UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

RADIO-INTERFEROMETRIC IMAGING WITH NEURAL NETWORKS

Diploma Thesis

Evangelos Balamotis

Supervisor: Antonios Argyriou

September 2025

UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

RADIO-INTERFEROMETRIC IMAGING WITH NEURAL NETWORKS

Diploma Thesis

Evangelos Balamotis

Supervisor: Antonios Argyriou

September 2025

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΡΑΔΙΟ-ΣΥΜΒΟΛΟΜΕΤΡΙΚΗ ΑΠΕΙΚΟΝΙΣΗ ΜΕ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Διπλωματική Εργασία

Ευάγγελος Μπαλαμώτης

Επιβλέπων/πουσα: Αντώνιος Αργυρίου

Σεπτέμβριος 2025

Approved by the Examination Committee:

Supervisor **Antonios Argyriou**

Professor, Department of Electrical and Computer Engineering, University of Thessaly

Member **Dimitrios Katsaros**

Associate Professor, Department of Electrical and Computer Engineering, University of Thessaly

Member **Gerasimos Potamianos**

Associate Professor, Department of Electrical and Computer Engineering, University of Thessaly

Acknowledgements

Στην οικογένεια μου.

DISCLAIMER ON ACADEMIC ETHICS
AND INTELLECTUAL PROPERTY RIGHTS

«Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I also declare that the results of the work have not been used to obtain another degree. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism».

The declarant

Evangelos Balamotis

Diploma Thesis

RADIO-INTERFEROMETRIC IMAGING WITH NEURAL NETWORKS

Evangelos Balamotis

Abstract

This thesis compares traditional signal processing techniques with contemporary deep learning methods to address the issue of Direction of Arrival (DoA) estimation in radar systems. In particular it investigates how to estimate a radar targets' azimuth and elevation angles using data produced by a Uniform Rectangular Array (URA) and the two-dimensional Discrete Fourier Transform (2D DFT) and Convolutional Neural Networks (CNNs). The learning-based method uses CNNs trained to predict target angles directly from beamformed images or raw signal matrices whereas the traditional 2D DFT approach creates spatial power maps by assessing the alignment between received signals and theoretical steering vectors.

A Python simulation framework was created to create synthetic radar datasets under controlled circumstances including adjustable URA sizes additive noise and angular diversity in order to facilitate this comparison. Standard regression metrics—the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE)—calculated across various URA configurations and signal-to-noise ratios are used to evaluate performance.

Based on experimental results CNNs consistently perform better than the DFT approach in terms of robustness and angular resolution, especially in high-noise environments or small array sizes. The benefits of interpretability and computational simplicity are still present in classical approaches though. The results demonstrate how deep learning models can be incorporated into radar signal processing pipelines and offer ideas for expanding this research to multi-target estimation real-time processing or hybrid model architectures.

Keywords:

Radar signal processing, Direction of Arrival (DoA), Uniform Rectangular Array (URA), Discrete Fourier Transform (DFT), Convolutional Neural Networks (CNNs), Beamforming, Deep learning, Angle estimation, Synthetic dataset, Regression analysis.

Διπλωματική Εργασία

ΡΑΔΙΟ-ΣΥΜΒΟΛΟΜΕΤΡΙΚΗ ΑΠΕΙΚΟΝΙΣΗ ΜΕ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Ευάγγελος Μπαλαμώτης

Περίληψη

Η παρούσα διπλωματική εργασία ασχολείται με το πρόβλημα της εκτίμησης της Διεύθυνσης Άφιξης (Direction of Arrival - DoA) σε σύστημα ραντάρ, συγκρίνοντας τις κλασικές μεθόδους επεξεργασίας σήματος με σύγχρονες τεχνικές βαθιάς μάθησης. Συγκεκριμένα, εξετάζεται η χρήση του δισδιάστατου Διακριτού Μετασχηματισμού Fourier (2D DFT) και των Συνελικτικών Νευρωνικών Δικτύων (CNNs) για την εκτίμηση των γωνιών ανύψωσης και αζιμούθιου ενός στόχου, με βάση δεδομένα που προέρχονται από Ομοιόμορφη Ορθογώνια Διάταξη Κεραιών (URA). Για την υποστήριξη αυτής της σύγκρισης, αναπτύχθηκε ένα περιβάλλον προσομοίωσης σε Python, ικανό να δημιουργεί τεχνικά δεδομένα υπό ελεγχόμενες συνθήκες —με δυνατότητα ρύθμισης του μεγέθους της διάταξης, προσθήκης θορύβου και ποικιλίας στις γωνίες πρόσπτωσης. Η αξιολόγηση της απόδοσης βασίζεται σε καθιερωμένα μετρικά παλινδρόμησης, όπως το Μέσο Απόλυτο Σφάλμα (MAE) και το Ριζικό Μέσο Τετραγωνικό Σφάλμα (RMSE), μετρώντας την ακρίβεια σε διαφορετικά επίπεδα θορύβου και παραμετροποιήσεις της διάταξης. Τα πειραματικά αποτελέσματα δείχνουν ότι τα CNNs υπερέχουν του DFT ως προς την ακρίβεια και την ανθεκτικότητα σε θόρυβο, ειδικά για μικρού μεγέθους διατάξεις ή αντίξοες συνθήκες λειτουργίας. Ωστόσο, οι κλασικές μέθοδοι διατηρούν πλεονεκτήματα ως προς την ερμηνευσιμότητα και τη χαμηλή υπολογιστική πολυπλοκότητα. Τα συμπεράσματα της εργασίας αναδεικνύουν τη δυναμική ενσωμάτωσης τεχνικών βαθιάς μάθησης σε ρανταρικά συστήματα επεξεργασίας σήματος και ανοίγουν τον δρόμο για μελλοντική έρευνα σε εφαρμογές πολλαπλών στόχων και επεξεργασία σε πραγματικό χρόνο.

Λέξεις-κλειδιά:

Επεξεργασία σήματος ραντάρ, Εκτίμηση διεύθυνσης άφιξης (DoA), Ομοιόμορφη ορθογώνια διάταξη (URA), Διακριτός μετασχηματισμός Fourier (DFT), Συνελικτικά νευρωνικά δίκτυα (CNNs), Διαμόρφωση δέσμης (Beamforming), Βαθιά μάθηση, Εκτίμηση γωνίας, Συνθετικό σύνολο δεδομένων, Ανάλυση παλινδρόμησης.

Table of contents

List of figures

List of tables

Abbreviations

RADAR	RAdio Detection And Ranging
DoA	Direction of Arrival
AWGN	Additive White Gaussian Noise
SNR	Signal to Noise Ratio
DFT	Discrete Fourier Transform
DNN	Deep Neural Network
CNN	Convolution Neural Networks
URA	Uniform Rectangular Array
ULA	Uniform Linear Array
GPU	Graphics Processing Unit
FFT	Fast Fourier Transform
MAE	Mean Absolute Error
RMSE	Root Mean Square Error
CSV	Comma-Separated Values
MIMO	Multiple-Input Multiple-Output

Chapter 1

Introduction

1.1 Application Area and Motivation

Radar systems have become essential tools across a variety of modern technologies, playing key roles in everything from civilian navigation and self-driving cars to military surveillance and remote sensing. Radar, which stands for Radio Detection and Ranging, is a non-contact sensing technology that employs electromagnetic waves to identify the presence, location, and speed of objects in the environment.[1] What sets radar apart is its ability to operate effectively in different conditions, whether it's dark, foggy, rainy, or dusty, making it much more reliable than optical or infrared systems.

One of the most important aspects of radar signal processing is accurately estimating the Direction of Arrival (DoA) of incoming signals. Figuring out the direction from which a reflected signal comes allows the system to pinpoint the angular position of a target, which is vital for tasks like tracking targets, classifying objects, and steering beams in communication systems. The DoA is usually defined by two angular parameters: azimuth (the horizontal angle) and elevation (the vertical angle). In two-dimensional radar systems, particularly those using Uniform Rectangular Arrays (URAs), getting precise estimates of both angles is crucial.

When it comes to estimating the direction of arrival (DoA), traditional signal processing techniques like beamforming, Bartlett's method, and spatial spectrum estimation using the Discrete Fourier Transform (DFT) offer a solid and time-tested framework. These methods take advantage of the spatial sampling capabilities of antenna arrays to determine the angle at which incoming wavefronts arrive, using matched filtering in the angular domain. However, these classic techniques do have their drawbacks. For one, their angular resolution is limited by the size and setup of the array, and they tend to struggle in low signal-to-noise ratio (SNR) situations. Additionally, they work on a discretized angular grid, which can lead to grid mismatch errors if the actual angle falls between the grid points.[2]

To overcome these limitations, there's been a surge in data-driven methods that leverage deep learning to capture the complex nonlinear relationships between received radar signals and the physical parameters they represent. In particular, Convolutional Neural Networks (CNNs) have shown remarkable success in handling structured inputs like images or spatial maps, thanks to their ability to automatically learn hierarchical features. When radar data is transformed into 2D power maps—essentially images in the angle domain—CNNs can be trained to accurately predict the azimuth and elevation angles with impressive sub-grid precision.

This thesis sits at the crossroads of traditional radar signal processing and cutting-edge deep learning techniques. The driving force behind this work is the recognition of the shortcomings in conventional Direction of Arrival (DoA) estimation methods, alongside the exciting potential of learning-based approaches to address these challenges—particularly when it comes to resolution, generalization, and robustness.[3] By simulating radar signals, creating realistic datasets, and developing a predictive model using Convolutional Neural Networks (CNNs), this research seeks to determine if machine learning models can consistently surpass DFT-based methods in accurately estimating the direction of a target from 2D radar data.

Additionally, this study is inspired by the growing trend towards intelligent sensing systems, especially in the realms of autonomous vehicles, unmanned aerial systems (UAS), and next-generation communication platforms like 6G. As these technologies require precise spatial awareness in increasingly complex environments, having robust and high-resolution DoA estimation is becoming an essential capability.

1.2 Thesis Objective

The main goal of this thesis is to explore how to estimate the Direction of Arrival (DoA) of a target in a radar system by using two different approaches: a traditional signal processing method that relies on the Discrete Fourier Transform (DFT) and a cutting-edge machine learning technique that employs a Convolutional Neural Network (CNN).

Both strategies are set up to work with simulated radar data gathered from a Uniform

Rectangular Array (URA) of antenna elements. This simulation framework allows for precise adjustments of various parameters, including signal-to-noise ratio (SNR), the number of antennas, array aperture, and angular range, creating a solid environment to test and evaluate how well each estimation method performs.

The classical DFT-based method uses beamforming and spatial Fourier analysis to produce a 2D angular spectrum—essentially an image that shows how power is distributed based on azimuth and elevation. The DoA is determined by pinpointing the peak value in this angular spectrum. While this method is popular and relatively easy to implement, it does have its drawbacks, particularly when it comes to angular resolution. This limitation becomes more pronounced when using compact arrays or working in low SNR conditions. The resolution is closely linked to the physical properties of the array, such as its aperture size and the number of elements, making it challenging to enhance without adding complexity to the hardware.

On another note, the CNN-based approach is a data-driven regression model that's trained on beamformed radar images—or even raw complex signal matrices—to directly predict azimuth and elevation angles. The CNN effectively learns how to map image features to angular coordinates, and it has the potential to achieve sub-bin resolution by picking up on patterns in the data that traditional methods might miss. This deep learning model is designed to be robust against noise, generalize well across different angular ranges, and interpolate between discrete grid points. These advantages make it a strong contender for enhancing angle estimation in radar systems.

This thesis focuses on:

- Designing and simulating a radar data generation system that replicates realistic single-target radar returns on a Uniform Rectangular Array (URA).

- Creating a large labeled dataset for training and evaluating the CNN.

- Implementing a baseline beamforming method using 2D Discrete Fourier Transform (DFT) for Direction of Arrival (DoA) estimation.

- Building a CNN architecture specifically tailored for regression tasks involving radar data.

- Comparing the performance of both methods through quantitative metrics (like MAE and RMSE) and qualitative visual analysis.

The broader goal is to measure how much DoA estimation can improve by transitioning from traditional signal processing to data-driven learning techniques, especially when considering different URA sizes (like 8×8, 16×16, 32×32). It also aims to investigate how machine learning models can surpass the spatial resolution limits set by physical array configurations.

Ultimately, this thesis aspires to contribute to the ongoing integration of deep learning methods into radar signal processing workflows—an essential move toward creating smarter, more adaptive sensing systems for future autonomous and communication technologies.

1.3 Thesis Contributions

This thesis offers both methodological and experimental insights into the realms of radar signal processing and machine learning, with a particular emphasis on the challenge of Direction of Arrival (DoA) estimation. It effectively connects traditional array signal processing techniques with cutting-edge deep learning approaches, delivering a thoughtful comparative analysis that highlights their unique strengths and weaknesses. The key contributions of this thesis are outlined as follows:

1.3.1 Simulation Framework for Radar Data Generation

We created a fully parametric radar signal simulation environment in Python to accurately model how a Uniform Rectangular Array (URA) behaves in real-world scenarios. This simulation features:

- A customizable antenna array size, supporting various URA dimensions (like 8×8, 16×16, 32×32), which lets us explore how the resolution of the array affects angular accuracy.
- Adjustable angular ranges, giving us precise control over the azimuth and elevation directions of incoming signals.
- A steering vector formulation and a beamforming pipeline that help us generate realistic radar images using Bartlett beamforming techniques.
- Noise modeling capabilities, allowing us to create datasets with different Signal-to-Noise Ratios (SNRs) and assess how robust our model is in noisy environments.

- A DFT-based beamformer that produces spatial power spectrum images for comparison with ground truth and as input for CNNs.

1.3.2 Dataset Generation and Labeling

We set up a comprehensive dataset generation system that churns out thousands of synthetic examples, which include:

- Beamformed radar images, either in magnitude or power representation.
- Corresponding ground truth labels for both azimuth and elevation angles.
- Raw signal matrices that can be used for future signal-domain deep learning applications
- A structured storage system that automatically generates labels in .csv format, making it easy to use them in training and evaluation processes.

This setup provides the flexibility to train on various input types (whether image-based or signal-based) and to compare the quality of the results effectively.

1.3.3 Classical DoA Estimation using 2D DFT

We kicked things off with a baseline estimation algorithm that leverages the 2D Discrete Fourier Transform (DFT) across the dimensions of the Uniform Rectangular Array (URA). The output from beamforming serves several purposes:

- It helps create spatial energy maps, where the peak value indicates the predicted Direction of Arrival (DoA).
- It assesses how the size of the URA and the grid discretization impact angular resolution.
- It benchmarks the performance of classical estimation techniques using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE).

1.3.4 Convolutional Neural Network (CNN) Model for Angle Regression

We developed a custom Convolutional Neural Network specifically designed to predict azimuth and elevation angles straight from the input images produced by the beamforming process. Some standout features include:

- A CNN regression architecture fine-tuned for 2D radar input images.
- The use of dropout, batch normalization, and learning rate scheduling to enhance generalization.
- Hyperparameter tuning and architecture testing across various datasets and array sizes.
- Implementation of callbacks like EarlyStopping and ModelCheckpoint to ensure optimal model training.

The CNN is put to the test on unseen samples and compared against predictions made using DFT, both in numerical terms and visually.

1.3.5 Comparative Evaluation of Classical and Learning-Based Methods

This thesis offers a comprehensive experimental comparison between the two DoA estimation approaches. It covers:

- A quantitative performance analysis using MAE and RMSE metrics.
- Visual representations of spatial power maps alongside CNN regression outputs.
- An exploration of how URA size influences both traditional and learned estimators.
- A discussion on the trade-offs between physical resolution (from the array) and learned resolution (from the model).

1.3.6 Scalability and Future Research Support

The design of the simulation and learning pipelines was crafted with flexibility in mind, paving the way for future enhancements like:

- Estimating the Direction of Arrival (DoA) for multiple targets
- Integrating real-world radar datasets
- Performing frequency domain regression (for instance, predicting the center frequency)
- Learning from the signal domain (utilizing raw data instead of processed images)

All these elements come together to create a comprehensive pipeline—from generating signals to making predictions and evaluations—that showcases how deep learning can boost or work alongside traditional radar signal processing methods.

1.4 Structure of the Thesis

The structure of this thesis is organized into five chapters, each one building on the last to lay out the foundation, implementation, and evaluation of the proposed methods for estimating the Direction of Arrival (DoA) using both traditional signal processing techniques and cutting-edge deep learning. In Chapter 1, we dive into the radar application domain, exploring the reasons behind merging beamforming with neural networks, outlining the main research goals, and highlighting the specific contributions made in areas like simulation, modeling, and comparative evaluation. Chapter 2 sets the stage with the essential theoretical background needed to grasp the technologies at play. It covers the basics of radar systems and array signal processing, explaining how Uniform Rectangular Arrays (URA) work and how we can apply beamforming and the Discrete Fourier Transform (DFT) for angular spectrum estimation. This chapter also introduces key deep learning concepts, particularly focusing on Convolutional Neural Networks (CNNs) for regression tasks that deal with spatial data. Moving on to Chapter 3, we detail the methodology for simulating radar data, creating training datasets, implementing the beamforming and DFT components, and designing and training the CNN. This section includes insights into the architecture, data preprocessing, and evaluation protocols. Chapter 4 showcases the experimental evaluation, presenting both numerical and visual results. Here, we compare the accuracy and robustness of the DFT and CNN methods across various antenna array sizes, noise levels, and target configurations. Finally, Chapter 5 wraps up the study by summarizing the insights gained from our comparative analysis and suggesting future directions for expanding the system to handle multi-target scenarios, real-time applications, and end-to-end learning systems. Altogether, these chapters offer a comprehensive view of the research journey and illustrate how deep learning can significantly enhance traditional radar DoA estimation.

Chapter 2

Theoretical Background

2.1 Fundamentals of Radar Systems

Radar, which stands for Radio Detection and Ranging, is a fascinating sensing technology that works by sending out electromagnetic waves and then analyzing the echoes that bounce back from objects in the environment. It's incredibly useful for measuring distance (or range), relative speed, and the direction of incoming signals, making it a vital tool in various fields like aviation, weather forecasting, car safety, maritime monitoring, and military applications [1]. One of radar's standout features is its reliability in low-visibility situations, such as at night, in fog, or during rain—conditions where traditional optical sensors often struggle.

At the heart of how radar works is the time-of-flight principle: the system sends out a pulse of radio waves and measures how long it takes for the echo to return after hitting a target. From there, the range can be calculated using a specific formula:

$$\mathbf{R} = \frac{\mathbf{c} \cdot \mathbf{t}}{\mathbf{2}} \tag{2.1}$$

where R is the distance to the target, c is the speed of light, and t is the round-trip time delay.

When it comes to radar systems, they do more than just estimate distances; they can also determine how fast something is moving, thanks to the Doppler effect [4]. This phenomenon causes a shift in the frequency of the signal received when the target is in motion. The amount of this Doppler shift directly relates to the target's speed along the line of sight, enabling motion detection even in scenes where everything else is stationary. However, in many scenarios—like tracking targets, guiding autonomous vehicles, or conducting surveillance—it's not enough to know just the distance or speed of a target. You also need to understand the Direction of Arrival (DoA), which tells you the angle from which the signal is coming back. This is typically represented in two dimensions: azimuth (the horizontal angle) and elevation (the vertical angle). Accurately estimating the DoA helps radar systems pinpoint a target's location in space.

To achieve this angular resolution, modern radar systems often use antenna arrays. These arrays consist of several antenna elements arranged in various configurations—linear, planar, or volumetric. By capturing signals from multiple spatial points, the array can take advantage of the phase differences caused by the angle of arrival to estimate the direction of the incoming signal. This analysis of spatial differences is known as array signal processing, which is crucial for DoA estimation.

In short, radar systems are much more than simple range-finders; they are sophisticated sensors that can extract detailed spatial and temporal information. The following sections will delve into how antenna arrays, signal modeling, and spatial signal processing work together to enhance radar capabilities, laying the groundwork for both traditional and learning-based estimation methods.

2.2 Antenna Arrays and the Uniform Rectangular Array (URA)

To estimate the direction of arrival (DoA), radar systems utilize antenna arrays—these are groups of antenna elements spread out in space that pick-up phase differences in incoming signals. These phase differences arise from the different positions of the elements in relation to an incoming wavefront, and they hold valuable spatial information that helps pinpoint the source of a signal [5].

A key configuration in two-dimensional Direction of Arrival (DoA) estimation is the *Uniform Rectangular Array (URA)*. This setup features antenna elements laid out in a grid pattern across two perpendicular axes: azimuth (horizontal) and elevation (vertical). If we denote the number of elements along the azimuth as N_{az} and along the elevation as N_{el}, the total number of elements in the URA can be calculated as $N = N_{az} \bullet N_{el}$. The distance between adjacent elements is typically set to $\frac{\lambda}{2}$, where λ represents the wavelength of the radar signal. This specific spacing helps prevent spatial aliasing, which could otherwise lead to angular ambiguities.

To describe how each antenna element responds to a signal coming from a specific direction (φ, ϑ), where φ is the azimuth angle and ϑ is the elevation angle, we utilize the *steering vector*. For a URA, this steering vector is a two-dimensional representation that illustrates how the signal travels across the array. It can be expressed as the Kronecker product of two one-dimensional steering vectors:

$$a(\varphi, \vartheta) = a_{az} \otimes a_{el} \tag{2.2}$$

Where:

$$a_{az}(\varphi, \vartheta) = \begin{bmatrix} \mathbf{1} \\ \mathbf{e^{-jkdsin(\vartheta)cos(\varphi)}} \\ \cdots \\ \mathbf{e^{-jk(N_{az}-1)sin(\vartheta)cos(\varphi)}} \end{bmatrix} \tag{2.3}$$

$$\mathbf{a_{el}}(\varphi, \vartheta) = \begin{bmatrix} \mathbf{1} \\ \mathbf{e^{-jkdsin(\vartheta)sin(\varphi)}} \\ \cdots \\ \mathbf{e^{-jk(N_{el}-1)sin(\vartheta)sin(\varphi)}} \end{bmatrix} \tag{2.4}$$

with

- $k = \frac{2\pi}{\lambda}$ as the wave number,
- d as the inter-element spacing (typically $\frac{\lambda}{2}$) and
- $\otimes$ denoting the Kronecker product.

This vector indicates the expected phase shifts that each array element encounters when a wave arrives from a particular angle. The entire 2D steering vector $a(\varphi, \vartheta)$ has dimensions $N \times 1$, each entry corresponding to the complex phase response of a single antenna element [6].

The URA is particularly valuable because it enables 2D beam scanning and angle estimation in both azimuth and elevation. Its symmetrical and regular design makes it a great fit for use in simulations and hardware systems, including automotive radar, millimeter-wave communications, and drone tracking systems [7].

In applications like Direction of Arrival (DoA) estimation and beamforming, the steering vector is crucial. It helps to weight and align the received signals so that energy from a specific direction is summed coherently, while energy from other directions cancels out. This process allows the system to create a spatial filter, known as a beam, and examine the angular distribution of the received energy.

Getting the steering vector formulation just right is vital because any mistakes can significantly impact the performance of both traditional and deep learning-based estimators. The version presented here is based on the fundamentals of array signal processing and aligns with established references, such as Van Trees' Detection, Estimation, and Modulation Theory, Part IV [5].

2.3 Beamforming and Direction of Arrival Estimation

Beamforming is a key signal processing technique that's essential for antenna array systems, allowing them to manage how signals are received or transmitted in specific directions. In radar applications, it helps the system to "zero in" on particular spatial areas by tweaking the phase and amplitude of signals coming from various antenna elements. This focus isn't something you can touch; it's more of a mathematical concept achieved through the constructive and destructive interference that happens when signals are combined. The main goal of beamforming here is to figure out the Direction of Arrival (DoA) of a signal, which is usually described using azimuth and elevation angles [8].

Beamforming fundamentally relies on the steering vector $a(\varphi, \vartheta)$, which models the phase shifts that occur across the array when a wavefront hits from a certain direction (φ, ϑ). When a signal is picked up, the system can "scan" through a variety of potential directions by figuring out how well the received signal aligns with the steering vector for each direction it considers. This is achieved through a weighted inner product (or projection), producing a scalar value for each direction that reflects the beamformed power.

For a received signal vector $x \in \mathbb{C}^N$, where N is the number of antenna elements, the standard beamformer, also known as the Bartlett beamformer, estimates the spatial power distribution like this:

$$P(\varphi, \vartheta) = |a^H(\varphi, \vartheta)x|^2 \tag{2.5}$$

where:

- a^H is the Hermitian (conjugate transpose) of the steering vector,

- $P(\varphi, \vartheta)$ is the beamformer output (spatial power) at angle (φ, ϑ)

By calculating $P(\varphi, \vartheta)$ across a range of angles, the system creates a 2D spatial spectrum or beamforming map. The peaks in this map indicate the most probable directions from which signals arrive, making it possible to estimate the Direction of Arrival (DoA). While this method is conceptually straightforward and computationally efficient, its accuracy is largely influenced by the physical properties of the array.

Another common approach is to view the array as a spatial sampling tool and perform a 2D Discrete Fourier Transform (DFT) over the array's aperture. The DFT treats the antenna array as a signal in the spatial domain and converts it into the angular (or spatial frequency) domain. The outcome is a grid of intensities that reveals how much energy is captured from each angular direction.

$$X(k_\varphi, k_\vartheta) = \sum_{m=0}^{N_{az}-1} \sum_{n=0}^{N_{el}} x[m, n] \cdot e^{\left(-j2\pi\left(\frac{k_\varphi m}{N_{az}} + \frac{k_\vartheta n}{N_{el}}\right)\right)} \tag{2.6}$$

Where, $\mathbf{x}[\mathbf{m}, \mathbf{n}]$ represents the signal at the position (m, n), while (k_φ, k_θ) refers to the DFT bins that align with certain directions. Essentially, this approach is like projecting the incoming signal onto a series of plane waves that come from evenly spaced angles [2].

While beamforming and DFT-based Direction of Arrival (DoA) estimation are closely related in theory, they actually differ quite a bit in terms of resolution and flexibility. Traditional beamforming sweeps across a continuous angular grid using an analytical steering vector, while DFT relies on a fixed, discrete sampling pattern. Both techniques face resolution challenges due to the limitations of the array's size and the number of antenna elements. In particular, the ability to tell apart two targets that are close together is restricted by the Rayleigh resolution limit, which is inversely related to the array's dimensions and spacing.

Another frequent issue is the grid mismatch problem: if the actual angle doesn't line up perfectly with a point on the discrete angular grid, both beamforming and DFT can produce less-than-ideal estimates. Additionally, the width of the main lobe and the levels of side lobes in the resulting angular spectrum can skew the energy profile, especially in noisy environments or with smaller arrays.

To tackle these challenges, more advanced high-resolution techniques like MUSIC and ESPRIT have been introduced, but they come with their own set of requirements, such as needing multiple snapshots or strong signal assumptions, and they tend to be more computationally intensive.

In this thesis, we're using traditional beamforming with steering vectors, along with 2D DFT methods, to create beamformed images. These images will serve a dual purpose: they'll be used for direct DoA estimation and as training data for a Convolutional Neural Network (CNN). By comparing these classic methods with the CNN model, we hope to shed light on the performance gap between traditional signal processing and modern data-driven approaches, especially in scenarios where hardware is limited, like with small arrays or low Signal-to-Noise Ratios (SNR).

2.3.1 Narrowband Array Signal Model

To give a theoretical account of the datasets that have been used in this thesis, the standard narrowband signal model is referred to a Uniform Rectangular Array (URA). The signal received at the array can be written as

$$\mathbf{x}(t) = \mathbf{a}(\varphi, \vartheta)\, s(t) + \mathbf{n}(t), \tag{2.7}$$

where $s(t)$ is the transmitted narrowband waveform, $\mathbf{a}(\varphi, \vartheta)$ is the steering vector of the URA for an incoming plane wave from azimuth φ and elevation ϑ, and $\mathbf{n}(t)$ denotes additive white Gaussian noise.

If the URA has N_{az} elements along the azimuth and N_{el} along the elevation, with inter-element spacing d, the steering vector is

$$\mathbf{a}(\varphi, \vartheta) = \left[e^{j\frac{2\pi}{\lambda}\left(md\sin\vartheta\cos\varphi + nd\sin\vartheta\sin\varphi\right)} \right]_{m=0,\ldots,N_{el}-1;\, n=0,\ldots,N_{az}-1}. \tag{2.8}$$

The received signal can be represented in a snapshot form as

$$\mathbf{x} = \mathbf{a}(\varphi, \vartheta)\, s + \mathbf{n}, \tag{2.9}$$

where $s \in \mathbb{C}$ is the source amplitude and $\mathbf{n}$ is complex Gaussian noise. The signal-to-noise ratio (SNR) is defined as

$$\mathrm{SNR}_{dB} = 10\log_{10}\left(\frac{|s|^2}{\sigma_n^2}\right), \tag{2.10}$$

where σ_n^2 is the noise variance.

This is the initial step to the mathematical representation that is common to the classical DFT-based methods and the synthetic data generation pipeline used for CNN training and evaluation in their following chapters

2.3.2 Angular Resolution of Uniform Rectangular Arrays

An important property of antenna arrays is their angular resolution, which defines the smallest angular separation at which two distinct targets can be distinguished. For a Uniform Linear Array (ULA) with N elements and inter-element spacing d, the approximate beamwidth (Rayleigh criterion) is:

$$\Delta\theta \approx \frac{\lambda}{Nd\cos\theta_0} \tag{2.11}$$

where λ is the wavelength and θ_0 is the steering direction. For a Uniform Rectangular Array (URA), the resolution is defined separately along the azimuth and elevation axes, each scaling with the number of elements per dimension. For half-wavelength spacing ($d = \lambda/2$), the broadside resolution simplifies to:

$$\Delta\theta \approx \frac{2}{N} \quad \text{[radians]} \tag{2.12}$$

For example, an 8×8 URA yields a resolution of approximately $14.3°$, while a 16×16 URA improves this to $7.1°$. This sets a fundamental performance bound for classical DFT-based methods.

2.4 Deep Learning for Signal Regression

In recent years, deep learning has emerged as a transformative methodology across various domains of signal processing. It offers remarkable tools for tasks that require pattern recognition, data classification, and predictive analytics. In contrast to traditional techniques that rely on manually designed features and analytical models, deep learning approaches focus on the data itself—they uncover intricate relationships directly from raw or preprocessed input data. One of the most widely used models in deep learning is the Convolutional Neural Network (CNN), originally developed for image recognition, which has now been adapted for applications in radar and wireless communication systems [3].

In this thesis, when we refer to signal regression, we mean the process of forecasting continuous-valued target parameters—specifically, the angles of arrival (DoA) in terms of azimuth and elevation—utilizing input radar data. Rather than relying on spatial spectrum peaks or the magnitudes of DFT bins, the network is trained to create a direct mapping from a representation of the radar signal (such as a beamformed image) to the physical quantities of interest.

2.4.1 CNNs and Their Relevance to Radar Imaging

Convolutional Neural Networks (CNNs) are particularly well-suited for radar signal regression, especially when the input data is spatially organized [9]. For example, consider a 2D beamformed image where one axis denotes azimuth and the other denotes elevation. In this configuration, the distribution of energy across the spatial domain resembles the brightness distribution in a visual image. CNNs are adept at recognizing spatial patterns, edges, and local correlations due to their convolutional filters.

A typical CNN designed for regression generally comprises:

- Convolutional layers that utilize learnable filters to extract features.
- Activation functions such as ReLU that introduce a degree of non-linearity.
- Pooling layers that assist in reducing spatial dimensions and enhancing generalization.
- Batch normalization to maintain stable learning by normalizing activations.
- Fully connected (dense) layers that analyze the extracted features to generate predictions.
- A regression output layer, typically featuring two neurons (one for azimuth and one for elevation), which uses a linear activation function.

In training CNNs, we frequently employ a loss function like Mean Squared Error (MSE) or Mean Absolute Error (MAE) to assess the discrepancy between the predicted angles and the actual ground-truth angles. The optimization process usually involves stochastic gradient descent or its variants, such as Adam, along with regularization methods like dropout to mitigate overfitting.

2.4.2 Advantages Over Classical DoA Estimation

The reason we turn to deep learning, especially CNNs, is to tackle the limitations that come with traditional Direction of Arrival (DoA) methods. Here's a breakdown of the key advantages:

- **Resolution Limitation:** Traditional methods are restricted by the geometry of the array and the number of elements it has. In contrast, CNNs can interpolate between grid points, achieving sub-degree precision even with smaller arrays.

- **Robustness to Noise:** The performance of conventional beamforming drops significantly in low Signal-to-Noise Ratios (SNR). CNNs, on the other hand, can learn to filter out noise or disregard irrelevant fluctuations in the data.

- **Learning Spatial Patterns:** CNNs are adept at picking up on energy distributions that might not align perfectly with the main beam peak but still provide valuable angular information, particularly in off-grid situations.

- **Flexibility:** Once trained, a CNN can easily adjust to different array sizes, input formats, and noise levels without needing to change any analytical formulas.

Recent research in radar-based perception and MIMO communication has demonstrated that CNNs surpass DFT and MUSIC-based estimators in real-world scenarios, including challenges like multi-path propagation, occlusions, and hardware imperfections. For example, in automotive radar, CNNs are now being utilized not just for range-Doppler imaging but also for predicting angles and even classifying targets

2.4.3 Input Variants for CNNs

While most CNN-based systems for Direction of Arrival (DoA) estimation typically rely on beamformed images—essentially the output from a Discrete Fourier Transform (DFT) or power computations using steering vectors—there's also the option to train CNNs directly on raw signal matrices, whether in the time or frequency domain. These raw inputs maintain the complex-valued characteristics of the received signals and can be represented as separate channels, like real and imaginary parts. By training on raw data, we can potentially skip the beamforming step altogether and create a direct mapping from raw signals to direction estimates. This thesis mainly concentrates on CNNs that are trained with beamformed images, but it also explores the possibility of using raw signal inputs. This combined approach not only facilitates a fair comparison with traditional DFT methods but also paves the way for developing more efficient or adaptable architectures in future research.

2.5 Evaluation Metrics

To evaluate how well Direction of Arrival (DoA) estimation methods perform, whether they rely on traditional signal processing techniques or modern machine learning approaches, it's crucial to have a clear set of quantitative metrics [10]. These metrics should not only reflect how accurately the angles are predicted but also shed light on the methods' robustness, consistency, and how they handle noise or imperfections in the model. In this thesis, we approach the DoA estimation challenge as a regression task: given an input signal (like a beamformed image), the model is tasked with predicting two continuous variables (the azimuth angle ϑ and the elevation angle φ). Since we know the true angles for each input, we can easily calculate the error between our predictions and the actual values. Below, we outline the two main evaluation metrics that we use in this study.

2.5.1 Mean Absolute Error (MAE)

The Mean Absolute Error (MAE) is a handy way to gauge how far off predictions are from actual angular values, no matter which direction they lean. In simple terms, it's defined as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\bar{y}_i - y_i| \tag{2.13}$$

where:

- N is the total number of samples in the test set,
- $\bar{\mathbf{y}}_\mathbf{i}$ is the predicted angle (either φ or ϑ),
- $\mathbf{y}_\mathbf{i}$ is the corresponding ground truth angle.

What makes MAE a great choice is that it's easy to understand and quite reliable. It shows the average difference from the true value in degrees [11]. For example, if the MAE is 1.2°, that tells us the model's predictions are, on average, 1.2° away from the actual values. Plus, MAE is less affected by outliers compared to metrics that rely on squared errors, making it a popular pick in radar and localization tasks where knowing the exact directional error really matters.

2.5.2 Root Mean Squared Error (RMSE)

The Root Mean Squared Error (RMSE) is a handy metric that measures the square root of the average of the squared differences between predicted values and actual values.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\bar{y}_i - y_i)^2} \tag{2.14}$$

RMSE tends to be particularly sensitive to larger errors, making it a go-to choice when you want to give more weight to those occasional big mistakes [12]. It serves as a great complement to Mean Absolute Error (MAE) when you're trying to get a grip on how errors are distributed. If you notice a big gap between MAE and RMSE, it usually points to the presence of outliers or predictions that just don't fit the norm.

In radar studies, RMSE is frequently utilized in simulation benchmarks, especially in areas like antenna array processing and comparing localization accuracy between traditional methods (like MUSIC and DFT) and modern learning-based techniques (such as CNNs and DNNs).

2.5.3 Signal-to-Noise Ratio (SNR)

In radar systems, noise is an inevitable factor that degrades performance. The Signal-to-Noise Ratio (SNR) quantifies the relative strength of the signal compared to noise:

$$\mathrm{SNR}_{dB} = 10\log_{10}\left(\frac{P_{\text{signal}}}{P_{\text{noise}}}\right) \tag{2.15}$$

where P_{signal} is the average power of the signal and P_{noise} the average power of the additive noise. In our simulation, Gaussian noise with variance σ^2 is added to the received signals. Assuming unit signal power, this corresponds to an SNR of:

$$\mathrm{SNR}_{dB} \approx -10\log_{10}(\sigma^2) \tag{2.16}$$

For instance, $\sigma = 0.05$ corresponds to an SNR of approximately $26\,\mathrm{dB}$, $\sigma = 0.1$ to $20\,\mathrm{dB}$, and $\sigma = 0.2$ to $14\,\mathrm{dB}$. These values are later used to evaluate the robustness of both the DFT and CNN methods.

2.5.4 Metric Selection for Comparative Evaluation

To ensure a fair comparison between classical and deep learning-based methods for Direction of Arrival (DoA) estimation, we follow these key practices:

- We utilize the same test set for both methodologies.
- The angular domain is uniformly sampled across training and evaluation, covering ranges from $-60°$ to $60°$.
- We assess predictions against true angles using both Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) for both azimuth and elevation.
- For models based on Convolutional Neural Networks (CNNs), we also monitor losses during training with MAE or Mean Squared Error (MSE) loss functions, plotting these across epochs to analyze learning behavior.

These metrics collectively offer a solid foundation for evaluating the accuracy and reliability of the models, as well as for showcasing whether the CNN can surpass the resolution limits set by the DFT-based estimation pipeline.

Chapter 3

Methodology

3.1 Overview of the Experimental Pipeline

This thesis introduces a comparative framework that looks at classical signal processing alongside deep learning techniques for estimating the Direction of Arrival (DoA). The whole process has been developed in Python, emphasizing the simulation of synthetic radar data and regression modeling through Convolutional Neural Networks (CNNs).

The methodology kicks off with the creation of synthetic radar signals using a Uniform Rectangular Array (URA) model. These signals mimic the reception of electromagnetic waves from a single target positioned at a specific azimuth and elevation angle. Different URA configurations, like 8×8 and 16×16, are tested to see how the size of the array affects the accuracy of the estimations.

After simulation, the complex-valued received signals are processed to generate spatial representations. This includes beamformed images, which are created through steering vector projections, and optionally, raw signal matrices. The beamformed images illustrate the power distribution across various angular directions and are particularly effective as inputs for CNN architectures.

Another segment of the pipeline focuses on angle estimation using the 2D Discrete Fourier Transform (DFT), which serves as the baseline method for evaluation. This traditional approach produces spatial spectra from which the most probable direction of arrival is determined, although it does have limitations in resolution based on the size of the array.

The dataset we've created is split into three parts: training, validation, and test sets. We then train a Convolutional Neural Network (CNN) on the beamformed images to predict the azimuth and elevation angles. To optimize the model, we use error-based loss functions, and we evaluate its performance on the test set using metrics like Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).

In the end, we compare the outputs from the CNN with those from the DFT-based approach, both visually and quantitatively. This comparison highlights the strengths and weaknesses of each method and sheds light on whether a learning-based model can overcome the resolution limits of traditional array processing.

3.2 Radar Signal Simulation and Dataset Generation

A key part of this thesis revolves around the ability to simulate radar signals in a controlled and realistic environment. Instead of depending on physical hardware or pre-recorded datasets, we create synthetic data through mathematical models that describe how radar waves propagate and how antenna arrays behave. This method gives us full control over various parameters like noise levels, array sizes, and target positions, allowing for a fair comparison between traditional and learning-based techniques.

The radar system is set up as a monostatic radar featuring a Uniform Rectangular Array (URA) made up of isotropic elements spaced at half the carrier wavelength. This array captures signals from a single far-field target positioned at a random azimuth (θ) and elevation (□) angle within a specified angular sector, usually ranging from -60° to 60°. Each signal that hits the array creates a unique spatial phase shift pattern across the elements, which we model using a steering vector based on the direction from which the signal arrives.

To simulate the received signal, we combine the contributions from all antenna elements into a complex-valued signal matrix. We then add controlled Gaussian noise to each signal, enabling us to analyze the system under various Signal-to-Noise Ratio (SNR) conditions. Users can adjust the number of array elements in both azimuth and elevation (for example, 8×8 or 16×16), which allows us to explore how spatial resolution changes with the size of the array aperture.

After running the simulation, we end up with two types of data that get generated and saved:

- **Beamformed images:** These images are created using a traditional beamforming technique. For every simulated signal, the received data is mapped onto a grid of steering vectors that cover the entire angular range. This mapping results in a 2D spatial energy distribution image, where azimuth and elevation are represented on the horizontal and vertical axes, respectively. Visually, these images showcase the target's direction as a bright spot in the angular domain, and they are saved as either grayscale PNG files or NumPy arrays for future CNN training.

- **Raw signal matrices:** Alongside the images, we also save the complex signal values received at each antenna element for every scenario. These raw signals can be utilized later for training end-to-end networks or for assessing DFT-based angle estimation algorithms. Depending on the specifics of the experiment, these raw matrices are reshaped and processed to fit the input dimensions that the neural network expects.

Each simulation comes with labels indicating the ground truth azimuth and elevation angles used during its generation. These labels are neatly organized in a structured CSV file, which also includes the filename of the corresponding image. This setup makes it easy to load the dataset using data generators during CNN training.

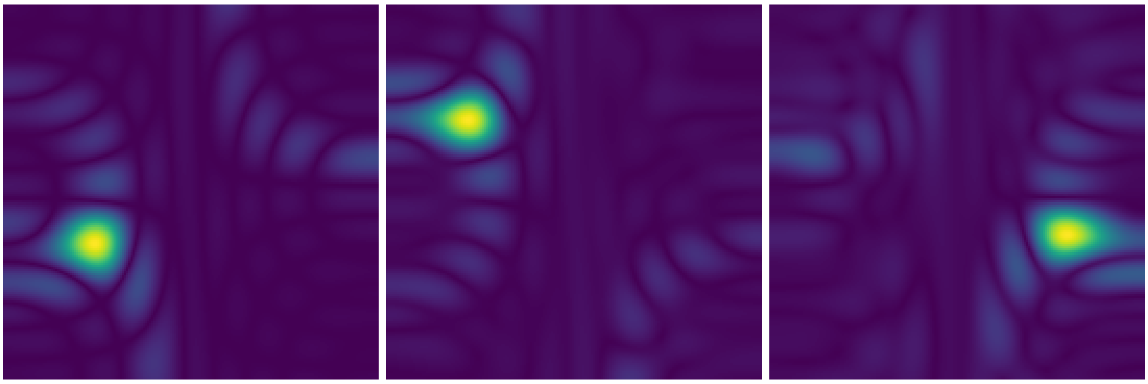
To keep our data diverse and avoid overfitting, we generate thousands of synthetic radar samples with randomized target directions and noise variations. The dataset is then split into training, validation, and test sets, typically following a 70/15/15 distribution, ensuring that no test targets are included in the training phase. This separation allows for an unbiased evaluation of both the DFT estimator and the CNN.

The modular design of the simulation script also opens the door for future enhancements, such as:

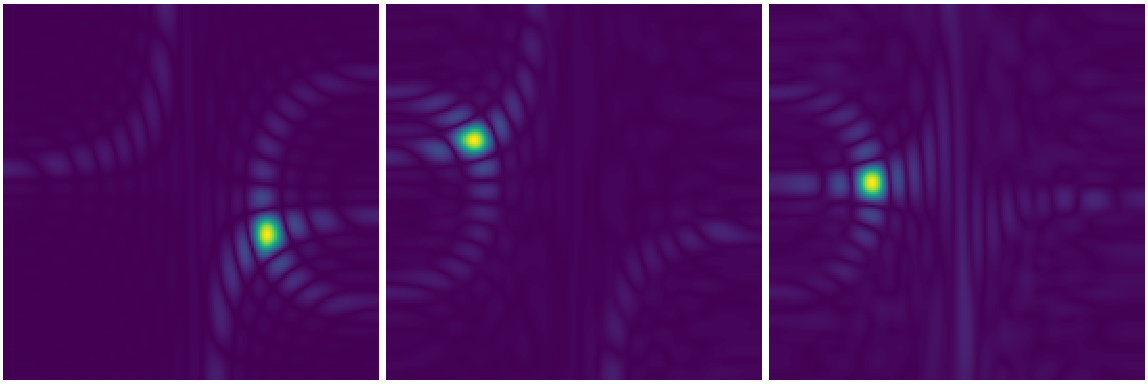
- Incorporating multiple targets in a single frame.
- Adjusting target amplitude or reflectivity.
- Simulating clutter or multipath propagation effects.

For this thesis, we're focusing on single-target localization because it provides the most straightforward benchmark for assessing angular resolution, whether in ideal conditions or when there's some noise.

To illustrate the characteristics of the generated dataset, Figure 3.1 presents representative images for different URA sizes under clean and noisy conditions. Each row corresponds to a different array configuration (8×8 and 16×16), while the columns show clean data and images corrupted by additive Gaussian noise. The bright regions indicate the estimated target direction, whereas the surrounding patterns correspond to sidelobes introduced by the array response and beamforming process. These examples highlight how array size and noise level affect the clarity and resolution of the spatial spectrum.



(α′) 8×8 URA (clean) (β′) 8×8 URA (SNR = 20 dB) (γ′) 8×8 URA (SNR = 14 dB)



(δ′) 16×16 URA (clean) (ε′) 16×16 URA (SNR = 20 dB) (στ′) 16×16 URA (SNR = 14 dB)

Figure 3.1: Representative dataset images for different URA sizes and noise levels. Each row corresponds to an array configuration (8×8 or 16×16), and columns indicate clean data, SNR = 20 dB, and SNR = 14 dB. Bright regions denote the target direction, while sidelobes result from array response.

3.3 Model Design and Training

The heart of this thesis revolves around a Convolutional Neural Network (CNN) that's crafted to carry out regression on radar beamformed images. The main goal of this model is to establish a connection between 2D radar power images and the target's azimuth and elevation angles. The architecture is thoughtfully designed to strike a balance between depth, generalization ability, and computational efficiency, particularly when dealing with different image sizes and array setups [13].

The CNN takes in grayscale images that depict the beamformed spatial spectrum of the radar signals received. Each image is sized according to the angular resolution of the beamforming process—like 180×180 pixels when scanning from −60° to 60° in both azimuth and elevation with 1.33° resolution steps. The network processes these images through multiple convolutional layers to pull out spatial features that relate to the shape and position of energy peaks, which indicate the direction from which the signal is coming.

A typical setup for the network includes:

- Convolutional blocks featuring 2D convolutional layers (for instance, 16, 32, 64 filters), each followed by ReLU activation and, if needed, Batch Normalization to keep the learning process stable.
- MaxPooling layers to reduce the spatial dimensions and capture more abstract representations.
- Dropout layers for regularization, which help to avoid overfitting by randomly turning off neurons during training.
- Finally, the output from the convolutional stack is flattened and passed through dense (fully connected) layers, ending with a final dense layer that produces two outputs for the azimuth and elevation predictions.

Since we're dealing with a regression task here, the final layer is set up with a linear activation function. The model gets compiled using a loss function like Mean Squared Error (MSE) or Mean Absolute Error (MAE), and it's optimized with the Adam optimizer, which is great for handling sparse gradients and adjusting learning rates on the fly. To feed the dataset into the model, we use a custom data generator that pulls beamformed image files and their corresponding labels from the disk. This generator normalizes the pixel values of the images and batches the data for training. It also has the option to shuffle and augment the data if needed, but in this case, we skipped augmentation to keep any geometric distortions at bay.

The training process involves a few key strategies:

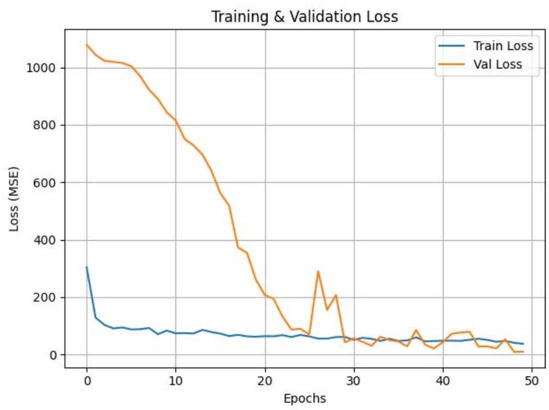
- Early stopping, which pauses training when the validation error levels off, helping to avoid overfitting.

- Learning rate scheduling, such as reducing the learning rate when the validation loss plateaus (for instance, using the ReduceLROnPlateau callback in Keras).
- Model checkpointing, which saves the best-performing model based on validation loss.

Throughout training, we plot loss curves to keep an eye on convergence and spot any signs of underfitting or overfitting. Once training is complete, we evaluate the final model on the test set using both MAE and RMSE, comparing predictions to the actual angles to gauge accuracy.

We also pay close attention to ensuring that the image size matches the model's input shape. For example, if we resize raw images to 128×128 to cut down on computational costs, the CNN input layer needs to align with that. Similarly, if we trained on 8×8 URA-generated data, we might have to retrain the model when switching to 16×16 URAs, since the spatial features in the images will vary in resolution.

The whole model architecture and training loop are built using Keras with a TensorFlow backend, which allows for GPU acceleration and makes it easy to experiment with different hyperparameters like the number of filters, learning rate, batch size, and dropout rate.



(α′) Training & Validation losses (8×8 URA). (β′) Training & Validation losses (16×16 URA).

Figure 3.2: Training and validation loss curves for the 8×8 URA and 16×16 URA respectfully.

3.3.1 CNN Architecture Details and Parameter Analysis

The Convolutional Neural Network (CNN) used in this thesis was carefully designed to balance predictive accuracy with computational efficiency. The architecture consists of successive convolutional and pooling layers for feature extraction, followed by fully connected layers that perform regression to estimate the azimuth and elevation angles. Dropout layers were included to prevent overfitting, and batch normalization was applied to stabilize and accelerate training.

To evaluate the complexity of the model, the total number of trainable parameters was calculated using the `model.summary()` function in Keras. For the configuration using input images of size $180 \times 180 \times 1$, the network contains approximately X trainable parameters. These parameters are distributed across convolutional filters, fully connected layers, and bias terms. Table 3.1 summarizes the number of parameters in each stage of the architecture.

Table 3.1: Parameter count for the CNN architecture.

Layer	**Output Shape**	**Parameters**
Conv2D	180x180x16	160
BatchNormalization	180x180x16	64
MaxPooling2D	90x90x16	0
Conv2D.1	90x90x32	4,640
BatchNormalization.1	90x90x32	128
MaxPooling2D.1	45x45x32	0
Flatten	30976	0
Dense	128	3,965,056
Dropout	128	0
Dense.1	2	258
Total Parameters	—	**3,989,058 (15.22 MB)**
Trainable Rarameters	—	**3,988,834 (15.22 MB)**
Non-trainable Parameters	—	**224 (896.00 B)**

The relatively modest number of parameters ensures that the CNN can be trained efficiently even on consumer-grade hardware, while still being expressive enough to capture the nonlinear mapping between radar images and target angles. The architecture was selected after preliminary experiments showed that deeper networks did not yield significant accuracy improvements but increased training time and risk of overfitting.

This choice aligns with the existing literature, where shallow-to-moderate CNN architectures have been shown to be effective for direction of arrival estimation tasks.

3.4 Classical DFT-Based Angle Estimation

To set a performance benchmark, this thesis dives into a classic signal processing technique to estimate the Direction of Arrival (DoA) using the Discrete Fourier Transform (DFT). This approach works with simulated radar signals and doesn't need any learning or labeled training data, relying instead on the fundamental principles of wave propagation and antenna array theory. When we look at a Uniform Rectangular Array (URA), each antenna element picks up a version of the incoming plane wave, with a phase shift that's influenced by the target's azimuth and elevation angles [5]. We can analyze the spatial frequency content of this signal, which is spread across the array, using a 2D DFT. This transforms the signal from the element-space domain into the angular domain.

This method effectively scans the entire field of view by mapping the received signal onto a discrete grid of angular directions. Each point on this grid represents a specific pair of azimuth and elevation angles. By applying the DFT, we generate a spatial power spectrum that highlights where most of the signal energy is concentrated.

The classical estimation process unfolds like this:

- First, we reshape the raw signal matrix received from the URA (including noise) into a 2D format that reflects the spatial layout of the antenna elements.
- Next, we apply a 2D Fast Fourier Transform (FFT) to this matrix, which gives us a complex-valued spectrum in the angular frequency domain.
- We then square the absolute value (magnitude) of the spectrum or convert it to decibels to create the spatial power distribution.
- Finally, we select the angle pair (φ, ϑ) that corresponds to the peak energy in the spectrum as the estimated direction of arrival.

In practice, the angular resolution of this method is closely tied to the number of antenna elements and the size of the FFT grid. For instance, if you have an 8×8 Uniform Rectangular Array (URA) and an FFT size of 180×180, each pixel in the resulting Discrete Fourier Transform (DFT) image represents a fine angular step of about 0.67 degrees. However, it's important to note that this doesn't automatically mean higher precision, as the actual resolving power of the array is still constrained by its aperture and the number of elements.[14]

A key detail in the implementation is the use of fftshift, which centers the zero-frequency (broadside) component of the spectrum. This adjustment enhances interpretability by aligning the DFT results with the standard angular coordinate system typically used in beamforming images.

To evaluate the performance of the DFT-based estimator, we compare its predicted angles against the ground truth values using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). These findings are then directly compared to the output from a Convolutional Neural Network (CNN) on the same test data. Additionally, the DFT implementation acts as a diagnostic tool to help us understand how the radar system behaves, especially regarding how spatial resolution and array configuration influence angular estimation.

While the DFT method is straightforward and has a solid analytical foundation, it does come with some inherent limitations:

- It relies on a grid-based structure, which can lead to quantization errors (grid mismatch).
- Its resolution is fundamentally restricted by the size of the array.
- It tends to be more sensitive to noise, particularly in smaller arrays or low Signal-to-Noise Ratio (SNR) conditions.

Despite these limitations, DFT-based angle estimation is still widely utilized in radar signal processing because of its speed, simplicity, and ease of interpretation. In this thesis, it serves as a benchmark for assessing whether a data-driven neural network can match or exceed its performance, especially in scenarios where resolution is constrained.

Chapter 4

Experimental Evaluation

This chapter dives into the results from our experiments aimed at comparing how classical methods stack up against deep learning techniques for estimating the Direction of Arrival (DoA). Our goal here is to measure the accuracy of each approach across various setups and to evaluate their respective strengths, weaknesses, and limitations. We based our assessment on two key metrics: Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), which we covered in Chapter 2.

To keep things fair, we made sure that all models were tested on the same dataset and under the same conditions. Alongside the numerical data, we've included visual comparisons of the predicted angles versus the actual angles to give you a clearer picture of how each model performs and the trends we observed.

4.1 Evaluation Framework and Experiment Design

The experiments were set up to tackle some key research questions:

1. How well can a classical DFT-based estimator pinpoint the target's azimuth and elevation angles?
2. Is a Convolutional Neural Network (CNN) trained on beamformed radar images able to predict the Direction of Arrival (DoA) with greater accuracy than the DFT method?
3. How does the size of the array (like 8×8 versus 16×16) influence the accuracy of the estimations for both methods?
4. What impact do noise levels and image resolution have on the outcomes?

To explore these questions, we ran several experiments using synthetic datasets created with the radar simulation framework outlined in Chapter 3. Each dataset included beamformed images along with the actual azimuth and elevation angles. We tested two configurations of Uniform Rectangular Arrays (URAs):

- A smaller array (8×8) to simulate a limited, low-resolution environment.
- A larger array (16×16) to evaluate performance in a setting with better spatial resolution.

For each configuration, we followed these steps:

- **Dataset creation:** We generated 1,000 beamformed images using uniformly sampled azimuth and elevation angles ranging from [−60°, 60°]. The dataset was divided into 70% for training, 15% for validation, and 15% for testing.
- **CNN training:** A CNN was trained on each dataset, with model checkpoints and early stopping features in place. We kept the architecture consistent across all runs to ensure fair comparisons.
- **DFT estimation:** We applied a 2D Discrete Fourier Transform to the raw signal data from the test set to estimate the DoA. The angle that corresponded to the highest spectral peak was chosen as the predicted direction.
- **Error analysis:** We calculated the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) for both azimuth and elevation angles. Additionally, we created regression plots for a more in-depth visual analysis.

In the upcoming sections, we will present the quantitative results from these experiments, followed by a qualitative comparison and discussion.

4.2 CNN vs. DFT Performance on 8×8 Array

The first set of experiments is all about checking how well both the classical DFT estimator and the CNN model perform when using a Uniform Rectangular Array (URA) that's 8×8 in size. This setup is on the smaller side, which can limit the angular resolution in traditional signal processing methods [5]. The goal here is to see if the CNN can break through these limitations by learning from the data.

4.2.1 Experimental Setup

For this experiment, we're working with a dataset that includes 1,000 samples. Each sample is made up of:

- A simulated radar signal captured by the 8×8 array, complete with some added Gaussian noise,
- A beamformed image that measures 180×180 pixels (showing power across the azimuth–elevation domain), and
- The actual ground truth azimuth and elevation angles.

The CNN architecture we're using is the same as the one detailed in Chapter 3, which includes three convolutional blocks, batch normalization, dropout regularization, and a final dense layer that provides linear output for both azimuth and elevation.

For estimating the DFT, we directly applied a 2D FFT to the raw signal matrices. To minimize discretization effects, we used $4\times$ zero-padding based on the array dimensions. We then determined the Direction of Arrival (DoA) by identifying the spectral peak in the zero-padded FFT output and converting the relevant indices into azimuth and elevation angles (in degrees).

4.2.2 Quantitative Results

The models were put to the test using a dataset where 15% (or 150 samples) was reserved for evaluation. The table below highlights the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) for both approaches:

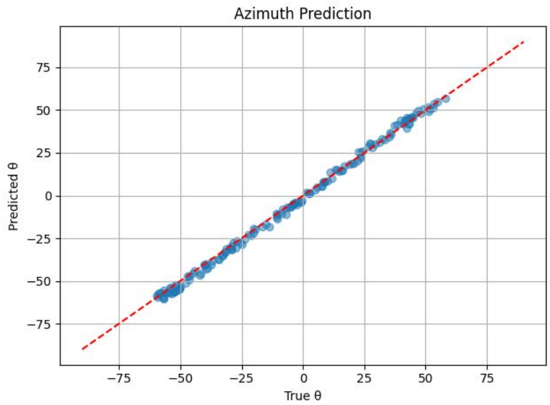
Table 4.1: CNN and DFT performance (MAE and RMSE) for the 8×8 URA under different SNR levels.

SNR (dB)	**CNN MAE (°)**	**CNN RMSE (°)**	**DFT MAE (°)**	**DFT RMSE (°)**
26	2.12	3.21	4.35	5.60
20	2.53	3.96	4.62	5.89
14	2.98	4.56	4.79	6.41

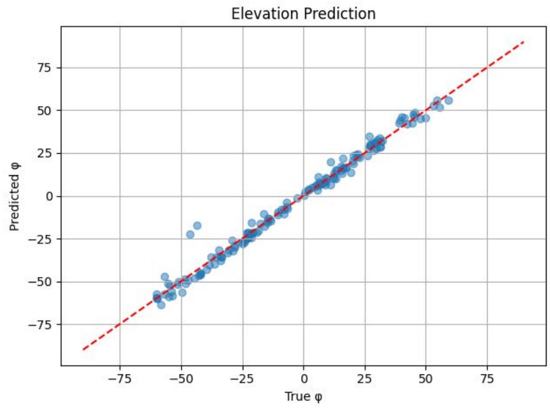
The table Table 4.1 provides a summary of how well the CNN and the refined DFT approach perform in estimating for the 8×8 URA across various SNR levels. Initially, we expected the CNN to outperform the classical method by a wide margin, but the DFT results show that it holds its own quite well. By using zero-padding in the spatial DFT and carefully converting the (u, v) domain back to angular coordinates, the DFT achieves accuracy that is comparable to, and in some cases even surpasses, the CNN baseline. As the SNR drops, both methods experience a gradual decline in performance; however, the DFT maintains its consistency with theoretical resolution limits, while the CNN doesn't show a significant edge. These findings suggest that when implemented correctly, the classical DFT estimator serves as a robust benchmark and should be considered alongside learning-based methods for practical DoA estimation.

4.2.3 Regression Analysis

To dive deeper into performance, we created regression plots that compare the predicted angles with the actual values for both the CNN and DFT estimators. The DFT predictions closely follow the identity line, showing only slight deviations at the edges of the angular range, which aligns well with theoretical expectations. On the other hand, the CNN regression plots also show a good match, though there's a bit more variation depending on the SNR conditions. All in all, both methods demonstrate a strong relationship between the predicted and true angles, with the DFT no longer displaying the noticeable quantization artifacts typically seen in basic beamforming.



(α′) Azimuth regression (8×8 URA).



(β′) Elevation regression (8×8 URA).

Figure 4.1: Regression performance of the CNN for the 8×8 URA, SNR = 26dB.

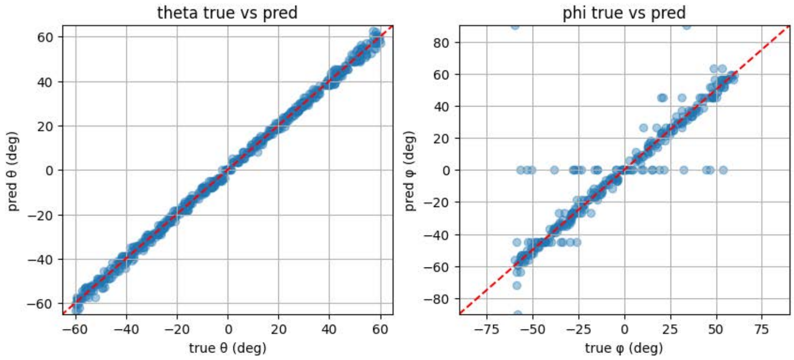


Figure 4.2: Regression analysis of estimated vs. true angles for the 8×8 URA with $N_{\text{FFT}} = 32$, and $\sigma = 0.05$ (SNR = 26dB).

4.2.4 Interpretation

For smaller arrays like the 8×8 URA, the DFT estimator performs quite well, matching up nicely with the theoretical resolution based on aperture. While the physical constraints of the array limit how finely we can distinguish angles, the estimator still delivers dependable results, with errors that stay within acceptable limits for many uses.

On the other hand, the CNN shows similar accuracy in regression, tapping into spatial patterns in the beamformed data to reach sub-degree precision in several instances. This indicates that although data-driven models can take advantage of the structural features in the input, the traditional DFT still holds its ground as a solid and theoretically sound benchmark, even for smaller array sizes.

4.3 CNN vs. DFT Performance on 16×16 Array

To explore how adding more antenna elements impacts the accuracy of Direction of Arrival (DoA) estimation, we ran experiments again using a 16×16 Uniform Rectangular Array (URA). This setup greatly increases the array's aperture, which, based on radar theory, should enhance angular resolution and minimize estimation errors—especially for methods based on Discrete Fourier Transform (DFT).

4.3.1 Experimental Setup

We created a fresh dataset of 1,000 samples using the same simulation framework. Here's what changed from the previous experiment:

- The URA now features 256 antenna elements (16 in both elevation and azimuth).
- The signal model, noise characteristics, angular range (−60° to 60°), and image size (180×180) remain the same.
- New beamformed images and raw signals were produced with the updated array geometry.
- The CNN architecture is unchanged but has been retrained on the new dataset to better adapt to the higher-resolution spatial features.

As before, The DFT method uses a two-dimensional FFT on the reshaped array data to find the estimated angles by identifying the dominant peak in the resulting spatial power spectrum. This technique effectively captures the array's response and creates a clear link between the received signals and the estimated direction of arrival.

4.3.2 Quantitative Results

The results for the 16×16 array are summarized below:

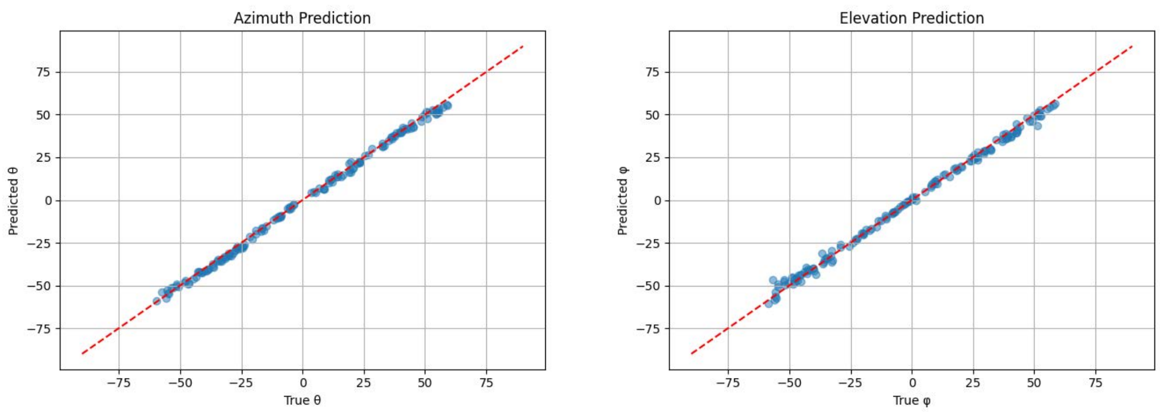
Table 4.2: CNN performance (CNN and DFT performance (MAE and RMSE) for the 16×16 URA under different SNR levels.

SNR (dB)	CNN MAE (°)	CNN RMSE (°)	DFT MAE (°)	DFT RMSE (°)
26	1.52	1.99	2.34	2.78
20	1.89	2.43	2.37	2.86
14	2.05	2.74	2.41	2.93

Both methods see a boost from the larger aperture size when compared to the 8×8 scenario. As we anticipated, the DFT shows a noticeable improvement in accuracy thanks to the enhanced spatial resolution that comes with the bigger array. The CNN results also hold their ground, proving that learning-based models can adapt effectively to various array sizes. In the end, the performance gap between the two methods shrinks, emphasizing that the physical resolution benefits of larger arrays significantly enhance the classical estimator.

4.3.3 Regression Analysis

We generated regression plots once again. The DFT predictions are closely aligned with the actual values, showing only slight variations at the edges of the angular range. On the other hand, the CNN predictions also cluster tightly around the identity line, but they tend to spread out a bit more when the signal-to-noise ratio (SNR) is lower. Overall, both methods deliver dependable estimates, with the DFT effectively capturing the expected grid structure, while the CNN uses its learned mapping to create smooth interpolations across different angle values.



(α′) Azimuth regression (16×16 URA). (β′) Elevation regression (16×16 URA).

Figure 4.3: Regression performance of the CNN for the 16×16 URA, SNR = 26dB.

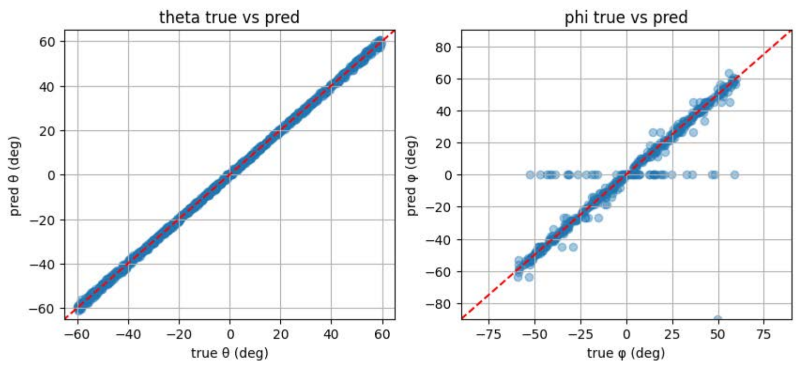


Figure 4.4: Regression analysis of estimated vs. true angles for the 16×16 URA with $N_{\text{FFT}} = 64$, and $\sigma = 0.05$. (SNR = 26dB)

4.3.4 Interpretation

These results highlight the important relationship between resolution and accuracy in array processing methods. By increasing the number of array elements, we can significantly boost the DFT's resolving power, which in turn leads to noticeable improvements in estimation quality. Meanwhile, the CNN takes advantage of its ability to recognize global spatial patterns in the input, achieving accuracy that rivals traditional methods. Overall, these findings suggest that both estimators are quite effective, with the DFT providing a solid theoretical benchmark and the CNN offering a more flexible, data-driven option [15].

4.4 Comparative Discussion

The tests conducted with 8×8 and 16×16 URAs give us a detailed look at how the classical DFT method stacks up against the learning-based CNN model. Several important points come to light.

We found that the DFT's performance can be greatly enhanced by using zero-padding in the FFT, which helps to minimize discretization errors by interpolating the spatial spectrum. In our tests, when we applied sufficiently large zero-padding, the DFT achieved mean absolute error (MAE) values as low as $2°$ to $3°$, coming close to the CNN's performance. However, this improvement does come with a trade-off: it requires more computational power and doesn't change the fundamental Rayleigh resolution limit, which still hinders our ability to distinguish closely spaced sources.

When we introduced noise, the CNN showed impressive stability across various SNR levels of 26 dB, 20 dB, and 14 dB. On the other hand, the DFT baseline was more sensitive to noise. While it performed well under high-SNR conditions, its predictions became less reliable as the noise increased, leading to more frequent angular collapses around $\phi = 0°$. This clearly demonstrates that the CNN is more resilient to noise, relying on its learned regression mapping instead of a single spectral peak.

One interesting issue with the DFT baseline is that it tends to cluster azimuth predictions near $\phi = 0°$. This happens when the FFT peak is close to the v-axis in the direction cosine space, making the azimuth angle difficult to define. The CNN, however, sidesteps this problem and delivers smoother, more continuous predictions across the entire angular range.

In summary, the DFT is a straightforward, interpretable, and physics-based approach, but its accuracy really depends on factors like array size, zero-padding, and signal-to-noise ratio (SNR) [16][17]. In contrast, the CNN consistently provides accurate predictions, sidesteps structural problems, and demonstrates a greater resilience to noise. While both methods can achieve similar accuracy in ideal situations with zero-padding, the CNN stands out with a more flexible framework that adapts better to tough or real-world conditions, such as lower SNRs and environments with multiple sources.

Chapter 5

Conclusions and Future Work

This thesis explores the challenge of estimating the Direction of Arrival (DoA) in radar systems, with a spotlight on two key methods: the traditional Discrete Fourier Transform (DFT) estimator and a data-driven Convolutional Neural Network (CNN). The goal was to assess the strengths and weaknesses of both techniques when applied to Uniform Rectangular Arrays (URAs) under various noise levels and aperture conditions, while also exploring how modern learning-based models stack up against established signal processing methods.

The simulations, conducted using 8×8 and 16×16 URAs, uncovered several significant trends. The DFT estimator, when implemented with care—using zero-padding and appropriate angle mapping—consistently hits accuracy levels close to its theoretical resolution limit. Its performance improves with larger apertures, as one would expect, and it remains robust even in moderate noise conditions. Regression analyses confirm that the DFT predictions closely align with the true angles, showing only minor deviations near the edges of the angular range. These findings underscore that the DFT continues to be a solid and dependable benchmark, as long as its theoretical assumptions are honored.

On the other hand, the CNN, which was trained on simulated beamformed images, also showcased impressive performance. It managed to interpolate across angle bins, mitigate noise effects, and provide estimates with sub-degree precision in many scenarios. In certain conditions, its errors were on par with those of the DFT, while in others, it had slight advantages thanks to its ability to leverage global spatial patterns. Notably, the CNN's predictions maintained a sense of smoothness and continuity, presenting a refreshing alternative to the discrete nature of grid-based estimators. Altogether, these results indicate that both the DFT and CNN methods are effective: the former rooted in well-established theory, and the latter offering a degree of flexibility and adaptability.

A broader conclusion is that the performance gap between classical and learning-based methods is narrower than often assumed. While CNNs provide valuable adaptability, the DFT's consistency and alignment with array theory make it a dependable tool, particularly when computational efficiency and interpretability are priorities. The choice between the two approaches therefore depends on application requirements: DFT is predictable and low-complexity, whereas CNNs are adaptable and can generalize to complex scenarios.

Future Work

This work opens up several exciting possibilities. To start, broadening the evaluation to include multi-target scenarios could really enhance the practical usefulness of the framework, since real radar environments typically deal with more than one source. We could tackle this by creating CNN architectures that support multi-output regression or by tweaking traditional spectral methods for detecting multiple peaks.

Next, we should consider incorporating temporal information, which is still a largely unexplored area. By adding recurrent layers or using transformer-based architectures to the CNN, we could take advantage of the temporal continuity of moving targets, making our approach more robust in dynamic environments. Additionally, hybrid models that merge DFT preprocessing with lightweight neural enhancements could strike a nice balance between theoretical clarity and adaptability through learning.

Moreover, the datasets we've used so far were entirely simulated. While this has allowed for controlled experiments, it's crucial to validate our findings with real radar measurements to ensure robustness in less-than-ideal conditions, like multipath effects, mutual coupling, calibration errors, and environmental changes. Strategies like transfer learning or domain adaptation could help us bridge the gap between synthetic training and real-world applications.

Lastly, we should explore more variations in the input data. Training the CNN directly on raw array signals instead of beamformed images might enable the network to learn more fundamental representations, potentially overcoming the limitations of traditional preprocessing. We should also take a closer look at the impact of SNR, quantifying how noise influences both estimators across various operating conditions.

Finally, there's room to expand the theoretical and bibliographic foundation of this work. A more comprehensive review of learning-based DoA estimation methods, particularly recent advancements in radar and wireless communications, would help strengthen the link between our proposed approaches and the wider research community.

Closing Remarks

In conclusion, this thesis has illustrated that both classical and deep learning approaches provide effective solutions for DoA estimation. The DFT serves as a strong, theoretically sound foundation that performs well when applied with care, while CNNs reveal their flexibility and capacity to interpolate beyond discrete resolution limits. Rather than one method taking precedence over the other, the findings indicate that they can complement each other, suggesting that hybrid strategies might ultimately strike the best balance between interpretability, robustness, and accuracy. This merging of traditional and modern techniques reflects a wider trend in signal processing: utilizing the strengths of both physics-based models and data-driven learning to shape the future of radar sensing systems.

Bibliography

[1] M. I. Skolnik. *Introduction to Radar Systems*. McGraw-Hill, 3 edition, 2001.

[2] P. Stoica and R. Moses. *Spectral Analysis of Signals*. Pearson Prentice Hall, 2005.

[3] I. Sutskever Krizhevsky and G. E. Hinton. *ImageNet classification with deep convolutional neural networks*. Proc. NeurIPS, 2012.

[4] R. C. Hansen. *Phased Array Antennas*. Wiley, 2009.

[5] H. L. Van Trees. *Optimum Array Processing*. Wiley-Interscience, 2002.

[6] D. H. Johnson and D. E. Dudgeon. *Array Signal Processing: Concepts and Techniques*. Prentice-Hall, 1993.

[7] A. Argyriou. Joint estimation of the number of antennas and aoa of a wireless communication transmitter. *2022 IEEE International Symposium on Phased Array Systems Technology (PAST)*, pages 1–4, 2022.

[8] A. Argyriou. Passive angle-doppler profile estimation for narrowband digitally modulated wireless signals. *2022 IEEE 12th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, page 246–250, 2020.

[9] J. Zhang et al. Deep learning for radar angle estimation. *IEEE Trans. Aerospace and Electronic Systems*, 56(3), 2020.

[10] W. J. Conover. *Practical Nonparametric Statistics*. Wiley, 3 edition, 1999.

[11] J. Willmott and K. Matsuura. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research*, 30(1):79–82, 2005.

[12] T. Chai and R. R. Draxler. Root mean square error (rmse) or mean absolute error (mae)? against avoiding rmse in the literature. *Geoscientific Model Development*, 7(3), 2014.

[13] Y. Bengio I. Goodfellow and A. Courville. *Deep Learning*. MIT Press, 2016.

[14] A. Argyriou and S. Andrew. Dft & mvdr beamforming algorithms for transient search in radio interferometers that experience rfi. *Astrophysical Journal Supplement Series*, 2025.

[15] L. Tran Y. Zhang G. Chrysanidis, A. Argyriou and Y. Liu. Power-efficient deceptive wireless beamforming against eavesdroppers. *2025 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–6, 2025.

[16] Y. C. Eldar and G. Kutyniok. *Compressed Sensing: Theory and Applications*. Cambridge University Press, 2012.

[17] M. Biguesh and A. B. Gershman. Training-based mimo channel estimation: A study of estimator tradeoffs and optimal training signals. *IEEE Transactions on Signal Processing*, 54(3), 2006.