame 377 to 484 for `drone_0_15_41` dataset.

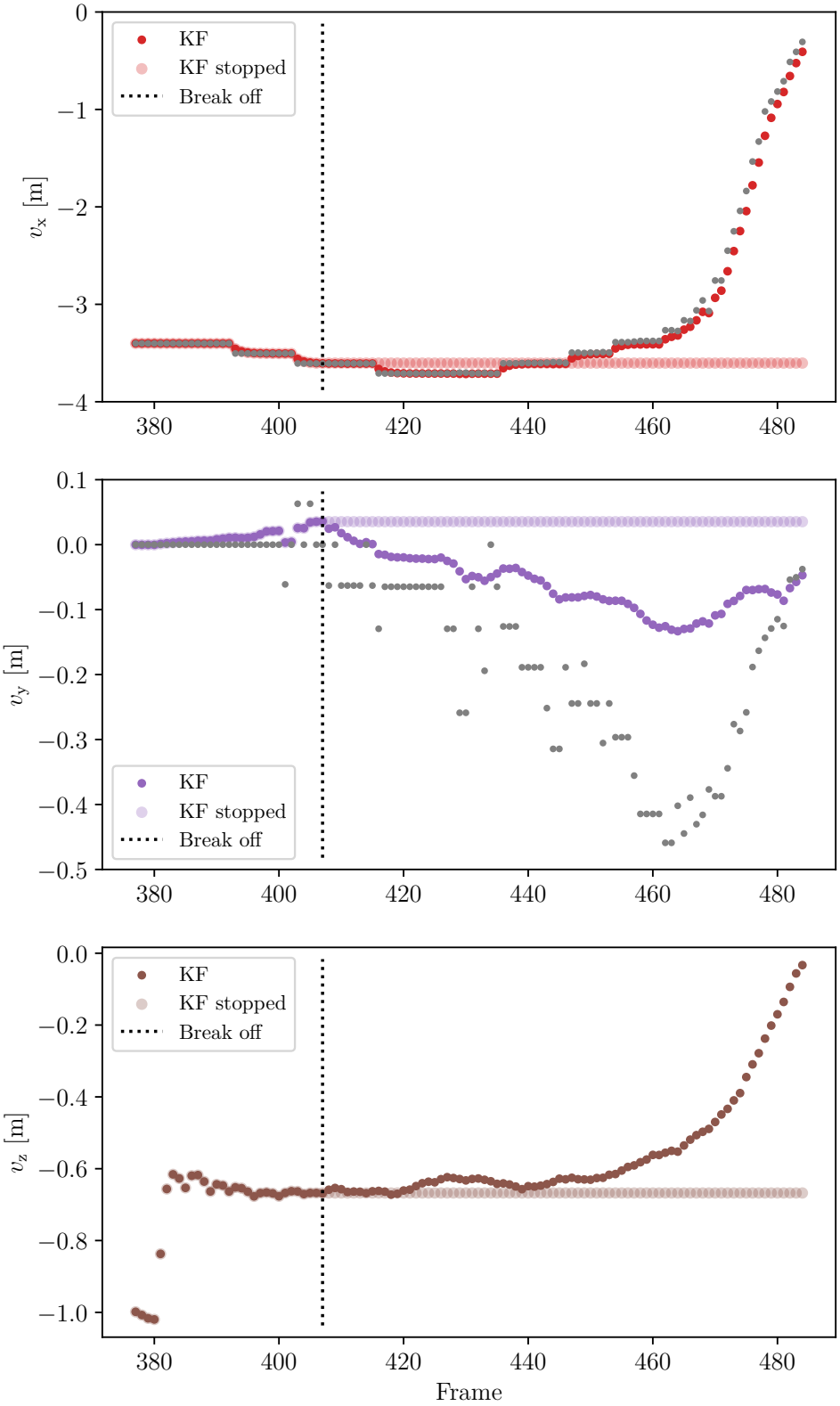


Figure 5.2: Velocity state estimates from Kalman filter from frame 377 to 484 for `drone_0_15_41` dataset.

The flight path does not align directly with the test setup, resulting in incorrect transformed velocity component measurements. Consequently, the dynamic system model attenuates the y state position as a result.

The Kalman filter is also evaluated on its performance without new input measurements in order assess the Kalman filter in its ability to model the movement of the drone. The data stream to the filter is ceased at frame 407, providing the filter with 30 measurements, denoted as the 'break off' point. Subsequently, the filter only performs the prediction step without any updates. The Kalman filter manages this situation well due to the linear movement of the drone. However, it struggles to accurately estimate the erratic variations in velocity.

To compare the data fusion results with those from the radar target detection the the estimated flight path in the horizontal plane is displayed in figure 5.3. This figure is directly related to figure 4.8 displaying the radar measurements indicated by the grey points. The noise is substantially reduced, but the filter suffers from the attenuated position along y.

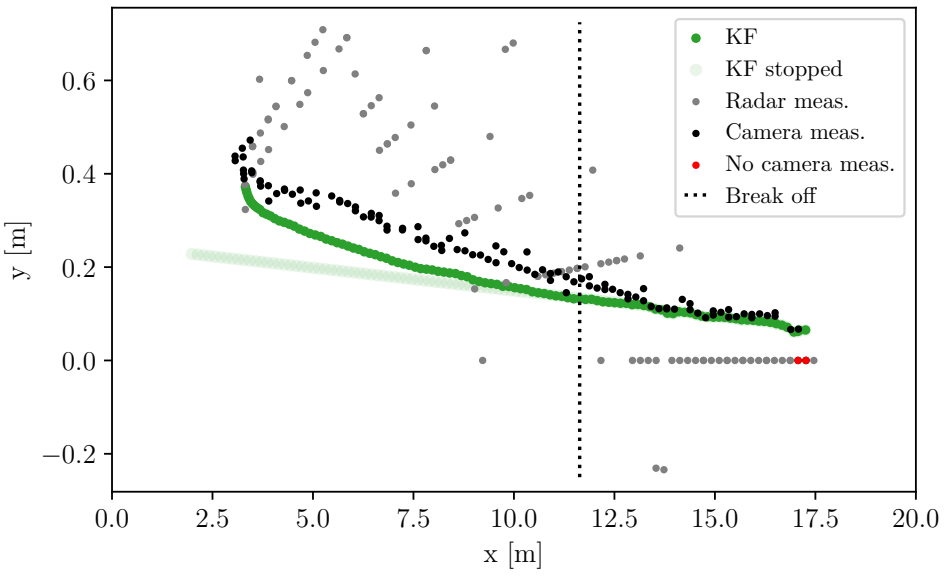


Figure 5.3: Combined Kalman filter states in horizontal plane shows the estimated flight path of the drone for the `drone_0_15_41` test.

The Kalman filter can be understood from a Bayesian perspective, as presented in [20]. The estimate covariance matrix $\mathbf{P}$ represents the posterior and can be interpreted as how certain the Kalman filter is the estimated states. The confidence region for the estimate

covariance matrix will have the shape of an ellipse (in two dimensions). The confidence region for the estimated covariance matrix takes the shape of an ellipse in two dimensions. This region is derived from the matrix as follows: the eigenvectors define the directions of the principal axes of the ellipse, while the eigenvalues represent the variance along these axes, scaled by a factor corresponding to the desired confidence interval 5.4 [24, 25]. The results of this approach are illustrated in the figure.

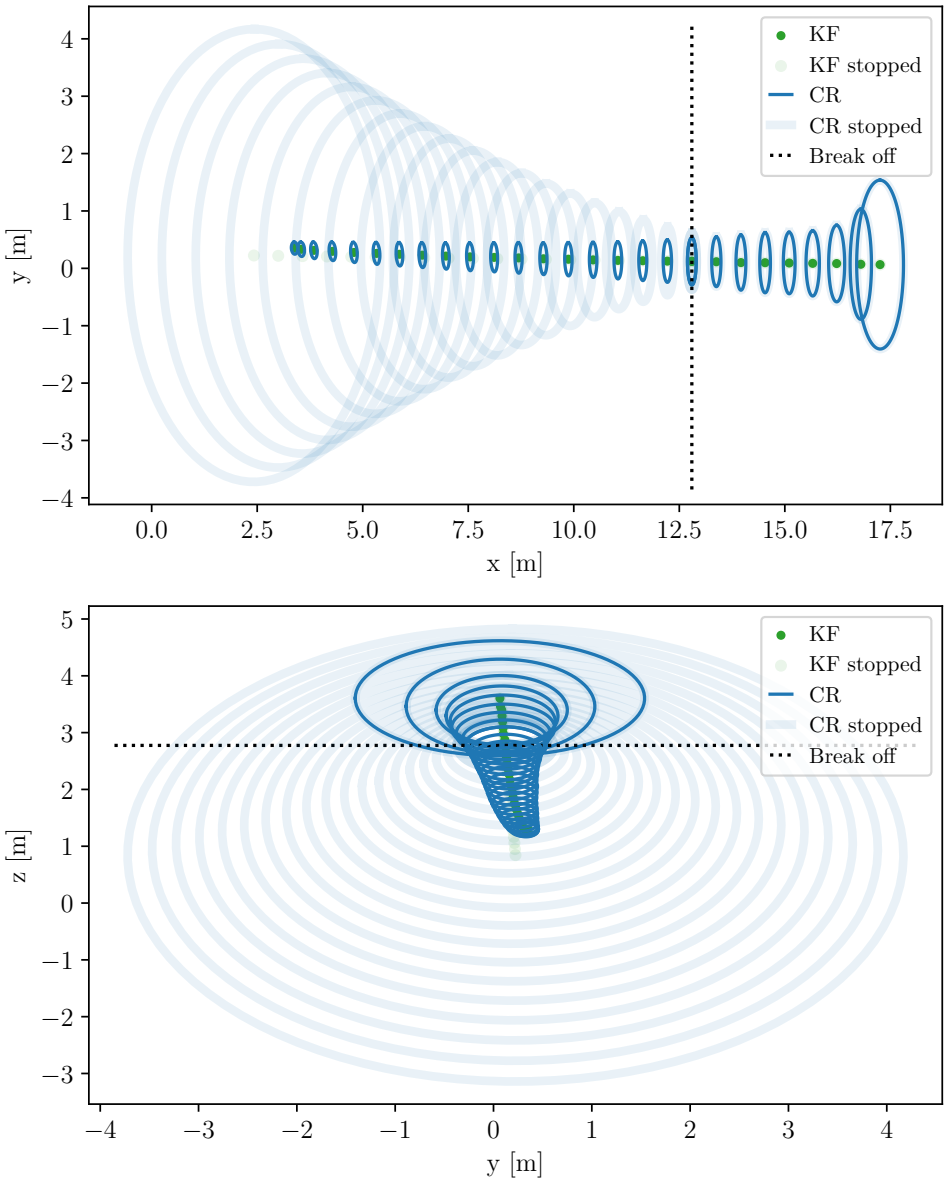


Figure 5.4: Confidence regions in the horizontal and elevation plane, respectively. The 95 % confidence regions are plotted every fourth frame for the Kalman filter with input measurements and without.

The figure displays the 95 % confidence regions in the horizontal and elevation plane plotted every fourth frame. This means that with the dynamic system model and the measurements the Kalman filter is 95 % certain that the drone is contained within the ellipsoids. This region presents a possible area that the projectiles should cover in the context of the Archangel system. The confidence regions gets smaller as more measurements are obtained substantiating the estimated states. It is worth noting that the confidence region can get smaller than the physical area of the drone since the Kalman filter interprets the drone as a single point in space. The confidence regions for the Kalman filter when no input measurements are received are also displayed. This highlights the increasing uncertainty in connection with not receiving measurements.

Video material

To easily display the results obtained from the Kalman filter the video `kalman.mp4` is produced and appended of the drone approaching the camera and radar setup in the `drone_0_15_15` test. At frame 150, displayed in the top left of the video, the drone is flown towards the sensors. The camera target detection identifies the drone far out at around 40 m which is visualised with the green box ■ surrounding the drone. At frame 377 the Kalman filter is initialised as the camera and especially the radar measurements have become reliable. The Kalman filter states are transformed back to spherical coordinates and are visualised with the blue cross ✖ and accompanying blue text displaying the state values r, v, θ, and ϕ. The confidence regions is indicated by the blue circle ● and displayed for each frame. The Filter tracks the drone well, however the effect of the attenuation of the y coordinate is noticeable and misplaces the drone a bit to the right. The results are also displayed for the Kalman filter when the input measurements are stopped and only prediction is applied. The position state estimate is visualised with the red cross ✖ and the circle ● indicating the confidence regions. The Kalman filter position estimates are affected, and it becomes increasingly uncertain in the state estimates. However, the filter still captures the linear movement of the approaching drone.

6 | Discussion

This chapter serves the purpose of discussing the developed scheme for the proposed Archangel system. The aim is to acknowledge the aspects of the project that were successful and effective, while also identifying and reflecting on areas that presented challenges and/or could have been handled differently. This will provide valuable insights into the most critical system design considerations.

Radar

The radar MIMO principle has been applied to obtain the azimuth angle of the approaching drone. The results of four different DOA estimation methods were presented in chapter 4. The angle FFT method and the Bartlett beamformer have similar performance providing a peak at the true DOA. The Capon beamformer proved inferior, possibly due to coupling between the radar antenna elements and the small number of snapshots used in computing the sample covariance matrix. Improving the estimation of the covariance matrix, either by increasing the number of snapshots or by non-coherently summing across the appropriate velocity bins, is expected to enhance the performance of the Capon beamformer. The MUSIC algorithm provided DOA estimates with sharp peaks, clearly indicating the effectiveness of the method. The MUSIC algorithm is suitable in the application of the Archangel system. Utilising the camera detection to obtain and number of targets, the parametric MUSIC algorithm can then be applied. For all four DOA estimation methods the data point resolution, i.e. the interval between data points, of one degree is fairly low. Increasing the resolution will provide more consistent measurements, which will improve the accuracy of the drone position estimation.

An additional improvement to radar detection involves obtaining the elevation angle alongside the azimuth angle. A straightforward approach is to add a second radar of the same type, rotated by 90°. Alternatively, a custom PCB board incorporating the radar should be developed. This approach also allows for the addition of more antenna elements, improving angle resolution. This should be considered based on the required radar capabilities, particularly in scenarios involving multiple attacking drones.

The limiting factor in the Kalman fusion scheme is the radar detection range. The attack angle implications on the detection range of the radar has been presented. Improvements to the detection range is a priority and is a necessary improvement to the

system. The addition of a rotated radar will improve the detection range as the radiation pattern has a wider 3 dB drop-off point. Another possibility is amplifying the transmit power of the radar. The radar equation relates the range and the transmit power and can provide numbers on the required transmit power for a desired range.

Camera

The focus of this project was not to investigate the vision aspect of the Archangel system. However, it is noteworthy that the use of a GoPro without a wide-angle recording format has improved the drone detection significantly. Drone detections are obtained at 40 m instead of 11 m as touched upon in [2].

Data fusion

The Kalman fusion scheme highlights the importance of data fusion and the implementation of a tracking system for the Archangel project. Significant opportunities remain untapped for this specific project and the future development of the system.

The issue of attenuated state estimation must be resolved. In a real-world scenario, where the drone flies directly towards the system, this problem will be less significant. However, the system should still be capable of handling such cases effectively. A possible solution is estimating the heading of the drone and adding a weighting to the indices related to the velocities in the observation model $\mathbf{H}$, i.e. at each iteration updating the observation model. With this implementation, the Kalman fusion will give equal weight to velocity and position measurements only when the drone flies directly towards the system and the defined transformation 5.7 is valid. Otherwise, it prioritises position estimates.

Transforming the spherical measurements from the radar and camera to Cartesian coordinates has significantly improved the model, as evidenced by the predictions of the model when no measurements are fed into the filter. However, using the extended Kalman filter offers additional benefits. The extended Kalman filter does not require transforming the spherical measurements, as it linearises the inputs. It should also address the issue associated with the non-direct approach of the drone.

Temporal aspects

Project data handling was entirely offline. The real-time implementation was not pursued due to the need for further investigation into the specific data handling board,

DCA1000EVM, as well as the lack of expertise in managing packet loss in such systems. This was not a priority, as developing a custom PCB will address these issues. However, the methods applied and algorithms developed in the project are compatible with real-time implementation; the computations in the signal processing chain take less time than the allotted interval for processing 25 frames per second.

Temporal synchronisation between the two sensors during data collection should be implemented. In this project, synchronisation was achieved using a visual cue visible on both sensors. However, a more reliable hardware solution should be developed for future implementations.

6.1 Future work

The immediate future improvements should focus on increasing the radars detection range, e.g. employing the double radar approach or amplifying the transmit power of the radar system. Moreover, obtaining elevation DOA estimates from the radar. Additionally, utilising the extended Kalman filter or weighting of the velocity measurements using the iteratively updated observation model will significantly improve the position and velocity estimates of the drone.

Furthermore, applying a temporal synchronisation along with a physical coordinate system synchronisation enables more reliable measurements and a general improvement in the data fusion scheme. Additional improvements include handling multiple targets and the training of the object detection algorithm related to the camera drone detection.

7 | Conclusion

The Archangel system has demonstrated a clear potential through a radar and camera sensor fusion strategy achieved with a Kalman filter for drone detection. This project has highlighted the capabilities of the system and also revealed areas for further improvement.

The addition of DOA estimation through applying the radar MIMO principle has proved successful. The feasibility of different estimation methods has been investigated and presented. The Kalman filter fusion strategy has been modified, making use of a spherical to Cartesian transformed coordinate system. This implementation effectively models the linear movement of the drone as indicated when no measurements are produced. The addition of radar DOA estimation provides multi-sensor measurements, i.e. obtaining the azimuth angle from both the radar and the camera, to the fusion scheme and increases the stability and reliability of the state estimates.

Further critical improvements remain to enable effective and real-time drone detection and localisation. Radar improvements such as estimating the elevation angle and applying multiple radars or amplifying the transmit power to facilitate a longer detection range prove essential. Improvements to the Kalman filter include employing the extended Kalman filter. This along with multi-target considerations and a real-time implementation are required for an effective operation of the Archangel system.

In conclusion, while the feasibility of the modified Archangel system has been established, continued development and optimisation is essential to meet the demanding requirements of real-time drone neutralisation. The insights gained from this project offer valuable guidance for improving the system's performance and reliability in operational environments.

7.1 Reflection on company stay

The *multiple-input multiple-output radar for drone detection and localisation* project has been a solo project at Nordic Wing, investigating a potential spin-off product for the company. This means that no other employees have worked on it, which has limited my work-related communication with other employees. However, I have always been able to get critical information relating to the application of the Archangel system product. Furthermore, the location of the company at a former/closed-down airbase provided

easy testing and data collection enabling a painless evaluation of the developed methods. Nordic Wing has offered a good work environment with the necessary equipment for developing and testing in relation to this project. Even though I have not myself had a lot of work-related communication with the employees, I have been involved in and felt part of the daily life and activities of the company, not least due to high levels of communication between departments. Nordic Wing is a small and growing company characterised by a lot inter-departmental collaboration and communication. All in all, the project-oriented study was a successful and educational experience, and has provided me with valuable insights into working at a company.

Bibliography

[1] Nordic Wing. *Nordic Wing website*. 2024. URL: https://nordic-wing.com.

[2] Aske Alstrup et al. *Detection and localisation of drones for use in anti-suicide drone system*. Aalborg University, 2024.

[3] Sandeep Rao. *MIMO Radar*. July 2018. URL: https://www.ti.com/lit/an/swra554a/swra554a.pdf?ts=1729493720365.

[4] P. Stoica and R. Moses. *Spectral Analysis of Signals*. Prentice Hall, 2004.

[5] Sandeep Rao. *Introduction to mmwave Sensor: FMCW Radars*. July 2020. URL: https://www.ti.com/content/dam/videos/external-videos/zh-tw/2/3816841626001/5415203482001.mp4/subassets/mmwaveSensing-FMCW-offlineviewing_0.pdf.

[6] Wikipedia. *MUSIC (algorithm)*. Feb. 2024. URL: https://en.wikipedia.org/wiki/MUSIC_(algorithm)#cite_note-schmidt-4.

[7] Texas Instruments. *xWR1843 Evaluation Module (xWR1843BOOST) Single-Chip mmWave Sensing Solution*. May 2020. URL: https://www.ti.com/lit/ug/spruim4b/spruim4b.pdf.

[8] Texas Instruments. *DCA1000EVM Data Capture Card*. May 2019. URL: https://www.ti.com/lit/ug/spruij4a/spruij4a.pdf?ts=1713851136375.

[9] Texas Instruments. *AWR1843 Single-Chip 77 to 79GHz FMCW Radar Sensor*. 2024. URL: https://www.ti.com/lit/ds/symlink/awr1843.pdf?ts=1713872734680&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FAWR1843.

[10] Texas Instruments. *MMWAVE-STUDIO*. May 2020. URL: https://www.ti.com/tool/MMWAVE-STUDIO?utm_source=google&utm_medium=cpc&utm_campaign=epd-rap-null-58700008489213089_mmwave_studio_rsa-cpc-evm-google-eu_en_int&utm_content=mmwave_studio&ds_k=mmwave+studio&DCM=yes&gad_source=1&gbraid=0AAAAAC068F3GXQeU0fg-P17ov0udlD6VD&gclid=CjwKCAiAyJS7BhBiEiwAyS9uNdKnG3M7UOQMiAUgnElQxgiwji8rH_sE26xUAED3MhVqHoygyQbj-xoCGvMQAvD_BwE&gclsrc=aw.ds.

[11] Radarsimx. *Radarsimpy*. Seen 03-04-2024. URL: https://github.com/radarsimx/radarsimpy.

[12] Wikipedia. *Moving target indicator*. July 2022. URL: https://en.wikipedia.org/wiki/Moving_target_indication.

[13] Mark A. Richards. *Fundamentals of Radar Signal Processing, Third Edition*. McGraw Hill, 2022.

[14] DJI. *PHANTOM 4 Quick Start Guide*. 2016. URL: https://dl.djicdn.com/downloads/phantom_4/en/Phantom_4_Quick_Start_Guide_v1.2_en_160317.pdf.

[15] GOALZERO. *YETI 500X*. Dec. 2024. URL: https://goalzero.eu/dk/product/goal-zero-yeti-500x/#product-description.

[16] GoPro. *HERO 11 BLACK*. URL: https://gopro.com/en/us/shop/cameras/hero11-black/CHDHX-111-master.html?srsltid=AfmBOooH8jckfujpGSL0vvKk0Hfv2PAZZb8DMD\allowbreakikvdDSTaXwLYXH3WCo.

[17] Torben Knudsen. *Kalman Filter lecture notes [LN]*. 2024.

[18] Wikipedia. *Kalman filter*. Dec. 2024. URL: https://en.wikipedia.org/wiki/Kalman_filter.

[19] Ultralytics. *YOLO11*. Nov. 2024. URL: https://docs.ultralytics.com.

[20] Christopher K. Wikle and L. Mark Berliner. "A Bayesian tutorial for data assimilation". en. In: *Physica D* 230.1-2 (June 2007), pp. 1–16.

[21] Mahbub Hassan Azade Fotouhi Ming Ding. *Understanding Autonomous Drone Maneuverability for Internet of Things Applications*. 2017. URL: https://research.csiro.au/isp/wp-content/uploads/sites/106/2016/08/Understanding-Autonomous-cameraready-2017.pdf.

[22] Wikipedia. *Delta method*. Nov. 2024. URL: https://en.wikipedia.org/wiki/Delta_method.

[23] Wikipedia. *Block matrix*. Dec. 2024. URL: https://en.wikipedia.org/wiki/Block_matrix.

[24] Alberto Leon-Garcia. *Probability, Statistics, and Random Processes for Electrical Engineering, Third Edition*. Pearson, Prentice Hall, 2008.

[25] Wikipedia. *Multivariate normal distribution*. Dec. 2024. URL: https://en.wikipedia.org/wiki/Multivariate_normal_distribution.

A | Logbook

An overview of the work performed each week is displayed in table A.1.

Week	MIMO	Data collection	Fusion	Writing
36				
37				
38				
39				
40				
41				
42				
43				
44				
45				
46				
47				
48				
49				
50				
51				
52				
1				

Table A.1: Overview of logbook content.

Week 36

2nd of September and onwards.
Startup at Nordic Wing will be the 23rd of September due to them not being inland at the moment. Therefore the first couple of weeks will be working from home. This first week was primarily about getting an overview and acquiring and studying relevant literature to further development of the Archangel project. I had a meeting with my supervisors Troels and Anders who assisted me with this and provided me with the radar. I have been reading about beamforming in [4] and about the Kalman filter from a Bayesian approach in [20] to gain a better understanding of the Kalman filter for estimating confidence region in estimating the position of the drone.

Week 37

This week I have evaluated the radar code developed from previous semester. I have developed a revised code regarding the signal processing of the data from the radar; a new implementation of CA-CFAR. This implementation is written by using the convolution principle to achieve faster computation time enabling an online implementation. Previously, the radar signal processing that is the input to the Kalman filter was able to compute $\simeq 7$ frames each second, now around $\simeq 130$ frames/s is achieved.

I have visited Jonas (the CEO of Nordic Wing) in Støvring to have a talk about the project and collect a computer for running the `MMWAVE-STUDIO` radar software.

Furthermore, I have looked at implementing the MIMO principle on the data from last semester, to extract the angle components.

Week 38

This week I have I have set up the computer provided by Nordic Wing. This has included setting up the `MMWAVE-STUDIO` radar software and getting familiar with the program; the various settings e.g. chirp parameter settings, chirp manager, data collection and export of the data in an appropriate format (this included the accompanying MATLAB script to obtain 'radar cube'). This further insight into the data handling might (I hope) be beneficial when an online implementation, requiring online data collection, is implemented.

In addition to this, I have continued working with the MIMO implementation. Initially I will acquire the angle estimates by performing a FFT on the peaks of the 2D FFT, i.e.

an angle FFT. Beamforming will also be attempted and compared with this. I had a meeting with Anders discussing the MIMO setup. Applying beamforming and the MUSIC algorithm. It will be good to be able to verify the estimates through a proper setup at Nordic Wing. Thereby, making sure what is the correct result.

Week 39

This week I have implemented the Bartlett beamforming method for DOA estimation as described in [4]. I have tried three different methods which produce different results. This will have to be evaluated based on a simple test to verify the correct result. Furthermore, I have worked on the angle estimation from the angle FFT. This work will be evaluated by testing next week at the facilities at Nordic Wing.

Week 40

This week has been my first week at Nordic Wing. I started there Monday. There has been an introduction to the company and I participated in their weekly meeting. Furthermore, some time has been spent on getting to know the company, setting up my workstation, participating in flying the drone and getting a tour around the facilities at Nordic Wing, which include the R&D department and the in-house production fascilities.

I have performed a ground truth test at Nordic Wing. This test was performed to evaluate the performance of the implemented methods/algorithms for beamforming and the phasor method; collect data to confirm the angle estimations. The test included two data collections: one where I walked away with a metal plate from the radar at bore sight (0 degrees) and towards and one at with the same procedure though at 45 degrees. This test is described in appendix B.

Further work on the implemented methods has been done. I have still only applied 1 TX and 4 RX elements for the estimations, which affect the angle resolution. I would like to implement all TX elements next week. The beamforming method yields somewhat promising results. Comparing the angle estimations there are clear distinctions between the results for the 0 degree and 45 degree test. There are clear trends in the plots relating to the ground truth angle. However, the results for the two tests in many frames also yield the same results, which indicates a problem. Performing padding on the angle FFT method increasing the resolution but not affecting the underlying oscillatory/frequency components also looks promising. I will discuss the results with my supervisors next week.

Week 41

This week I have worked on completing the implementation of the two angle estimation methods: angle FFT and the Bartlett beamformer. I have implemented all TX and RX elements in the test, i.e. implemented the virtual array for MIMO.

Furthermore, I have studied the material about time division multiplexing and the approach when performing this. This includes applying a predetection matrix. the implementation of this unveiled a problem that the target peak indices across RX elements for some frames were not identical. I have obtained results for this across the ULA for TX1 and TX3, only the virtual antenna array elements corresponding to these, since there are double representations if I include TX2.

Moreover, I had a meeting with Anders. The results for the beamforming method show that there are many frames for which the curves were identical and indicates a problem. This meeting gave me insight into a possible reason and a solution for this. I will look at this next week.

Week 42

The beamforming results are now good using Bartlett. The covariance matrix obtained is non-singular and invertible. I will therefore implement Capon next week, which should provide a better results, with a smaller peak in the power spectrum. Moreover, I have investigated the spectrum for the angle FFT. I have implemented the non-linear relationship between spatial frequency and angle.

The goals for next week is to implement Capon, sanity check on the angle FFT, study the MUSIC method, test with actual drone, study results for the drone test. I aim to soon be done with the angle estimation and continue by looking at the real-time / online implementation of the algorithms, revising the data fusion implementation.

Week 43

This week I achieved most of the goals from last week. I performed a sanity check on the angle FFT method. This along with a meeting with Anders made complete sense of the previous irregularities in the results for the angle FFT method. The correct spectrum for the angle estimates are now achieved.

Additionally, the Capon beam former and MUSIC algorithm have been implemented and are yielding excellent results. Thereby, I have now implemented four different angle estimation methods for this project. I also wrote a lot in the work sheet (notes for project report) before moving forward, since I wanted to note the thoughts and insights gathered in relation to the first goal of the project, i.e. the goal of getting angle estimates from the radar of the drone. Therefore, I saved the test and data collected of an actual drone to next week.

Week 44

This week I finished writing the ideas of thoughts of the project down in the worksheet. Furthermore, I revised the python code and the implementation of the beamforming methods in python.

I took the drone certificate. Now ready for the data collection test with the actual drone. There were bad weather conditions Tuesday. I overcame fairly strong winds Wednesday and performed different tests with the drone even though it had trouble handling the winds. Next week studying the results, obtaining range, velocity, and angle estimates.

Week 45

I obtained the spectrum for all frames for the data collected with the drone. I obtained the Range and velocity estimates, and angle estimates from all four methods: angle FFT, Bartlett, Capon, and MUSIC.

I studied some material relating to the real-time implementation. I had a meeting with my supervisors Anders and Troels discussing the obtained results/estimates. We also concluded that I should not do the real-time implementation, since this is not my primary interest and would take some time, and instead focus on the data fusion utilising the Kalman filter. Furthermore, I should perform new tests, with fairly exact physical measurements, to get an idea of the accuracy of the methods. I will work on this next week.

Week 46

The weather was bad Monday and Tuesday (windy and rainy), hence I postponed flying with the drone and collection of data to Wednesday. Instead, I started looking at the vision part and the implementation of YOLO algorithm for object detection. Since this is not an area of interest/focus for this project, I do not want to invest a lot of time in it. I am considering evaluating the data fusion with the data collected last semester: direct translation and easy to see the hopefully expected improvements to the system, by having increased the dimension utilising the MIMO radar principle and implementing a revised Kalman filter.

I spent the entirety of Wednesday performing the new tests. The rest of the week I wrote on the worksheet.

Week 47

This week I started out with identifying an error in the angle estimation calculations: when calculating the covariance matrix I averaged over three velocity bins around the peak value, i.e. ±1, and since this was done non-coherently it naturally affected the phase information. The yielded angle estimates are now a lot "better" and similar across the methods: angle FFT, Bartlett, Capon, and MUSIC. Capon still has some variations, but a lot fewer.

Furthermore, I investigated possible improvements to the CA-CFAR algorithm implementation. This resulted in changing the window size, specifically the number of guard cells in Doppler axis and range axis. This increases the number of detections and the results all in all look very promising now.

I performed a data collection test to obtain a video recording of the drone approaching, not using a wide angle video format, along with radar data, in the case I wanted new data besides the old data from last semester to study the performance of the revised system.

Additionally, I plotted the obtained state estimates of the drone in Cartesian coordinates in the x-y plane to hold against the theoretical/measured path. This looks good.

A coupling problem between the tramsitter elements was also confirmed. Obtaining angle estimates from the radar when only one TX was applied yielded fewer fluctiations in the Capon angle estimates.

Week 48

This week i continued with the work of investigating the parameters to obtain the best target detection using CA-CFAR. This involved studying the threshold and the 2D FFT plot for frames where there were no detections.

Additionally, I worked on the data fusion scheme using the Kalman filter. This involved revising the design: conversion of input to Kalman filter to Cartesian coordinates and new model matrices. Studying the plots of the camera data and radar data in the x-y plane (Cartesian coordinates) uncovered inconsistencies. i.e. the estimated paths of the drone from each sensor were different on the data from previous semester. This is most likely due to a poor test setup, where the sensors have not had a common coordinate system (they are turned with respect to each other) and to the massive distortion on video from the camera. I need to perform the vision drone detection on the collected data from last, which was performed exactly in the case of this, where the setup was done correctly and thoroughly, and where the camera is not a 360 wide angle measurement. This should fix the problem.

Week 49 and onwards

The remaining weeks of the project-oriented study involved obtaining angle measurements from the camera og the new data collection, finishing the Kalman filter data fusion scheme and writing the report. My last day at Nordic Wing was in week 51. Some problem were discovered when developing the Kalman fusion. The inversion of the measurement noise covariance matrix were troublesome due to dependencies between the sensor measurements providing dependent columns in the matrix. This problem was solved by applying the block matrix to force no correlation between these. The Kalman filter works well now. Additionally the confidence regions for the Kalman filter were obtained. Additionally the results were appended to the video obtained from the data collection. The remaining time was spent on finishing the report.

B | Radar angle data collection

Conducted by	Aske Best
Date	01/10/2024

B.1 Test objective

The radar angle test aims to collect ground truth data for verification of the angle estimates produced by implemented methods: angle FFT, the Bartlett and Capon beamformers, and the MUSIC method, i.e. to obtain data to hold the angle estimates against the ground truth.

B.1.1 Test equipment

Equipment	Manufacturer	Model	Specifications
Radar	TI	AWR1843Boost, DCA1000EVM	[7, 8]
PC	Lenovo	Lenovo Legion 17"	||
Power station	Goal Zero	Yeti 500X	[15]

B.2 Setup

The test was carried out to collect data from the radar to confirm the angles estimates (only the azimuth) from the implemented methods. A subject held up a metal plate (good to reflect the signals from the radar) and moved towards the radar at an approximately known azimuth angle. The elevation angle $\phi = 0°$. Note, that the azimuth angles θ are determined observationally and should be considered as approximations. The data collection is performed with two different setups:.

1. Moving towards the radar at bore sight, i.e. $\theta = 0°$
2. Moving towards the radar at $\theta = 45°$

The setup is seen in figures B.1 and B.2.

Figure B.1: Setup for the radar angle test (first view).

Figure B.2: Setup for the radar angle test (second view).

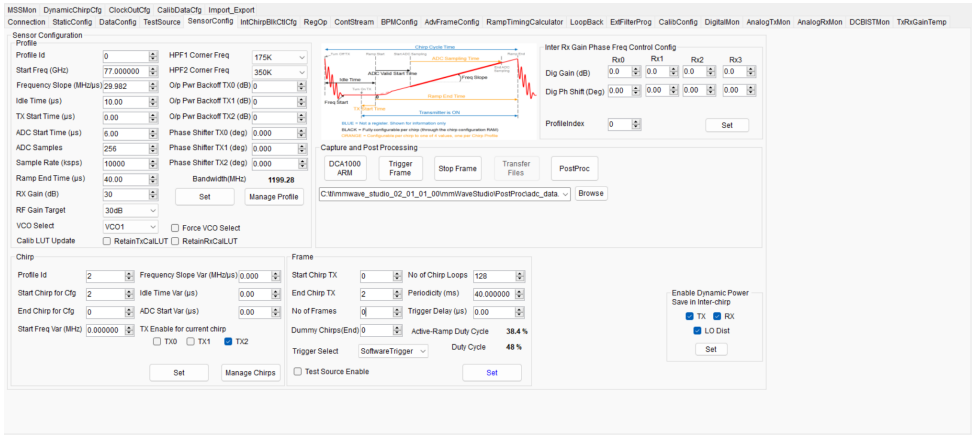


Figure B.3: Settings for the radar in mmWave – studio

B.3 Procedure

When conducting the test the area was secured. The procedure of the test was the following:

1. The metal plate is positioned at desired angle.
2. The radar starts capturing data.
3. The metal plate is moved backward at a constant speed (regular walking speed) away from the radar.
4. Next the reverse: the metal plate is moved forward towards the radar.
5. The movement is stopped close to the radar.
6. Radar recordings are saved.

The resulting data sets are described in section B.4.

B.4 Results

The format of the data collection name designation is: `drone_angle_`θ.

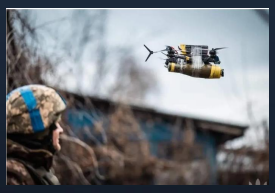
Dataset	Test type	Test description
`radar_angle_0`	Radar angle	Obtaining ground truth data for a target at $\theta = 0^\circ$
`radar_angle_45`	Radar angle	Obtaining ground truth data for a target at $\theta = 45^\circ$

C | Nordic wing Information

Typical targets of engagement

Attach speed 120-210 km/h

Approach vector 30-50 degrees.



Attach speed 20-80 km/h

Approach vector 10-50 degrees.

Requirements for Archangel

Requirements	
Detection range	150m
Coverage	360 degrees
Dead angel in top	Allowed
Max price	500.000 dk
Standalone unit	Yes
Build in power	Not required
Weight	Max 100 kg
Identification	Not required