

Millimetre Wave Data Representation Analysis in Skin Cancer Classification



Master Thesis

Submitted by:

Ravindra Rankawat

Matriculation No.: 1531011

Submitted to the Embedded Systems Department
of University of Siegen

May 2024

First Supervisor:

Prof. Dr. Roman Obermaisser, Universität Siegen

Second Supervisor:

Dr. Daniel Onwuchekwa, Universität Siegen

MSc. Fatima Mammadova, Universität Siegen

Declaration

I **Ravindra Rankawat** hereby declare that this thesis project entitled "**Millimetre Wave Data Representation Analysis in Skin Cancer Classification**", submitted for the fulfilment of requirements for the award of **Master of Science in Mechatronics** is the result of my work. It has not been submitted anywhere for any other degree or other purposes. Further, I have faithfully and accurately cited all sources that assisted me throughout my thesis project, including other researchers' work, published or unpublished.

I am aware of the consequences of not recognising any sources used in the respective work, and it is considered fraud and plagiarism.

.....

Ravindra Rankawat

Matr.-nr.: 1531011

May 2024

Acknowledgement

I extend my heartfelt appreciation to all those who have stood by me throughout the journey of crafting this thesis.

My deepest gratitude goes to **Prof. Dr.-Ing. Roman Obermaisser** for allowing me to embark on this academic endeavor.

I wish to express profound thanks to my supervisor, **Dr.-Ing. Daniel Onwuchekwa**, for his invaluable guidance and unwavering support throughout every stage of this process.

Special recognition is due to my advisor, **MSc. Fatima Mammadova**, whose indispensable assistance and continuous support were instrumental in bringing this thesis to completion.

I am also grateful to the Chairs of the Embedded System Department for their backing and encouragement.

To my family and friends, your unwavering encouragement has been a constant source of strength and motivation, guiding me through every challenge I faced. This accomplishment would not have been attainable without the dedicated support of these individuals and institutions.

Thank You.

Abstract

Skin cancer classification poses significant challenges and opportunities for advancements in healthcare technologies. This thesis explores the use of machine learning and deep learning models, with a focus on skin cancer classification using mmWave data representations. The primary objective is to evaluate the effectiveness of various classification models across different data representations and feature sets, with an emphasis on identifying the most robust and accurate methods for classification.

This study commences with a collection of mm-Wave recordings, addressing the challenge of limited availability. Despite its small size comprising 44 samples, detailed comprehensive experimental setups using mmWave radar data in both frequency and time domains, along with spectrogram image classification. The findings suggest that ensemble techniques, notably Random Forest and Extra Trees classifiers, consistently exhibit strong performance, showcasing high levels of accuracy, precision, recall, and F1-scores. These models excel in handling complex, high-dimensional data, capturing intricate patterns that simpler models like the Logistic Regression Model, Gaussian Naive Bayes, and Ridge Classifier fail to recognize. However, convolutional neural network (CNN) models used for spectrogram image classification exhibited signs of overfitting, achieving perfect scores on training and validation sets but questionable generalization capabilities.

The high performance of ensemble methods suggests significant potential for deployment in classification tasks where reliable and accurate diagnostics are crucial. Nevertheless, the tendency of deep learning models to overfit underscores the need for improved training strategies and model validation to enhance their applicability and robustness.

In conclusion, this thesis demonstrates the promising capabilities of advanced machine learning techniques in skin cancer classification through mmWave radar and spectrogram data analysis. It lays the groundwork for future research that could extend these models' applicability, ensuring to contribute effectively to patient care.

Contents

Acknowledgement	I
Abstract	II
Table of Contents	III
1 Introduction	1
1.1 Problem Statement	1
1.2 Aim and Objective	2
2 Background Knowledge	3
2.1 Radar Theory	3
2.2 Radar Equations	10
2.2.1 Radar Range Equation	11
2.3 Millimetre Wave Radars	14
2.4 Radar Data Acquisition	16
2.5 Radar Data Representation	20
2.5.1 Raw Frequency Domain Data	21
2.5.2 Raw Time Domain Data	21
2.5.3 Spectrogram Generation	22
2.5.4 Comparison of Data Representation Techniques in mmWave Radar	23
2.6 Artificial Intelligence	25
2.6.1 Introduction to Machine Learning Models	25
2.6.1.1 Logistic Regression Model	26
2.6.1.2 Ridge Classifier	26
2.6.1.3 K-Nearest Neighbors Classifier	27
2.6.1.4 Support vector classifier (SVC)	28
2.6.1.5 Decision Tree Classifier	29
2.6.1.6 Random Forest Classifier	30
2.6.1.7 Gradient Boosting Classifier	31
2.6.1.8 AdaBoost Classifier	32
2.6.1.9 Extra Trees Classifier	33
2.6.1.10 Gaussian Naive Bayes classifier	34
2.6.1.11 Multilayer Perceptron Classifier	35
2.6.2 Deep Learning Model	37
3 Related Works	39

4	Methodology	48
4.1	Data Collection	49
4.1.1	IMAGEVK-74 - The mmWave Radar	49
4.1.2	Data Set	50
4.1.3	Meta-Data	51
4.2	Data Pre-Processing	51
4.2.1	Data Labeling	52
4.2.2	Data Cleaning	52
4.3	Data Representation	52
4.4	Raw Feature	53
4.5	Feature Extraction - Domain Specific	53
4.5.1	Feature Extraction - Frequency Domain	54
4.5.2	Feature Extraction - Time Domain	55
4.5.3	Feature Extraction - Spectrogram	58
4.6	Model Evaluation Metrics	60
4.6.1	Key Performance Metrics	60
4.6.2	Advanced Metrics for Model Performance	61
5	Experiments and Results	63
5.1	Experimental Setup	63
5.2	Raw mmWave Data Classification	67
5.2.1	Frequency Domain Data Classification	67
5.2.1.1	Raw Frequency as a Feature	67
5.2.1.2	Extracted Features: Frequency Domain	70
5.2.2	Time Domain Data Classification	74
5.2.2.1	Raw Time Domain as a Feature	74
5.2.2.2	Extracted Features: Time Domain	77
5.3	Balanced mmWave Data Classification	80
5.3.1	Frequency Domain Data Classification	80
5.3.1.1	Raw Frequency as a Feature	80
5.3.1.2	Extracted Features: Frequency Domain	83
5.3.2	Time Domain Data Classification	86
5.3.2.1	Raw Time Domain as a Feature	86
5.3.2.2	Extracted Features: Time Domain	89
5.4	Spectrogram Image Classification	92
6	Discussions and Conclusion	94
6.1	Implications	95
6.2	Limitations	95
6.3	Future Scope	96

List of Figures

Fig. 2.1	Block diagram of a basic radar that uses a superheterodyne receiver in the bottom part and a power amplifier as the transmitter [72]	4
Fig. 2.2	Attenuation of electromagnetic waves [21]	9
Fig. 2.3	Rain Attenuation of Horizontal Polarization [48]	10
Fig. 2.4	Spherical coordinate system for radar measurements [5].	11
Fig. 2.5	Infinite-length continuous wave (CW) signal of frequency f_0 and its frequency spectrum [68].	18
Fig. 2.6	FMCW radar block diagram [12].	19
Fig. 2.7	FMCW radar fast-time/slow-time coherent processing interval (CPI) data [68].	19
Fig. 2.8	Recording a chirp involves a system equipped with TX and RX antennas. The TX (Transmitter) antennas emit chirps sequentially, while each RX (Receiver) antenna captures the chirp simultaneously. Subsequently, the Digital Signal Processor (DSP) executes a range FFT, and the outcomes are preserved in the radar cube before the subsequent chirp. [7, 68].	20
Fig. 2.9	Spectrogram plot frequency against time with power distribution.	23
Fig. 2.10	Types of Dropout Layers with Fully connected layer [47].	38
Fig. 4.1	Overview of the methodology	48
Fig. 4.2	IMAGEVK-74 - The mmWave Radar [52]	50
Fig. 4.3	Class Distribution of Samples	51
Fig. 4.4	Data Representation	53
Fig. 4.5	Confusion Matrix [41]	60
Fig. 4.6	Flowchart to visualize the structure of algorithm	62
Fig. 5.1	ROC curve for K-Nearest Neighbours	69
Fig. 5.2	Confusion Matrix for Extra Tree Classifier	70
Fig. 5.3	ROC curve for Random Forest Classifier	72
Fig. 5.4	Confusion Matrix for Random Forest Classifier	73
Fig. 5.5	ROC curve for K-Nearest Neighbors	75
Fig. 5.6	Confusion Matrix for Extra Tree Classifier	76
Fig. 5.7	ROC curve for Random Forest Classifier	78
Fig. 5.8	Confusion Matrix for Random Forest Classifier	79
Fig. 5.9	ROC curve for Support Vector Classifier	81
Fig. 5.10	Confusion Matrix for Support Vector Classifier	82

Fig. 5.11 ROC curve for K-Nearest Neighbors	85
Fig. 5.12 Confusion Matrix for Extra Tree Classifier	85
Fig. 5.13 ROC curve for Random Forest Classifier	87
Fig. 5.14 Confusion Matrix for Random Forest Classifier	88
Fig. 5.15 ROC curve for Random Forest Classifier	90
Fig. 5.16 Confusion Matrix for Random Forest Classifier	91
Fig. 5.17 Epoch vs Training and Validation Metrics for Experiment 3	93

List of Tables

Tab. 2.1 Standard radar-frequency letter band nomenclature [31]	7
Tab. 5.1 Hyper-tuning of ML Models	64
Tab. 5.2 Hyperparameters of the Deep Learning model	65
Tab. 5.3 Hyperparameters of Spectrogram Deep Learning model	66
Tab. 5.4 Performance of ML Models on Raw Frequency Data	68
Tab. 5.5 Performance of ML Models on Extracted Frequency Domain Data . . .	71
Tab. 5.6 Performance of ML Models on Raw Time Domain Data	74
Tab. 5.7 Performance of ML Models on Extracted Features of Time Domain Data	77
Tab. 5.8 Performance of ML Models on Balanced Raw Frequency Data	81
Tab. 5.9 Performance of ML Models on Balanced Extracted Frequency Domain Data	84
Tab. 5.10Performance of ML Models on Balanced Raw Time Domain Data . . .	87
Tab. 5.11Performance of ML Models on Extracted Features of Balanced Time Domain Data	89
Tab. 5.12Experiment Results for Spectrogram Classification	92

Chapter 1

Introduction

In skin cancer detection, the utilization of AI remains constrained primarily due to doctors' scepticism regarding the reliability of AI model outputs based on image data. This scepticism stems from the opaque nature of AI models, which provide results without clear explanations, posing challenges in interpretation. Given the high stakes in the medical field, where errors can result in significant loss of life and resources, explaining the rationale behind AI model outputs is crucial for fostering trust among users.

Applying millimetre waves in skin cancer detection can provide another perspective on the data. The reflected signals of cancerous and non-cancerous lesions change accordingly based on the dielectric properties of the lesion. Millimetre waves can provide in-depth information and overcome the challenges faced by image data-based AI skin cancer classification. This thesis endeavours to address this issue by explaining the classification of millimetre wave data Representation in skin cancer, aiming to demonstrate the most precise classification of mmWave data while providing transparent explanations.

Technological advancements and computational methods have opened new avenues for enhancing medical diagnostic processes. Among various technologies, millimetre-waves (mmWaves) have shown promise [13, 37, 80, 82] due to its non-invasive nature and the depth information of data it can provide. This technology's promising ability to classify skin conditions, including various forms of skin cancer, opens up a new field of research that combines healthcare, signal processing, and machine learning.

1.1 Problem Statement

Skin cancer, a prevalent global cancer type, highlights the urgency of early detection for effective treatment. Traditional diagnostic methods like biopsies, while accurate, are

invasive and unsuitable for frequent screening due to procedural risks and patient discomfort. Non-invasive alternatives such as mmWave and spectrograms present potential solutions but necessitate advanced analytical techniques to interpret their complex data accurately. The primary challenge is the effective collection, processing, and classification of this data in the absence of standardized mmWave skin cancer datasets that could reliably rival the accuracy of existing diagnostic methods.

1.2 Aim and Objective

The central objective of this thesis is to investigate and construct efficient machine learning and deep learning models capable of precisely categorizing skin conditions based on mmWave data representation. This endeavor encompasses a comprehensive strategy to grasp the intricacies of mmWave signals and their translation into diagnostic information using sophisticated computational models. The objectives of the study are as follows:

- To investigate the suitability of various machine learning models for the classification of mmWave radar data in both frequency and time domains, focusing on their ability to handle the data's complexity and provide accurate diagnostic predictions.
- To design and evaluate convolutional neural networks (CNNs) for the classification of spectrogram images generated from mmWave data, assessing their performance and potential overfitting issues.
- To assess how traditional machine learning models and advanced deep learning approaches handle the distinct challenges posed by mmWave data, such as its high dimensionality and signal variability.
- To improve the dependability and precision of the predictive models, various techniques such as feature engineering, regularization, and advanced training methodologies are to be used.

This study aims to analyze the different data representations of mmWaves, classifying them using several ML and DL algorithms. The analysis should provide the scope of the mmWave data representation in skin cancer detection. By fulfilling these objectives, the research not only aims to propel advancements in skin cancer detection but also to improve patient care by enabling earlier and less invasive diagnostic methods.

Chapter 2

Background Knowledge

The basic framework for a thorough comprehension of the thesis is established in this part. Since this thesis centers on Millimetre Wave Radar data, it heavily relies on Radar background theory. Radar technology is the basis of this thesis. Now, Radar is quite extensive, with many different aspects and applications. This chapter will go over the RADAR background theory and provide an overview of the key information that can be used in future studies and applications for skin cancer diagnosis.

2.1 Radar Theory

Radar, which stands for Radio Detection and Ranging, operates as a detection system utilizing radio waves to ascertain the distance, angle, or velocity of objects. It can find objects at a distance under almost any condition, providing essential data for various applications ranging from aviation safety to weather forecasting and military operations. Radar systems achieve precise mapping of object locations and movements through the emission of radio waves and subsequent analysis of the reflected echoes [76].

Here is the basic parts of radar and their functions, which can be listed as follows:

1. **Transmitter:** Acts as a power amplifier, generating waveforms tailored to the radar's specific tasks. It can vary in average power output from milliwatts to megawatts, with average power being a more accurate indicator of radar performance than peak power. Most radars use short pulse waveforms to facilitate the use of a single antenna for both transmitting and receiving.

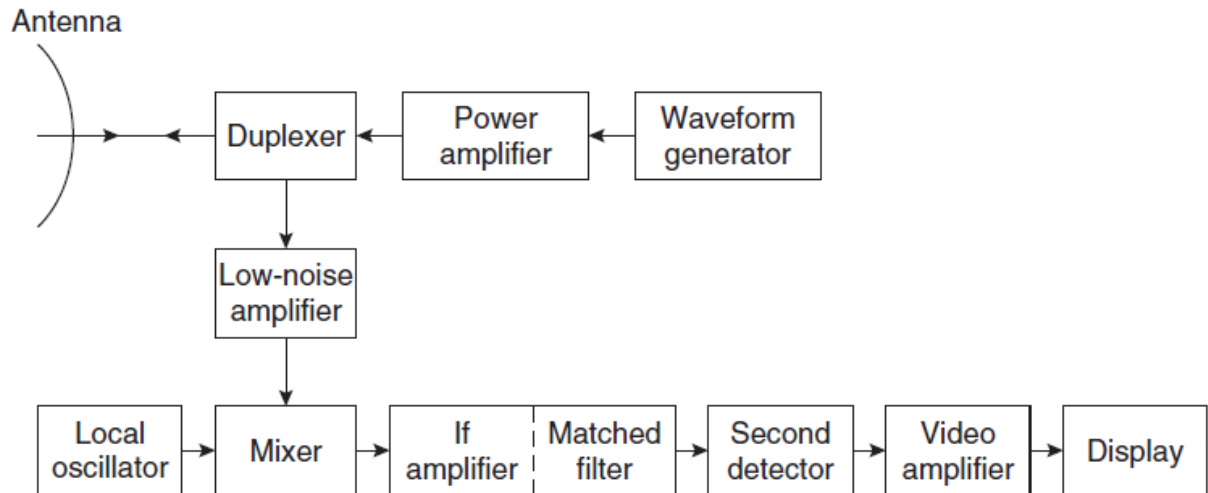


Figure 2.1: Block diagram of a basic radar that uses a superheterodyne receiver in the bottom part and a power amplifier as the transmitter [72]

2. **Duplexer:** Enables the use of a single antenna for both transmitting and receiving by protecting the sensitive receiver from the high power transmission signals. It directs the received echo signals to the receiver and prevents them from going back into the transmitter, safeguarding the system from damage.
3. **Antenna:** Facilitates the propagation of transmitted energy into space and collects echo energy upon reception. Typically directive, focusing radiated energy into a narrow beam to concentrate power and enable target direction determination. It has a large area for receiving to gather weak echo signals and acts as a spatial filter for angle resolution and other capabilities.
4. **Receiver:** Amplifies the weak received signals to detectable levels, with careful design to minimize its own noise production, as noise limits radar's reliable detection capability. Often includes a low-noise amplifier, especially at microwave frequencies where radar typically operates.
5. **Dynamic Range of Receiver:** Defined as the ratio of the maximum to minimum input signal levels over which the receiver maintains specified performance. It's crucial for avoiding saturation from clutter echoes, which can hinder detection of moving targets.
6. **Signal Processor:** Located typically in the Intermediate Frequency (IF) portion of the receiver, it separates desired signals from undesired ones, enhancing detection processes. Incorporates matched filtering for optimal signal-to-noise ratio and doppler processing for improved signal-to-clutter ratio, aiding in distinguishing moving targets from clutter.

7. **Detection Decision:** Made based on whether the receiver output exceeds a preset threshold. The threshold is carefully set to balance between false alarms from too low settings and missed detections from too high settings, aiming for an acceptable average rate of false alarms [20].
8. **Data Processing and Target Tracking:** Following detection, the target's track, or its location over time, is determined. The data can be presented to an operator, utilized for missile guidance, or subjected to further analysis for in-depth target assessment.
9. **Radar Control:** Ensures coordinated operation of the radar's components, such as timing signal provision to various parts, enhancing the overall functionality and performance of the radar system.

Radar systems can be distinguished based on several key features, leading to a wide variety of types each tailored for specific applications [73].

1. **Pulse Radar:** Pulse radars emit a series of almost-rectangular pulses, representing the conventional radar form. They are engineered to execute a wide array of detection functions.
2. **High-resolution Radar:** These radars achieve high resolution in measuring range, angle, or Doppler velocity, often achieving range resolutions down to fractions of a meter or even centimeters for precise detection.
3. **Pulse Compression Radar:** Utilizing long pulses with internal modulation, pulse compression radars combine the energy benefits of long pulses with the resolution advantages of short pulses.
4. **Continuous Wave (CW) Radar:** CW radars continuously emit sine waves, primarily using Doppler frequency shifts to detect or measure the velocity of moving targets.
5. **FM-CW Radar:** A variant of CW radar, the FM-CW radar introduces frequency modulation to the waveform, facilitating range measurement.
6. **Surveillance Radar:** Surveillance radars are essential tools for detecting and pinpointing the positions of objects in terms of both distance and direction, and they are vital for ongoing monitoring and tracking of these objects.
7. **Moving Target Indication (MTI) and Pulse Doppler Radar:** MTI radars are specifically designed to identify moving objects amid environmental clutter, using lower pulse repetition frequencies to reduce ambiguities in distance measurement.

Pulse Doppler radars, available in high and medium PRF variants, utilize Doppler frequency shifts for extracting moving targets from clutter, each with its specific advantages and limitations regarding range and Doppler ambiguities.

8. **Tracking Radar:** These radars are specialized in calculating the paths of moving targets. They come in various forms, including Single Target Tracker (STT), Automatic Detection and Tracking (ADT), Track-While-Scan (TWS), and Phased Array Trackers, each tailored to provide specific tracking functions and data processing speeds.
9. **Imaging Radar:** Imaging radars, including Sidelooking Airborne Radar (SLAR), Synthetic Aperture Radar (SAR), and Inverse Synthetic Aperture Radar (ISAR), produce two-dimensional images of targets or scenes, leveraging movement and phase information for high-resolution imaging [77].
10. **Specialized Radar Types:** This category includes Weapon Control Radar, Guidance Radar, Weather Observation Radars (including Doppler Weather Radar for meteorological observations), and Target Recognition systems, each designed for specific functions from defense to weather analysis and target identification.
11. **Multifunction Radar:** Multifunction radars are capable of performing multiple radar functions, usually on a time-shared basis, demonstrating the versatility of radar technology across various applications.

Radar technology encompasses a broad spectrum of types, each optimized for specific applications ranging from target detection and tracking to imaging and weather observation. The classification of radars extends further into deployment platforms (land, sea, airborne, space-borne) and operational bands (UHF, L, S, etc.), highlighting the technology's adaptability and wide-ranging utility.

Radar can also be classified on the basis of frequency spectrum defined by IEEE radar bands and ITU nomenclature, which are shown in Table 2.1 [31]

International table				
Band designation	Nominal frequency range	Specific frequency ranges for radar based on ITU assignments (see Notes 1, 2)		
		Region 1	Region 2	Region 3
HF	3–30 MHz	(Note 3)		
VHF	30–300MHz	None	138 –144 MHz 216– 225 MHz (See Note 4)	223-230 MHz
UHF	300–1000 MHz (Note 5)	420–450 MHz (Note 4) 890–942 MHz (Note 6)		
L	1–2 GHz	1215–1400 MHz		
S	2–4 GHz	2300–2500 MHz		
		2700–3600 MHz	2700–3700 MHz	
C	4–8 GHz	4200– 4400 MHz (Note 7)		
		5250–5850 MHz	5250–5925 MHz	
X	8–12 GHz	8.5–10.68 GHz		
Ku	12–18 GHz	13.4–14 GHz		
		15.7–17.7 GHz		
K	18–27 GHz	24.05–24.25 GHz	24.05–24.25 GHz 24.65–24.75 GHz (Note 8)	24.05–24.25 GHz
Ka	27–40 GHz	33.4–36 GHz		
V	40–75 GHz	59–64 GHz		
W	75–110 GHz	76–81 GHz		
		92–100 GHz		
mm (Note 9)	110–300 GHz	126–142 GHz		
		144–149 GHz		
		231–235 GHz 238–248 GHz (Note 10)		

Table 2.1: Standard radar-frequency letter band nomenclature [31]

The details of the Notes in Table 2.1 is given below [31]:

1. The international frequency allocations from the ITU [32]. The ITU has not designated specific services for radar; frequencies mentioned are chosen from those services using radiolocation. These include both primary and secondary services and are indicative, subject to future amendments.
2. Details on radiolocation-specific frequency ranges [1].
3. Official ITU radiolocation bands for HF do not exist. HF radars may operate in a frequency range that begins just above the broadcast band, approximately 1.605 MHz, extending upwards to 40 MHz or more.
4. The frequency range of 216–450 MHz has sometimes been referred to as the P-band.
5. ITU’s official classification for ultra-high frequency bands extends up to 3000 MHz.

In the context of radar, the upper limit is typically 1000 MHz, with the L- and S-bands representing the higher frequencies within the UHF spectrum.

6. Certain frequencies are occasionally included in the L-band.
7. Reserved almost exclusively for aeronautical navigation, a particular band is allocated for airborne radar altimeters, barring rare exceptions.
8. The frequency spectrum between 24.65 and 24.75 GHz is designated for satellite radiolocation services, specifically from Earth to space.
9. The term 'mm' is derived from millimeter-wave radar technology, and it also refers to the V- and W-bands, as well as portions of the Ka-band, when broadly addressing the spectrum above 30 GHz.
10. There are no ITU allocations for frequencies higher than 275 GHz.

Radar has drawbacks as well. An object may be obscured behind another object, as the high-frequency radar pulse is only effective in a straight line of sight. Radars are also susceptible to multipath, which occurs when a signal is reflected by multiple objects before it returns to the sensor. This implies that a significant amount of noise that is a reflection of the moving object may still be present in the signal even after filtering out all stationary objects.

In the radar technology, especially from the High Frequency (HF) to the Ka bands, specific frequencies have been internationally designated for radar operations. Notably, beyond the X band, the impact of atmospheric attenuation on electromagnetic waves becomes markedly significant. As this can be seen in Figure 2.2, this attenuation is primarily due to the absorption of signals by atmospheric gases such as oxygen (O_2) and water vapor (H_2O) [67]. As a result, radars operating in these higher frequency bands tend to utilize frequencies known as "atmospheric windows," where attenuation levels are comparatively lower. For instance, most radars in the Ka band function near 35 GHz, while W band systems predominantly operate around 95 GHz, leveraging the relatively minimal atmospheric attenuation at these frequencies [68].

Lower frequency radars are generally favored for long-range surveillance owing to their lower atmospheric attenuation and the capability for high power outputs.

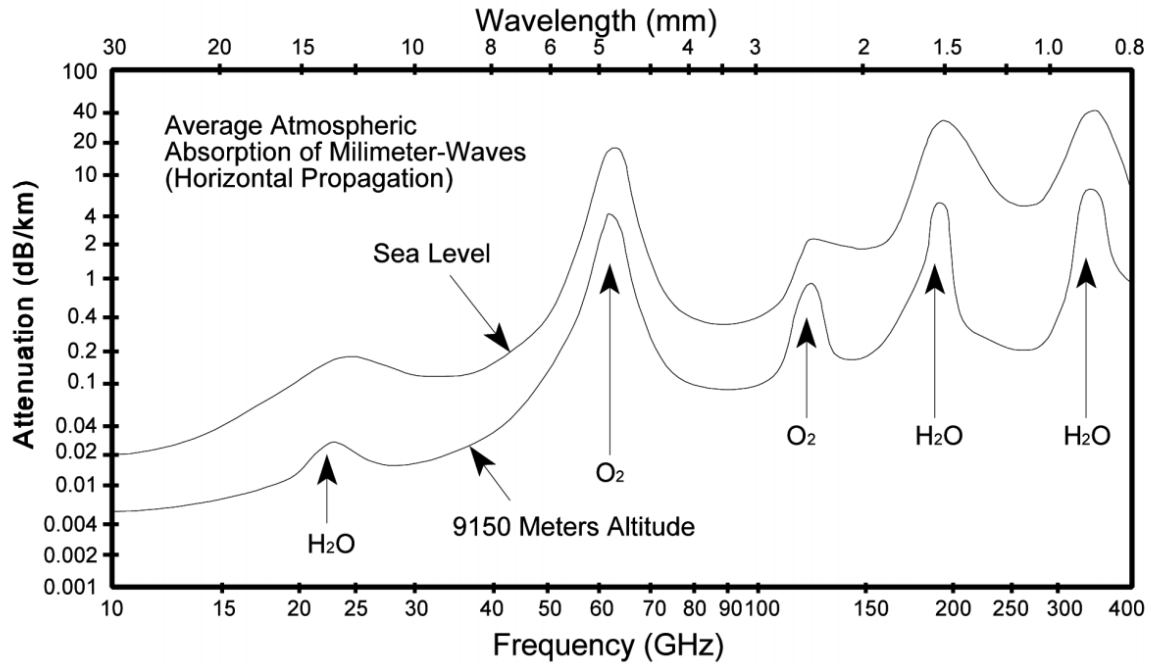


Figure 2.2: Attenuation of electromagnetic waves [21]

Conversely, higher frequencies are sought after for applications requiring fine resolution and shorter ranges, attributed to the potential for narrower antenna beamwidths for a given antenna size, notwithstanding the higher attenuation and reduced power outputs [21].

Weather conditions further influence radar signal propagation, with phenomena such as rain significantly affecting signal strength. While X band frequencies (around 10 GHz) and lower are mostly impacted by heavy rainfall, millimeter-wave frequencies encounter substantial losses even in conditions of light to moderate rain. The attenuation escalates significantly as frequencies increase, particularly within the millimeter-wave range (30 to 300 GHz), owing to the heightened absorption characteristics of atmospheric constituents like H₂O and O₂. Notably in Figure 2.3, frequencies around 60 GHz (oxygen) and 24 GHz and 184 GHz (water vapor) experience peak attenuation. However, several frequency windows exhibit notably lower atmospheric attenuation, which is where millimeter-wave frequency band applications predominantly operate [32].

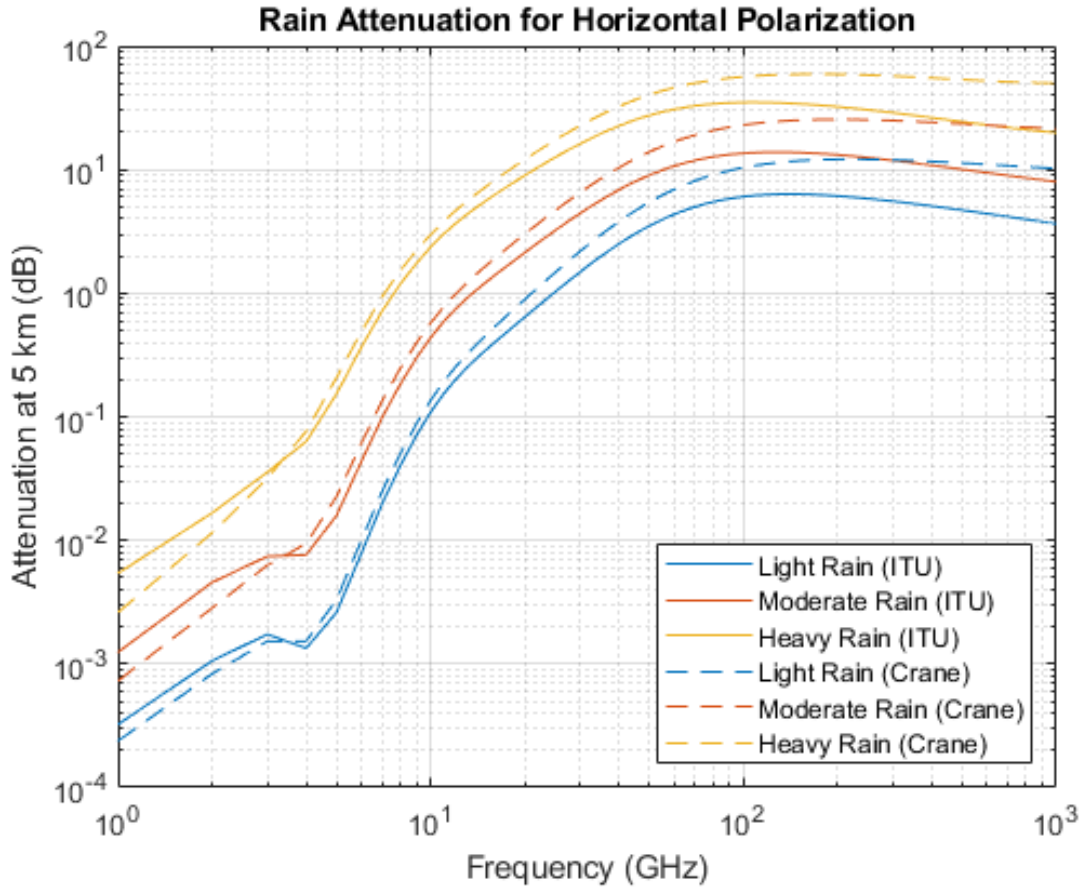


Figure 2.3: Rain Attenuation of Horizontal Polarization [48]

2.2 Radar Equations

The primary challenge in radar technology involves the detection of objects or phenomena by distinguishing between actual echoes from targets and mere noise in the receiver output [12]. Radar systems aim to identify an object's presence and determine its properties, such as range, location, height, and velocity, through the transmission and reception of electromagnetic waves. The core principle of radar operation relies on emitting electromagnetic energy, which disperses in all directions. A fraction of this energy, when reflected back by an object, is captured by the radar, providing vital data about the object [39].

These electromagnetic waves, specifically radio waves within the electromagnetic spectrum, are best understood through wave theory, offering a practical framework for analyzing the frequencies employed by radar systems for detection and measurement purposes.

When an object is identified, monitoring its location and speed may be required. Using a monostatic radar enables the determination of an object's position using a spherical coordinate system originating at the radar antenna's phase center [30, 67], as illustrated

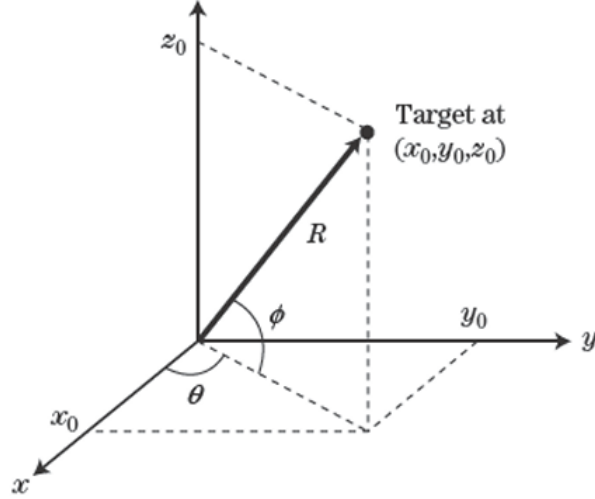


Figure 2.4: Spherical coordinate system for radar measurements [5].

in the diagram 2.4. In this coordinate framework, the direction of the antenna's focus, often referred to as the boresight direction, extends along the positive x-axis. The azimuth angle is designated by θ , and the elevation angle by ϕ . The range, R , the calculation of the distance to an object is based on the time interval between when a signal is sent and when it is received. Determining the elevation ϕ and azimuth θ angles is influenced by how the antenna is positioned, as the object needs to fall within the primary beam of the antenna for successful detection. Velocity estimation involves the measurement of the Doppler shift in the target echoes, which offers only the radial velocity component [69].

2.2.1 Radar Range Equation

Range of the target, or range, R , is the distance measured by radar from the target. We are aware that the target emits an echo signal to the radar at the speed of light, or C , in response to the signal sent by the radar.

The time it takes for a radar signal to travel to a target and return is denoted by " T ". Assuming the radar is separated from the target by a distance R , the total distance covered by the radar signal during its round trip is $2R$. The Speed formula is as follows [67].

$$\text{Distance} = \text{Speed} \times \text{Time} \quad (2.1)$$

$$2R = C \times T \quad (2.2)$$

$$R = \frac{C \times T}{2} \quad (2.3)$$

In pulsed radar systems, the waveform is typically characterized by three key parameters: the pulse duration (τ), the pulse repetition interval (PRI, denoted as T), and the pulse repetition frequency (PRF, denoted as f_p). The PRI represents the temporal gap between consecutive pulses. Conversely, the PRF indicates the frequency at which pulses recur every second and is mathematically defined as the reciprocal of the PRI [45]:

$$f_p = \frac{1}{T} \quad (2.4)$$

In the scenario where the time delay falls below the pulse repetition interval (PRI), the distance calculated using equation (2.3) remains unambiguous. Conversely, should the pulse repetition frequency (PRF) be excessively high or the PRI notably brief, it becomes possible for echo pulses from a target to return subsequent to the emission of the following pulse, thereby introducing ambiguity in the calculated range. The formula for determining the maximum range without ambiguity is expressed as follows [34]:

$$R_{\text{un}} = \frac{C}{2f_p} \quad (2.5)$$

where R_{un} denotes the maximum unambiguous range, C the speed of light, and f_p the pulse repetition frequency.

In the context of radar systems, the radar equation establishes a connection between a radar's detection range and the attributes of the transmitting and receiving components, the antenna, the target, and their mutual separation. Focusing on a monostatic radar configuration, assume that P_t represents the power transmitted, G symbolizes the gain of the antenna, and R signifies the distance to the target. Under these conditions, the power density reaching the target can be described as follows [45]:

$$P_d = \frac{P_t G}{4\pi R^2} \quad (2.6)$$

Assuming that the target intercepts the incoming power across an area denoted by σ , known as the radar cross-section (RCS), and disperses it uniformly in all directions, the power density observable at the radar can be articulated by:

$$P'_d = \frac{P_d \sigma}{4\pi R^2} = \frac{P_t G_t \sigma}{(4\pi R^2)^2} \quad (2.7)$$

It is posited that the radar's antenna possesses an effective area A_e , which correlates with the antenna's gain G_r via the relation:

$$A_e = \frac{G_r \lambda^2}{4\pi} \quad (2.8)$$

Consequently, the power that the radar receives is delineated as follows:

$$P_r = P_d A_e = \frac{P_t \cdot G_t \sigma G_r \lambda^2}{(4\pi R^2)^2 \cdot 4\pi} = \frac{P_t G^2 \sigma \lambda^2}{(4\pi)^3 R^4} \quad (2.9)$$

In this scenario, the identical antenna is used for both sending and receiving signals, where a power divider is utilized to differentiate between downlink and uplink transmissions. This setup is common among multiband RF antennas.

Let S_{\min} represent the minimum signal that can be detected, and let R_{\max} signify the furthest distance at which a target can be identified given a specific RCS σ , which can be represented as [68]

$$R_{\max} = \left[\frac{P_t G^2 \sigma \lambda^2}{(4\pi)^3 S_{\min}} \right]^{1/4} \quad (2.10)$$

Equation (2.10) delineates the received power for an ideal radar system operating in free space, excluding any signal processing enhancements for improved sensitivity. To accommodate real-world considerations, additional factors for loss and gain are typically included in the equation [40]. Losses from various components, such as duplexers, power dividers, waveguides, and the radome – a protective shield enveloping the antenna – as well as propagation anomalies not present in free space, can be aggregated into a system loss factor L_s , which attenuates the received power [22]. System losses are generally within the range of 3 to 10 dB, though they can significantly vary. A key element contributing to loss at X band and frequencies beyond is the atmospheric attenuation, denoted $L_a(R)$. This form of loss is range-dependent. If the one-way atmospheric loss per kilometer is given as α in Fig. 2.2, then the atmospheric loss over a range R in meters, not kilometers, can be represented [68]:

$$L_a(R)(\text{dB}) = 2\alpha(R/1000) = \alpha R/500 \text{ dB} \quad (2.11)$$

In linear units, the loss is therefore:

$$L_a(R) = 10^{\alpha R/5000} \quad (2.12)$$

The attenuation due to the atmosphere may be minimal at 10GHz for intermediate distances, but it can increase to significant levels—measured in tens of decibels—at 60GHz for spans of a few kilometers. This illustration further verifies that atmospheric attenuation, similar to systemic losses, is heavily dependent on the radar’s operational frequency [8]. When considering both atmospheric and system losses, Equation (2.10) can be amended to yield the received power P_r :

$$P_r = \frac{P_t G^2 \lambda^2 \sigma}{(4\pi)^3 R^4 L_s L_a(R)} \quad (2.13)$$

where, P_r is the instantaneous power received, P_t the power transmitted, G the antenna gain, λ the wavelength, σ the radar cross-section, R the range, L_s the system losses, and $L_a(R)$ the atmospheric losses as a function of range.

Equation (2.13) extracts the radar range equation, establishing a relationship between the received echo power and crucial radar and target characteristics such as transmitted power, frequency, antenna gain, the radar cross-section (RCS), and target distance [6]. As signal power corresponds to the squared amplitude of the electric field, the equation provides a framework for understanding the amplitude dynamics of both the target signal and clutter [56].

2.3 Millimetre Wave Radars

Millimeter-wave (mmWave) radar refers to radar systems that operate at electromagnetic spectrum frequencies between 30 GHz and 300 GHz. This range falls within the millimeter wave band, as the wavelengths vary from 1 to 10 millimeters. mmWave radar systems are distinguished by their ability to use these higher frequency waves to achieve higher resolution and accuracy compared to lower frequency radar systems [81].

The core mechanism of mmWave radar involves the emission of electromagnetic waves followed by the detection of the reflections from various objects. The characteristics of the reflected signals, such as their time of arrival, frequency shift, and the angle of arrival, provide information about the object’s distance, velocity, and shape.

Many of the transmitted radar signal’s properties can be adjusted using software because of the sensor’s architecture. One must choose between the measurements’ resolution and range while choosing these parameters.

mmWave radar technology presents multiple benefits compared to conventional radar systems:

- **High Resolution:** Due to their higher frequencies, mmWave radars can detect objects with much finer resolution, allowing for the identification and differentiation of closely spaced objects [27].
- **Compact Size:** Higher frequencies enable smaller antennas while maintaining high directional accuracy, which is beneficial for applications requiring compact sensor sizes [75].
- **Penetration Capabilities:** mmWave signals can penetrate certain materials, such as clothes, which can be useful in security and biomedical applications [3].
- **Less Susceptible to Interference:** Higher frequency waves have less interaction with small particles (like rain or dust), reducing the radar's clutter [51].
- **Multi-Target Capability:** mmWave radar can effectively handle multiple targets in close proximity due to its high spatial resolution [65].
- **Reduced Vulnerability:** Electronic countermeasures known as radar jamming and deception send out radio frequency signals with the intention of producing noise or misleading information in order to interfere with the radar's functionality by overloading its receiver. Small beamwidth radar antennas reduce the likelihood that a jammer may damage the signal.

Due to the advantages of mmWave radars, they have found application in various sectors:

- **Automotive:** Advanced driver-assistance systems (ADAS) employ infrared sensors to identify vehicles, pedestrians, and various obstacles, contributing to accident prevention and enhanced road safety. [62].
- **Security:** For through-clothes and through-walls imaging applications in security screenings at airports or public events [18].
- **Health-care:** In medical imaging and vital sign monitoring, such as contactless breathing and heartbeat detection [44].
- **Telecommunications:** Supporting high-speed data transmission and mobile communications in 5G and beyond networks [65].
- **Industrial Automation:** For precise location tracking and collision avoidance in automated industrial environments [14].

Despite the advantages, mmWave radars face several challenges:

- **Atmospheric Absorption:** Higher frequency waves experience significant attenuation due to atmospheric gases, impacting the effective range of the radar under certain environmental conditions [5].
- **Line of Sight Requirement:** Millimeter wave (mmWave) radar systems necessitate an unobstructed view of the target since any physical barriers can diminish their operational efficacy [27].
- **Interference:** As more devices utilize the mmWave spectrum, there is an increased risk of interference, which can degrade radar performance [29].
- **Cost and Complexity:** The production of mmWave radar systems involves advanced and expensive technology, making it more costly compared to lower frequency radar systems [65].

The advancements brought by mmWave radar technology are notable, including enhanced resolution, a smaller form factor, and adaptability for various uses. However, the challenges related to atmospheric effects, line of sight dependency, and cost need to be mitigated for broader adoption.

2.4 Radar Data Acquisition

Radar technology captures the reflectivity's spatial distribution using a three-dimensional spherical coordinate system consisting of elevation, azimuth, and range. This is accomplished by pulsed radars that emit a series of discrete pulses and then record the returning voltage over time, which correlates directly with range.

Modern pulsed radars employ coherent receivers that facilitate the capture of complex-valued voltage measurements. These systems also digitally record and process the incoming data. Selecting appropriate sample rates and quantization methods is crucial in the design of digital data collection systems. The selected configurations can significantly impact the processor's demand for computational power and memory, and they also influence the levels of noise, aliasing, and signal fidelity [67].

The radar signal, comprised of a finite RF pulse train, can be characterized using four longer time scales: the RF wave period, the pulse width, the pulse repetition interval (PRI), and the coherent processing interval (CPI) [68]. Correspondingly, the spectrum of this pulse train can be analyzed across four related frequency scales: the radar RF, the pulse BW, the pulse repetition frequency (PRF), and the spectral line bandwidth.

A core principle of Fourier analysis is the inverse relationship between time and frequency scales: a function that is longer or more spread out in time will have a shorter or more compact frequency spectrum, and vice versa. This inverse relationship and its implications are essential for understanding Doppler processing techniques [68].

Spectrum of a Continuous Wave Signal

The most basic radar waveform is the real-valued, continuous wave (CW) sinusoidal signal with an infinite duration and frequency f_0 Hz. This signal is characterized by a single time scale, its period $T_0 = 1/f_0$. Its spectrum is comprised of two impulse functions located at f_0 and $-f_0$. This spectral form can be derived using Euler's relation:

$$x(t) = A \cos(2\pi f_0 t) = \frac{A}{2} \{e^{j2\pi f_0 t} + e^{-j2\pi f_0 t}\} \quad (2.14)$$

Applying the Fourier transform to $x(t)$, based on equation (2.14), yields:

$$X(f) = \frac{A}{2} [\delta_D(f - f_0) + \delta_D(f + f_0)] \quad (2.15)$$

These results are visualized in Figure 2.5, which shows plots of the time function and its corresponding amplitude in the frequency domain.

Euler's formula also allows for the representation of a sine function as:

$$x(t) = \sin(2\pi f_0 t) = \frac{1}{2j} \{e^{j2\pi f_0 t} - e^{-j2\pi f_0 t}\} \quad (2.16)$$

and its Fourier transform is given by:

$$X(f) = \frac{A}{2j} [\delta_D(f - f_0) - \delta_D(f + f_0)] \quad (2.17)$$

These equations describe the Fourier transform pairs for the cosine and sine functions, respectively, showing their transformation from time domain to frequency domain.

Echo Reception and the Doppler Effect

When a radar emits a wave and it encounters an object, the wave bounces back towards the radar with a certain delay τ and, if the object is in motion, a change in frequency Δf due to the Doppler effect. The received signal $r(t)$ can then be expressed as [74]:

$$r(t) = A \cos(2\pi(f_0 + \Delta f)(t - \tau) + \phi) \quad (2.18)$$

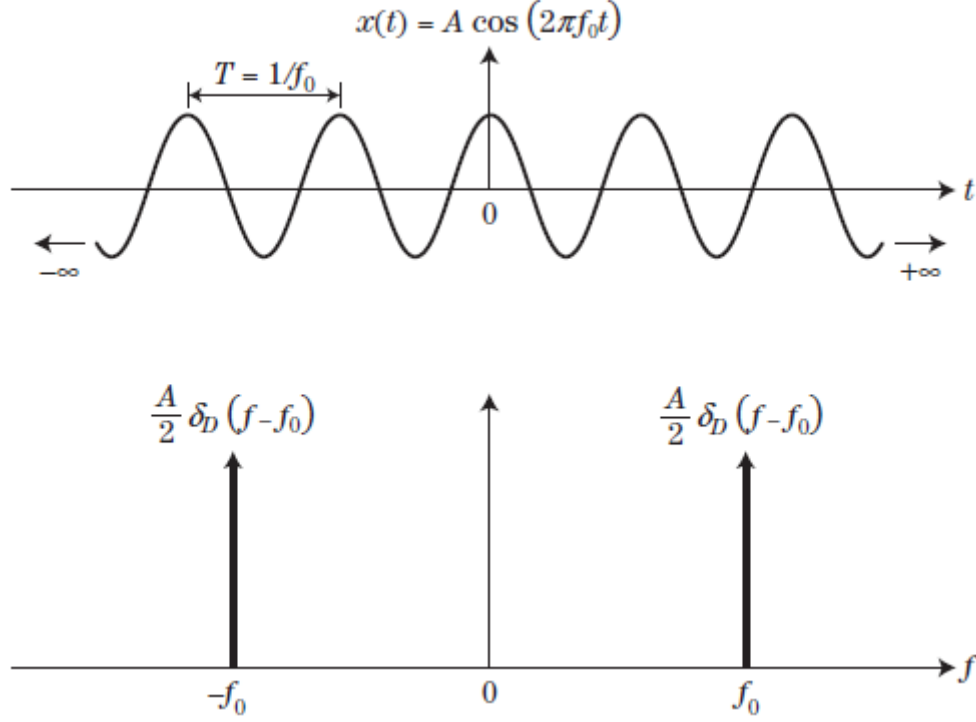


Figure 2.5: Infinite-length continuous wave (CW) signal of frequency f_0 and its frequency spectrum [68].

Signal Processing: Range and Velocity Determination

A key operation in radar technology is the determination of an object's distance and speed by analyzing the differences between the wave sent out and the wave that returns. In FMCW radar systems, a method known as dechirping is employed. This process involves mixing the outgoing signal with the incoming signal to generate a beat frequency, which helps in calculating the precise range and velocity of the object detected [74]:

$$f_{\text{beat}} = \frac{2\Delta R B}{c T} \quad (2.19)$$

where ΔR is the range difference, c is the speed of light, B is the bandwidth of the chirp, and T is the chirp duration.

Velocity is determined by analyzing the Doppler frequency shift, which is given by:

$$f_{\text{Doppler}} = \frac{2v}{\lambda} \quad (2.20)$$

where v is the velocity of the object, and λ is the wavelength of the radar signal.

These formulas represent the basic mathematical underpinnings for mmWave radar data acquisition and processing. The accurate detection and interpretation of f_{beat} and f_{Doppler} enable the determination of both range and velocity of targets, which are critical for applications like automotive radars and air traffic control.

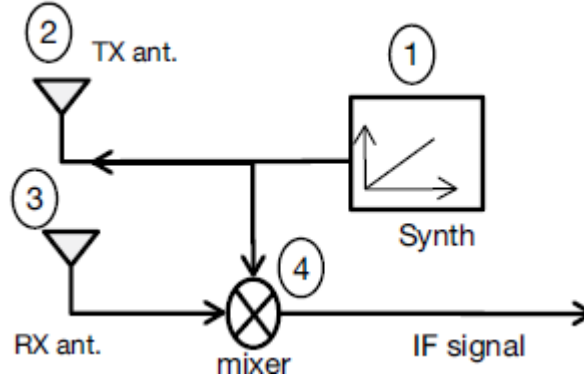


Figure 2.6: FMCW radar block diagram [12].

This phase discrepancy allows for the execution of an FFT across the antenna dimension, similar to the process utilized for measuring velocity. By incorporating multiple transmitter antennas, the radar system effectively increases the number of antennas. Each transmitter/receiver combination creates a virtual antenna, which enhances the resolution. The transmitters must alternate their transmission times, as illustrated in Figure 2.6.

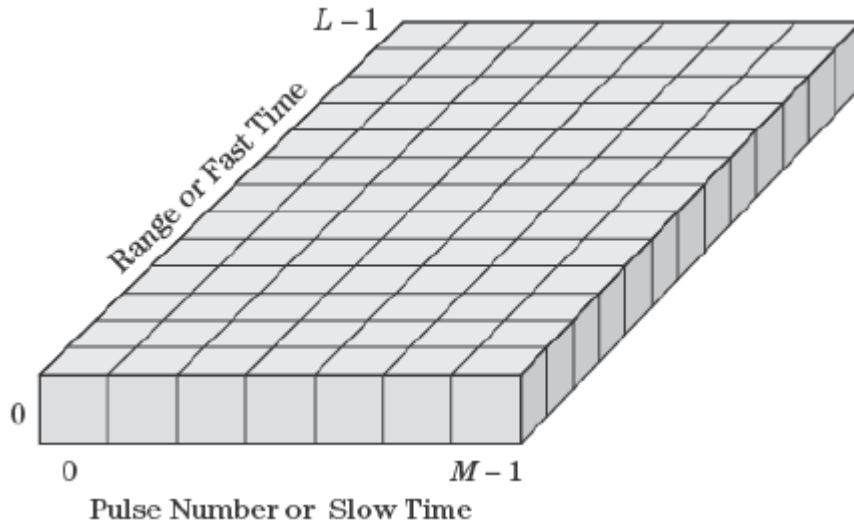


Figure 2.7: FMCW radar fast-time/slow-time coherent processing interval (CPI) data [68].

Antennas in certain systems can produce multiple simultaneous outputs, a prime example being those employing a phased array antenna with several subarrays.

Each subarray may have its own dedicated receiver, or in some cases, each array element

might connect to an individual receiver. Each receiver outputs a data matrix similar to that depicted in Fig. 2.7 for every pulse burst.

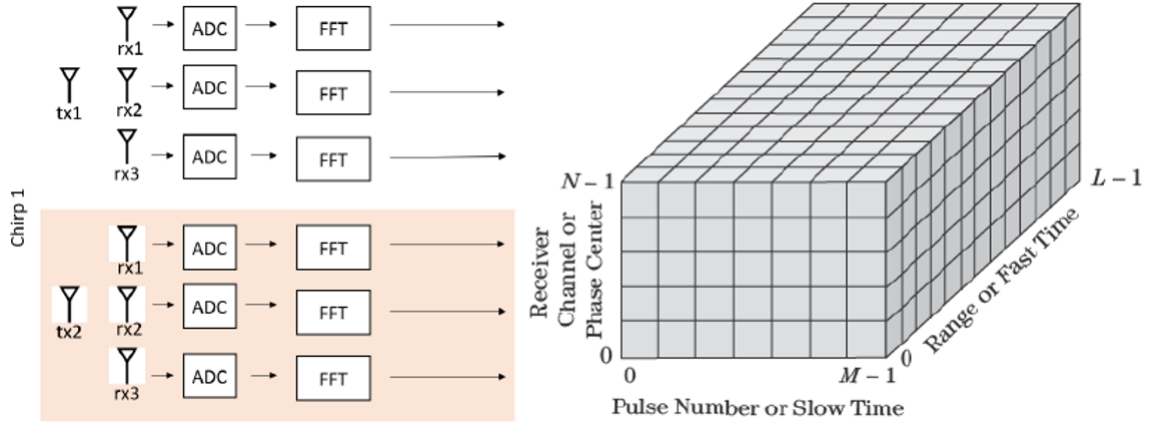


Figure 2.8: Recording a chirp involves a system equipped with TX and RX antennas. The TX (Transmitter) antennas emit chirps sequentially, while each RX (Receiver) antenna captures the chirp simultaneously. Subsequently, the Digital Signal Processor (DSP) executes a range FFT, and the outcomes are preserved in the radar cube before the subsequent chirp. [7, 68].

The aggregate data $y[l, m, n]$ from all N channels is termed a datacube, as illustrated in Fig. 2.8. The data structure utilizes an additional layer, sometimes called the "receiver channel dimension" or the "phase center dimension" [67].

Additionally, systems utilizing a monopulse antenna, which are prevalent in certain tracking radar applications, also generate datacubes. A monopulse antenna typically has three output channels, resulting in a datacube with $N = 3$ layers. In radar processing systems, it is customary to organize radar data explicitly in the memory as a datacube, which is a three-dimensional structure composed of complex-valued data.

2.5 Radar Data Representation

Radar systems are sophisticated sensors that collect data which is then transformed into forms that can be easily analyzed and interpreted. This section explores the fundamental concepts of radar data representation, focusing on the structure and format of the data as it is typically processed and utilized in radar systems.

At the core of radar data representation is the concept of the data matrix, which is an essential data structure in radar signal processing. A data matrix organizes the data collected by radar systems into a two-dimensional array.

2.5.1 Raw Frequency Domain Data

In utilizing millimeter-wave (mmWave) radar for skin cancer classification, raw frequency domain data serves as a fundamental component for distinguishing between malignant and benign lesions. This data representation captures the frequency response of emitted mmWave signals after they interact with the skin. As mmWave radar emits signals that penetrate the skin, different layers and structures within the skin reflect these signals based on their dielectric properties. These reflections are then received and recorded in the frequency domain.

The raw frequency domain data essentially reflects the skin's impedance profile, which varies significantly between cancerous and healthy tissues due to differences in water content, cellular density, and structural composition. By analyzing this data, distinct patterns that correspond to either benign or malignant skin conditions can be identified. Such patterns are crucial for training classification algorithms, as they provide the high-dimensional features needed for accurate discrimination. This form of data representation is highly advantageous as it enables the capture of intricate tissue responses at various frequencies, enhancing the sensitivity and specificity of the classification process.

2.5.2 Raw Time Domain Data

To analyze and manipulate this data in the frequency domain, we use the method of an inverse Fast Fourier Transform (IFFT) algorithm [59]. The inverse Fast Fourier Transform (inverse FFT) is utilized to convert data from the frequency domain back to the time domain. Let $X[k]$ represent the frequency domain samples and $x[n]$ the corresponding time domain signal. The inverse FFT is mathematically defined as follows [59]:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j \frac{2\pi kn}{N}} \quad (2.21)$$

where:

- N represents the complete set of samples,
- n ranges from 0 to $N - 1$ and represents the current sample in the time domain,
- k is the current frequency index,
- j is the imaginary unit, and
- e is the base of the natural logarithm.

The equation 2.21 computes each time domain point $x[n]$ as the sum of all frequency

domain components $X[k]$, each value is multiplied by a complex exponential that oscillates at a rate determined by the multiplication of the time index n with the frequency index k . The coefficient $\frac{1}{N}$ serves as a normalization factor to maintain the original energy level of the signal after applying the inverse Fast Fourier Transform (IFFT).

This operation effectively reconstructs the original time-domain signal from its frequency components, providing a detailed temporal representation crucial for analyzing the dynamics and movement within radar systems.

This transformation converts the complex frequency domain data into raw time domain data. The time domain representation is instrumental for understanding the temporal characteristics of the radar signals, offering insights into the dynamics and movement of targets, as well as enhancing object detection and ranging accuracy.

2.5.3 Spectrogram Generation

A spectrogram, or Short Time Fourier Transform (STFT), performs Fourier spectral analysis within a confined time window. This method segments the input signal into successive time windows and computes the Fourier transform for each segment. The spectrogram is commonly used to represent signals across both time and frequency dimensions. The formula for the spectrogram is represented below, where $w(t - \tau)$ denotes the finite time window:

$$X_{ST}(t, \omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} x(t)w(t - \tau)e^{-i\omega t} dt \quad (2.22)$$

where:

- $x(t)$ is the signal being analyzed, a function of time.
- $w(t - \tau)$, known as the window function, is used to temporally localize the signal around the point τ . This function is essential in the STFT to analyze the signal in smaller segments over time.

Sometimes, analyzing a signal exclusively in the time or frequency domain does not adequately capture the dynamics of what's occurring within the signal. Spectrograms provide a valuable tool for illustrating how a signal's spectral content evolves over time. They treat the data as being piece-wise stationary and aim to localize specific time segments by employing fixed-length windows. The chosen window size must be sufficient to encapsulate at least one complete cycle of the frequency component of interest. However, it's important not to select an overly large window size as this can degrade the temporal

resolution.

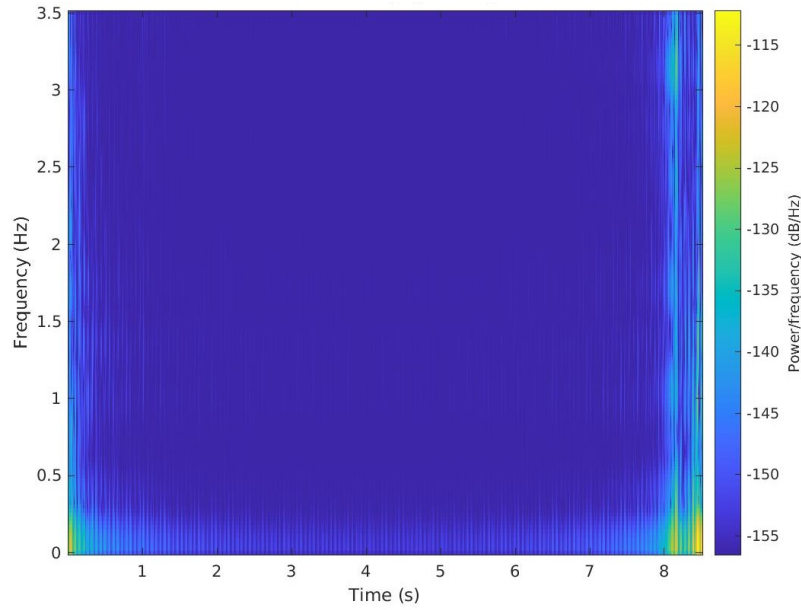


Figure 2.9: Spectrogram plot frequency against time with power distribution.

Building on the transformed time domain data, a spectrogram for each point in the data matrix is generated. As in Fig. 2.9, this spectrogram plots frequency against time, where the color intensity represents the power distribution of the signal at each point. The use of spectrograms is pivotal in visualizing the frequency spectrum over time, allowing researchers to observe how the spectral characteristics of objects evolve. This is particularly useful for identifying and distinguishing between different materials and objects based on how they affect the frequency spectrum of the emitted radar signals.

2.5.4 Comparison of Data Representation Techniques in mm-Wave Radar

Raw Frequency Domain Data vs. Transformation to Time Domain Data

- **Nature of Data:**

1. *Raw Frequency Domain Data:* Captures data directly from radar sensors, reflecting the electromagnetic signal characteristics across multiple frequency points. Essential for high-resolution target structure analysis and material properties differentiation.
2. *Time Domain Data:* Derived from the inverse FFT of frequency domain data, emphasizing temporal behaviors of signals and useful for dynamic tracking and velocity estimation.

- **Purpose and Utility:**

1. *Raw Frequency Domain Data*: Crucial for applications requiring detailed object profiling and material differentiation.
2. *Time Domain Data*: Valuable for analyzing object movements and improving spatial resolution in detection tasks.

- **Complexity and Processing:**

1. *Raw Frequency Domain Data*: Requires less computational processing initially but is complex due to high data volume.
2. *Time Domain Data*: Involves additional computational steps (FFT) and is intuitive for analyzing temporal changes.

Raw Time Domain Data vs. Spectrogram Generation

- **Form of Data Representation:**

1. *Time Domain Data*: Provides a one-dimensional view of signal variations over time, essential for proximity and motion analysis.
2. *Spectrogram*: A two-dimensional representation, plotting frequency against time and colored by signal power, offering a comprehensive view of signal dynamics.

- **Analytical Depth:**

1. *Time Domain Data*: Simplifies the radar signal to focus on temporal aspects, suitable for straightforward distance calculations.
2. *Spectrogram*: Combines time and frequency analysis, allowing for advanced studies on material properties and motion patterns.

- **Applications:**

1. *Time Domain Data*: Crucial for applications such as collision detection, speed monitoring, and range finding.
2. *Spectrogram*: Ideal for complex scenarios requiring material identification or where environmental factors significantly alter signals.

The progression from raw frequency domain data through time domain transformation to spectrogram generation enhances the level of detail and utility in mmWave radar data analysis.

2.6 Artificial Intelligence

The term "artificial intelligence" (AI) in computer science encompasses a broad spectrum. It refers to the application of mathematics and technology to mimic human decision-making processes. While sometimes as simple as an "IF this, THEN that" logic, it broadly includes all machine learning and deep learning techniques.

Alan Turing famously shifted the discussion around the capability of computers from whether they can think to "Can machines display intelligent behavior?" in 1950 [79]. Alan Turing developed the Turing Test as a way to evaluate whether a computer can conduct conversations that are indistinguishable from those with humans, focusing on external behavior rather than what might be occurring inside the mind.

In line with Turing, Russell and Norvig also advocated for defining AI by actions rather than thoughts, cautioning against equating computers directly with humans, much like distinctions in aeronautical engineering [71]. Marvin Minsky defined intelligence as "the ability to solve hard problems" [53], and John McCarthy described it as "the computational part of the ability to achieve goals in the world" [50]. Both perspectives highlight the practical aspects of intelligence, focusing on problem-solving and steering clear of deeper philosophical entanglements.

Google defines intelligence as the ability of systems to integrate knowledge in a manner akin to biological intelligence.

2.6.1 Introduction to Machine Learning Models

Building on the foundation laid by the pioneers of artificial intelligence, this thesis incorporates modern machine learning techniques to address complex problems. As discussed previously, AI encompasses a broad spectrum of methodologies from simple logical constructs to advanced machine learning and deep learning models.

This section details the specific machine learning models employed to train data, reflecting on their relevance and efficacy in achieving the objectives of this research.

Description of Machine Learning Models Used:

2.6.1.1 Logistic Regression Model

Logistic regression is a pivotal machine learning method for binary classification tasks. Unlike linear regression, which predicts continuous outcomes, logistic regression estimates the probability that an instance belongs to a particular category, typically represented as 0 or 1. This model is widely adopted across various fields such as banking, healthcare, and natural language processing, owing to its effectiveness and straightforwardness [35].

Logistic regression uses a special function, called the sigmoid function, to convert any number into a chance (between 0 and 1). The logistic function is defined as:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (2.23)$$

where z is a linear combination of the input features weighted by corresponding coefficients, plus a bias term. This can be mathematically expressed as:

$$z = b_0 + b_1x_1 + b_2x_2 + \cdots + b_nx_n \quad (2.24)$$

Here, b_0 represents the bias term, and b_1, b_2, \dots, b_n are the coefficients for the input features x_1, x_2, \dots, x_n .

The logistic function acts as a gatekeeper, ensuring the outcome of the linear combination stays within the 0 to 1 range, which is crucial for interpreting the results as probabilities. An output probability of 0.5 or higher categorizes the instance as belonging to class 1, while a probability less than 0.5 classifies it as class 0.

To refine the model parameters (coefficients and bias), techniques such as Maximum Likelihood Estimation (MLE) or gradient descent are employed, enhancing the alignment of predicted probabilities with actual class labels.

2.6.1.2 Ridge Classifier

The Ridge Classifier is an adaptation of Ridge regression tailored for binary classification tasks. Unlike logistic regression which outputs probabilities, the Ridge Classifier assigns direct class labels to instances. This model proves to be highly effective when the variable of interest is of a categorical type.

Ridge regression, which forms the basis for the Ridge Classifier, tackles the problem of multicollinearity among predictors. This issue can cause the coefficients in a regression

model to be inconsistent and unreliable. The regularization term added by the Ridge method helps stabilize these estimates by penalizing large coefficients. The model is mathematically expressed as [28]:

$$\hat{\beta} = \arg \min_{\beta} \left\{ \sum_{i=1}^n (y_i - X_i \beta)^2 + \lambda \|\beta\|^2 \right\} \quad (2.25)$$

where:

- y_i represents the dependent variable.
- X_i denotes the independent variables.
- β are the coefficients.
- λ is the regularization parameter, controlling the size of the penalty $\lambda \|\beta\|^2$, where $\|\beta\|^2$ is the square of the $L2$ norm of the coefficients.

In classification scenarios, the Ridge Classifier utilizes the sign of the predicted value from the regression model to assign class labels. The targets are typically encoded as $\{-1, 1\}$. The class assignment is determined by:

$$\text{Class} = \text{sign}(X \hat{\beta}) \quad (2.26)$$

- **Choosing λ :** The regularization strength λ is typically selected through cross-validation to ensure the model is neither overfitting nor underfitting.
- **Scaling of Features:** It is essential to scale the input features when using the Ridge Classifier since the magnitude of the features affects the penalty term.

The Ridge Classifier is favored in fields where robustness in predictions is crucial and features exhibit high multicollinearity. It finds applications in finance, genetic data analysis in biology, and other domains where stability in prediction is key.

2.6.1.3 K-Nearest Neighbors Classifier

The K-Nearest Neighbors (KNN) classifier is a fundamental tool in machine learning, known for its straightforwardness and effectiveness in categorizing data. As a non-parametric, instance-based learning algorithm, KNN determines the classification of new instances based on the most frequent labels of the nearest samples in the feature space [17].

The operational principle of KNN is straightforward: it assumes that similar things exist in close proximity. It operates on the principle that objects close together in the feature

space tend to be similar. Here's how KNN classifies a new data point:

1. **Choose the number K of neighbors:** Determine the number K , which is the count of nearest training data points to consider.
2. **Calculate the distance:** Calculate the distance between a new instance and each instance in the training dataset using an appropriate distance measure.
3. **Identify K nearest neighbors:** Locate the K closest instances to the new data point.
4. **Vote for labels:** Determine the class of the new instance by selecting the class label that appears most frequently among its closest neighbors.

The classification rule for a new instance x can be mathematically represented as:

$$\text{Class}(x) = \text{mode}\{y_{i1}, y_{i2}, \dots, y_{iK}\} \quad (2.27)$$

where y_{ij} represents the class labels of the K nearest neighbors to x .

Practical Considerations

- **Choosing K :** Choosing the right number of neighbors, K , in the K-Nearest Neighbors (KNN) algorithm is crucial; opting for a smaller K can lead the model to be overly influenced by outliers, whereas a higher K may overly generalize the decision boundary.
- **Distance Metric:** The effectiveness of the KNN algorithm largely hinges on the choice of distance metric, which is used to determine the closeness of various data points.
- **Feature Scaling:** Proper scaling of features is vital since KNN relies on the distances between data points.

KNN is extensively employed in systems where the decision boundary is not initially clear, such as recommendation systems and anomaly detection, due to its robustness and straightforward logic.

2.6.1.4 Support vector classifier (SVC)

The Support Vector Classifier (SVC) stands out as a robust method under the umbrella of supervised learning, applicable to both classification and regression tasks. Its prowess is particularly noted in classification, where it efficiently deals with both binary and

multiclass problems. SVC operates by identifying the most favorable hyperplane within a multi-dimensional space to distinctly segregate various classes.

The Support Vector Machine (SVM) described by Noble (2006) [57] acts primarily as a binary classifier, creating an ideal separation between two distinct groups of data. In a two-dimensional space, this separator is a line, but in more complex, higher-dimensional spaces, a clearly defined decision plane is crucial for effective segmentation. Central to the methodology of Support Vector Machines (SVM) is the goal to maximize the distance, or margin, between the closest points of the data set and the dividing hyperplane. This enhancement of the margin ensures a clear and precise separation of the classes [60].

The classification function is expressed as:

$$f(x) = w^T x + b \quad (2.28)$$

SVM utilizes the concept of a support vector, denoted by the margin r , defined as the geometric distance between the closest data points on either side of the hyperplane. This distance, r , is computed as twice the margin, $2r$, using the following equation:

$$r = \hat{r} = \frac{y \cdot f(x)}{\|w\|} \quad (2.29)$$

Here, \hat{r} denotes the margin within the functional interval, illustrating SVM's fundamental principle of maximizing distances between classes for effective separation.

2.6.1.5 Decision Tree Classifier

A Decision Tree Classifier is a non-parametric supervised learning method used for classification tasks. The decision tree model employs a structured division of data into subsets using specific conditions tied to feature values, forming a tree-shaped model. This approach is widely appreciated for its simplicity and effectiveness across various fields.

Essentially, a decision tree continually breaks down a dataset into smaller segments through recursive partitioning until each segment achieves maximum purity. Each node in the tree determines the best split based on a criterion that evaluates the impurity or the informational value gained from that particular division. Mathematically, the purity of a node is often measured using Gini impurity or entropy. the Gini impurity, which for any given subset is calculated as follows:

$$I_G(p) = 1 - \sum_{i=1}^k p_i^2 \quad (2.30)$$

where p_i is the probability of an object being classified to a particular class.

Entropy, another common measure, is defined as:

$$H(T) = - \sum_{i=1}^k p_i \log_2(p_i) \quad (2.31)$$

where $H(T)$ is the entropy of the subset T , and p_i is the proportion of the class i within the subset.

In decision tree algorithms, the optimal partitions are those that lead to the greatest reduction in uncertainty or increase in purity, evaluated by comparing the impurity levels before and after a given split.

Decision Tree Classifiers are extensively used in the fields of data mining, customer relationship management, and medicine for diagnostic purposes. These algorithms are favored for their straightforward interpretability and the clear, graphical depiction they provide of decision-making pathways.

2.6.1.6 Random Forest Classifier

The widely utilized machine learning technique, Random Forest, addresses both classification and regression challenges. As part of the ensemble learning category, it enhances prediction accuracy and robustness by aggregating outcomes from numerous individual models, primarily decision trees [9].

A Decision Tree can be visualized like a flowchart diagram where each internal node denotes a specific characteristic, each branch represents a rule for decision-making, and each terminal node, or leaf, corresponds to a particular result.

For a classification task involving a dataset with N samples and M features, a random subset of the data, denoted as $N_{\text{subsample}}$, is utilized to construct each tree within the Random Forest. During the training phase, the algorithm randomly selects m features from M , determining the optimal split for each node.

If K decision trees are constructed, the Random Forest's prediction for a new input x is given by:

$$\hat{y}_{RF}(x) = \frac{1}{K} \sum_{k=1}^K \hat{y}_k(x) \quad (2.32)$$

where:

- $\hat{y}_{RF}(x)$ represents the Random Forest prediction for input x .
- $\hat{y}_k(x)$ denotes the prediction of the k -th decision tree.
- K is the total number of decision trees in the Random Forest.

By integrating multiple decision trees and other machine learning methodologies, Random Forests mitigate overfitting and discern complex patterns in data, thereby improving both the accuracy and reliability of predictions.

2.6.1.7 Gradient Boosting Classifier

The Gradient Boosting Classifier is a sophisticated method that belongs to the ensemble family of learning models. It enhances the capabilities of simple models by aggregating several weak predictors, which are often decision trees, to build a comprehensive and powerful predictive model. The `GradientBoostingClassifier` with a parameter `n_estimators` specifies the creation of n sequential trees, each one correcting errors made by its predecessors.

The underlying principle of Gradient Boosting involves sequentially adding predictors to an ensemble, each one correcting its predecessor. The method specifically minimizes a loss function, an approach that can be viewed as a gradient descent in function space [24]. Mathematically, each addition to the ensemble is formulated as follows:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \quad (2.33)$$

where:

- $F_m(x)$ is the boosted model after the m -th iteration.
- $F_{m-1}(x)$ is the model from the previous iteration.
- $h_m(x)$ is the weak learner (e.g., a decision tree).
- γ_m represents the learning rate or step size, influencing the degree to which $h_m(x)$ affects the overall model.

The training process aims to find a basic model (weak learner) $h_m(x)$ that minimizes a metric indicating how wrong the predictions are (loss function) L , given by:

$$\gamma_m = \arg \min_{\gamma} L(F_{m-1}(x) + \gamma h_m(x)) \quad (2.34)$$

The initialization of the Gradient Boosting Classifier involves setting up an initial model $F_0(x)$, usually a constant, which is optimized to reduce the loss function across the training data. Subsequently, each weak learner is trained on the residual errors of the previous model, and the ensemble is built iteratively, with each new model added aiming to minimize the remaining misclassification errors. This process continues until *n_estimators* models are built or another stopping criterion is met.

Practical considerations include the need to control overfitting, especially when increasing the number of estimators or tuning other model parameters. The training process of boosting is inherently sequential, which can lead to lengthy training times, especially when working with substantial datasets. To achieve the best results, it's crucial to meticulously adjust various parameters, including the number of estimators, the learning rate, and the depth of the trees.

Gradient Boosting Classifiers are extensively used in competitive machine learning and have proven effective in various domains including finance, biology, and risk management. They are particularly valued for their predictive accuracy and flexibility.

2.6.1.8 AdaBoost Classifier

The AdaBoost (Adaptive Boosting) Classifier is a robust ensemble method that integrates several simple classifiers, often decision trees, to create a more effective composite classifier. The AdaBoostClassifier, when specified with *n_estimators* employs n such basic classifiers sequentially. Each new classifier in the sequence places a higher emphasis on the training examples that were misclassified by its predecessors.

AdaBoost works by assigning weights to each training instance, which are adjusted as learning progresses. Weak learners are added sequentially, each correcting its predecessor, to improve the ensemble's performance. The method increases the weights of the misclassified instances, ensuring that the following classifiers pay more attention to these challenging examples. The final decision of the AdaBoost ensemble is derived from a weighted aggregate of the predictions made by all the individual classifiers [23]. Mathematically, the final model can be expressed as:

$$F(x) = \text{sign} \left(\sum_{i=1}^N \alpha_i h_i(x) \right) \quad (2.35)$$

where:

- $h_i(x)$ is the prediction of the i -th weak learner.
- The weights, denoted by α_i , are assigned to each classifier's output based on its performance accuracy.
- N is the total number of weak learners.

The training process of the AdaBoost Classifier begins by assigning uniform weights to all training instances. Each iteration involves training a weak learner on the weighted data, calculating its error rate, and determining the learner's weight α , which inversely depends on the error rate. Training instance weights are updated to increase those of misclassified instances, thereby focusing subsequent learners on these harder cases. This iterative cycle continues until n learners are formed or the error rate reaches zero.

In practice, AdaBoost is less prone to overfitting compared to other algorithms but can still overfit on noisy data. The complexity of weak learners needs careful tuning: overly complex models may overfit, while overly simple ones may fail to capture sufficient information. Additionally, AdaBoost is sensitive to noisy data and outliers because of its focus on fitting every point perfectly.

AdaBoost is widely used in applications where prediction accuracy is critical, such as customer churn prediction, classification of types of topics within documents, and facial recognition systems.

2.6.1.9 Extra Trees Classifier

The Extremely Randomized Trees Classifier, or Extra Trees Classifier, is an advanced ensemble learning method that enhances the random forests approach by introducing additional randomness into the model. By modifying how the splits are chosen in the construction of the trees, Extra Trees can often achieve better performance and increased computational efficiency. This model uses a user-specified number of trees, $n_estimators$, which controls how many trees are constructed before making a prediction.

Extra Trees constructs multiple decision trees during training. Unlike classic decision trees, which choose the optimal split at each node, Extra Trees selects splits randomly, and among these, the best split is chosen to split the node. This method can increase diversity among the trees in the model, potentially leading to better generalization over

unseen data. This method involves averaging outcomes from multiple trees to predict continuous targets and uses majority voting for categorical outcomes [25].

The training process of the Extra Trees Classifier involves the independent construction of a specified number of trees, denoted as $n_estimators$, utilizing the entire training set. Each individual tree within the ensemble is constructed by randomly selecting feature subsets and corresponding thresholds for splits, effectively reducing the model's variance without notably increasing bias. The trees in this model are grown to their fullest extent without any pruning, allowing for a comprehensive interplay between features, which aids in capturing intricate patterns within the data.

In terms of practical considerations, the number of trees specified by $n_estimators$ is critical and should be selected carefully to balance the trade-off between overfitting and underfitting; more trees generally reduce variance but increase computational demand. Due to the independent nature of tree construction, the Extra Trees Classifier is well-suited for parallel processing across multiple CPUs, which can expedite the training phase considerably. Unlike many other predictive models, Extra Trees does not require feature scaling, making it more straightforward to prepare data and maintain the integrity of original measurement scales.

Extra Trees Classifier is used in various applications where robustness against noise is crucial, such as in medical diagnosis, financial analysis to predict stock movements, and in biology for gene classification. It is favored for its efficiency and effectiveness in handling high-dimensional data.

2.6.1.10 Gaussian Naive Bayes classifier

The Gaussian Naive Bayes classifier is a specific type of Naive Bayes classifier that assumes a Gaussian (normal) distribution for the likelihood of the features given a class. This principle simplifies the computational process and proves particularly useful when handling continuous variables, especially when the Gaussian distribution provides an adequate model of the underlying data distribution [33].

The "naive" assumption in Naive Bayes refers to its treatment of features as independent. This basically says that once a document's category is determined, words in the document are independent of one another in the context of classification. While this might not always be true (words can be related!), Naive Bayes prioritizes simplicity and efficiency. This allows for faster calculations and makes the algorithm computationally efficient [49].

Gaussian Naive Bayes calculates the probability of a class c given features x using Bayes' theorem:

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \quad (2.36)$$

where:

- $P(c|x)$ is the posterior probability of class c given predictor x .
- $P(x|c)$, is modeled as a Gaussian distribution:

$$P(x_i|c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} \exp\left(-\frac{(x_i - \mu_c)^2}{2\sigma_c^2}\right) \quad (2.37)$$

- $P(c)$ is the prior probability of class c .
- $P(x)$ is the prior probability of predictor x .

Each feature x_i is assumed to be independent. The parameters μ_c and σ_c^2 (Mean and variance) are estimated from the training data.

The Gaussian Naive Bayes classifier is trained by calculating the mean μ_c and variance σ_c^2 for each class from the training dataset. These parameters represent the average and variability of the features within each class c . This process does not require iterative steps as the parameters are derived directly from the data, simplifying the training phase and enabling a straightforward implementation of the classifier.

Practical considerations for deploying Gaussian Naive Bayes include the critical examination of the independence assumption and the Gaussian distribution assumption of features. The independence assumption simplifies calculations but may lead to suboptimal performance if it does not hold, which is often the case in real-world data.

Additionally, if the feature distribution significantly deviates from Gaussian, the model's effectiveness may decrease, necessitating checks on the distribution of features to ensure the appropriateness of using Gaussian Naive Bayes. Common applications of this classifier include text classification, medical diagnosis, and scenarios where the data attributes are continuous and approximately follow a normal distribution.

2.6.1.11 Multilayer Perceptron Classifier

A Multilayer Perceptron (MLP) Classifier is a neural network model used primarily for classification. It features several layers of nodes arranged in a directed graph, where each layer is completely interconnected with the subsequent one. Except for the input nodes, every node employs a nonlinear activation function. The MLP classifier with n number

of iterations configures the network to iterate up to n times over the training data to minimize the error between predicted and actual classifications.

The MLP makes use of a back-propagation algorithm for training, which effectively updates the weights of the neurons. In mathematical terms, each neuron's output is calculated as [70]:

$$o_i = \sigma \left(\sum_j w_{ij} x_j + b_i \right) \quad (2.38)$$

In neural networks, the function σ represents the activation function, which can be types like Sigmoid or ReLU. The w_{ij} denote the weights, x_j the inputs from a preceding layer, and b_i the bias term. Typically, the network error is evaluated using a loss function, with cross-entropy loss being a common choice for classification tasks.

$$L = - \sum_{c=1}^M y_{o,c} \log(p_{o,c}) \quad (2.39)$$

where, $y_{o,c}$ is the binary indicator (0 or 1) if class label c is the correct classification for observation o , and $p_{o,c}$ is the predicted probability that observation o is of class c .

The process of training a Multilayer Perceptron (MLP) Classifier involves several steps. Initially, weights are randomly assigned. Training then progresses through a series of iterations, or epochs, during which the network processes inputs to generate outputs, compares these outputs to true values to determine errors, and adjusts the weights and biases via back-propagation to reduce these errors. Training continues until reaching a maximum number of epochs or until an early stopping mechanism is triggered to avoid overfitting.

Practical considerations when deploying an MLP include managing overfitting, particularly as network complexity increases, which can be mitigated by implementing techniques such as dropout, regularization, and early stopping. The learning rate is a critical parameter in this process. If it is set too low, the training can be unnecessarily slow; if too high, it may prevent the network from converging to a good solution.

Additionally, the selection of activation functions plays a significant role in the network's ability to converge and in the performance of the model. MLPs are widely employed in complex tasks such as image and speech recognition, where they model intricate patterns in large-scale data.

2.6.2 Deep Learning Model

The Sequential model [16] is a foundational approach in deep learning for addressing classification tasks. This method constructs models layer-by-layer for problems where inputs explicitly map to outputs. It is particularly suited for simple architectures, such as those typically found in classification tasks where data flows in one direction from input to output.

Model Architecture

The model is structured as a Sequential model from the Keras API, facilitating the linear stacking of layers. As shown in figure 2.10, the architecture comprises the following layers:

- **Dense Layers:** These are fully connected layers characterized by:

$$\mathbf{z}^{[l]} = \mathbf{W}^{[l]} \mathbf{a}^{[l-1]} + \mathbf{b}^{[l]}, \quad \mathbf{a}^{[l]} = g^{[l]}(\mathbf{z}^{[l]}) \quad (2.40)$$

where, $\mathbf{W}^{[l]}$ and $\mathbf{b}^{[l]}$ are the weight matrix and bias vector for layer l , respectively, $\mathbf{a}^{[l-1]}$ is the activation from the previous layer, and $g^{[l]}$ is the activation function. For the hidden layers, the ReLU function is used, defined as:

$$g(z) = \max(0, z) \quad (2.41)$$

- **Dropout Layers:** Integrated to mitigate overfitting by randomly setting a fraction of the input units to 0 during training. For each layer, a dropout rate of 0.5 is utilized, which results in the inactivation of roughly half of the neurons from the preceding layer during each phase of training.
- **Output Layer:** Configured dynamically based on the classification task:

- Binary classification uses a single neuron with a sigmoid activation function:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (2.42)$$

- Multi-class classification employs a softmax function across multiple neurons:

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \quad (2.43)$$

where, z_i is the input to neuron i in the output layer.

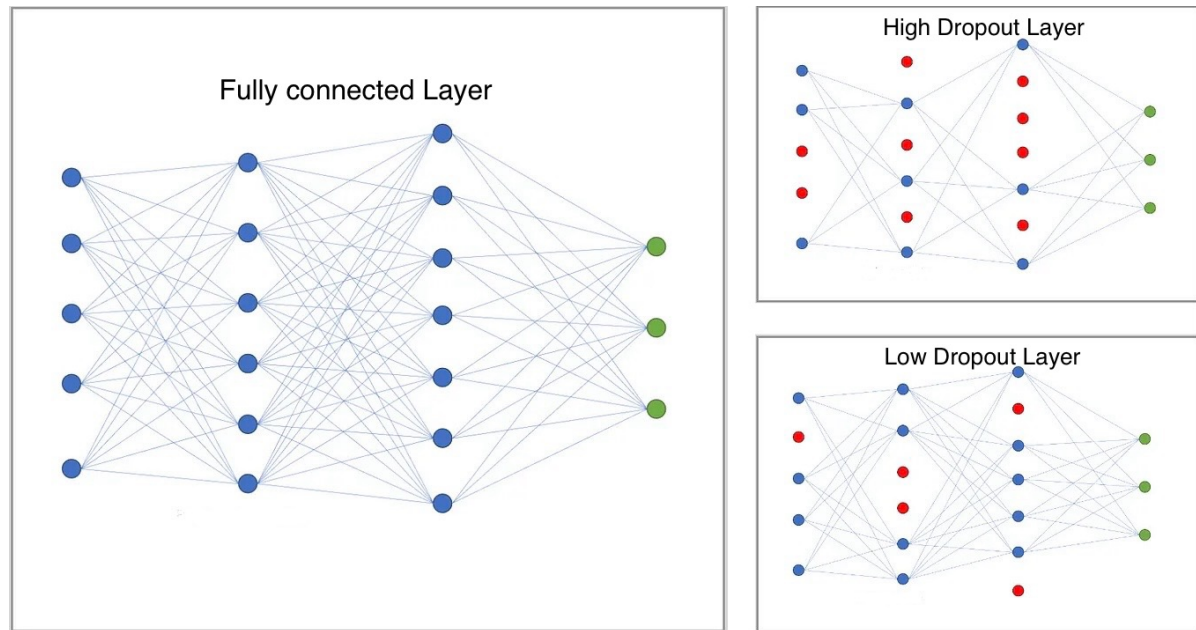


Figure 2.10: Types of Dropout Layers with Fully connected layer [47].

Model Compilation and Training

When implementing a Sequential model, practical considerations are crucial for optimizing performance and ensuring generalization to unseen data. Key aspects include thorough data preprocessing, such as normalization or scaling, to prepare the input data effectively. Fine-tuning hyperparameters is crucial for optimizing a neural network model. This includes modifying the layer count, neuron count, dropout rates, learning rates, and training duration to establish the most efficient model setup. To mitigate the risk of overfitting, a prevalent challenge in neural networks, strategies like dropout, data augmentation, and early stopping are recommended. Additionally, it is vital to employ appropriate metrics for evaluating the model, with particular attention to differentiating between training and validation metrics, ensuring that the model not only learns well but also generalizes well to new, unseen datasets.

The model is compiled using the Adam optimizer, an algorithm for first-order gradient-based optimization of stochastic objective functions, based on adaptive estimates of lower-order moments [38]. The loss function varies with the task type, using binary cross-entropy for binary tasks and sparse categorical cross-entropy for multi-class tasks.

Chapter 3

Related Works

This chapter examines the diverse applications of millimeter wave technology and data representation strategies in skin cancer classification. The reviewed literature not only outlines the evolution of mmWave technology but also explores its promising intersection with advanced diagnostic methods for skin cancer. This chapter delves into the essence of literature that intersects the domains of mmWave technology, data representation strategies, and their application in skin cancer classification, laying a solid foundation for understanding the novelty and necessity of the current study.

Studies highlighted in this chapter illustrate how mmWave radar data, coupled with innovative data fusion and AI techniques, offers significant advancements in detecting and classifying skin cancer at early stages, potentially revolutionizing patient care.

The significant advancements in millimeter wave (mmWave) technology have enabled its utilization in various domains, from telecommunications to bio-medical engineering.

This chapter will delve into the core concepts and recent progress in mmWave imaging. It will cover the operational methods, the development of related technologies, and the specific physical properties that render mmWave technology appropriate for bio-medical uses.

In Fahad Jibrin Abdu et al. [2] explore the integration of radar data with deep learning technologies. The paper highlights the underutilization of radar compared to other sensors like Lidar and cameras, despite its robust performance in adverse weather conditions and its capability to measure an object's range, velocity, and angle simultaneously. The authors provide a historical overview of autonomous driving advancements, noting significant milestones such as DARPA's urban challenges and the ongoing issues in developing reliable autonomous systems due to sensor limitations in challenging environments.

The introduction and analysis are well-supported by a broad range of studies, including works by F. de Ponte Müller [19] on radar technologies in vehicles and references to deep learning breakthroughs that have driven progress in multimodal sensor fusion for accurate object detection [26, 42].

The body of the review meticulously categorizes and discusses various deep learning models that have been adapted to work with radar data, offering insights into the potential and challenges of these integrations. The authors critique the sparse nature of radar detections compared to Lidar and the complexities of using radar data in deep learning frameworks, citing the need for better radar signal representations and the development of dedicated deep learning models that can effectively handle these signals. The final section of the paper advocates for the creation of additional open-access datasets containing radar data to enhance the scope of research and enable comparative analysis, reflecting the challenges faced by researchers who are compelled to develop their own proprietary datasets [55]. This paper describes a fusion of radar point clouds and RGB images through a middle-fusion framework. It uses 3D proposals based on radar data, which are then refined using image feature maps in the Radar Proposal Refinement (RPR) network. The fusion of data from different sources (radar and camera) to improve detection and distance estimation mirrors the potential in this thesis research to combine mmWave radar frequency domain data with other modalities (like optical imaging) to improve skin cancer classification. The method of using radar to propose candidate regions and then refining these proposals could be adapted to propose areas of potential malignancy based on mmWave data.

X. chen et al. [15] utilizes a multi-view representation of LIDAR point clouds combined with RGB images. It encodes sparse 3D point cloud data into a compact form, allowing for efficient handling and processing. The concept of transforming raw 3D data into a more compact, multi-view representation can be paralleled in your study by representing mmWave radar data in various forms that highlight different aspects of skin tissue properties. Just as MV3D uses different views to enhance object detection, different frequency domain representations of radar data might enhance the distinction between malignant and benign lesions.

In both cases [15, 55], the sophisticated handling and representation of data to effectively capture and utilize complex information can be adapted to this thesis focus.

Khushaba et. al's [37] study on material classification via radar leveraged raw radar signals, which were processed using the Wavelet Scattering Transform (WST) to extract features, and subsequently refined through the Spectral Regression (SR) feature projection method. The research explores the comparison of microwave and millimetre waves

(Walabot and ImageVk). The characteristics were analyzed via Linear Discriminant Analysis (LDA) in both simulated and real-time evaluation settings. Offline evaluations utilized confusion matrices derived from 10-fold cross-validation, where the Walabot demonstrated slightly higher accuracy with an average of $92.33 \pm 5.49\%$ on the main diagonal of the confusion matrix compared to the IMAGEVK-74's $90.66 \pm 11.40\%$. Notably, the IMAGEVK-74 tended to confuse materials with similar textures, such as broad beans and chickpeas. In online testing, although the IMAGEVK-74 processed a larger volume of raw time-domain signals, leading to faster computational performance, Walabot's results showed better consistency and less class confusion. These findings highlight the nuanced trade-offs between the two radar units and suggest the need for further investigation into their application in environments with mixed material samples, as might be encountered in industrial applications and potentially in interplanetary scenarios.

Yeo et al. [82] introduce RadarCat, a compact radar-based system designed for material and object classification that facilitates innovative interactions with digital devices. The system's efficacy was evaluated through three studies involving diverse materials, including 26 different substances, 16 types of transparent materials with varying properties, and body parts from six individuals, using robust methodologies such as leave-one-out and 10-fold cross-validation. These studies demonstrated the system's ability to accurately and reliably classify materials using a random forest classifier, enabling real-time recognition. The authors illustrate RadarCat's practical applications through several prototypes, including a physical object dictionary and interfaces for painting, photo editing, and automated refilling tasks. They conclude by affirming the versatility and robustness of RadarCat, discussing its potential to enhance everyday interactions, and suggesting its use in conjunction with other sensing technologies to develop advanced mobile, wearable, and context-aware interfaces.

Weiß et al. [80] introduce a 60-GHz radar-based material classification system utilizing a Deep Convolutional Neural Network (DCNN) that processes images generated from radar signals through Capon beamforming. This novel approach incorporates dilated convolution layers in the DCNN, adapting to the unique physical characteristics of radar images and enhancing feature representation. The use of image augmentation techniques such as random erasing, flipping, and brightness adjustment significantly reduces the need for extensive training data while improving classification accuracy. Their experimental setup included a dataset of 1875 images representing ten different materials and everyday objects, achieving an impressive classification accuracy of 97.81%. The system demonstrated robust performance against sensor noise and device nonlinearities, suggesting potential applications in environments where noise variance is a concern. Looking forward, the researchers plan to optimize the network for embedded systems, explore the impact of

material casings on classification accuracy and develop a vibration cancellation feature to facilitate aerial deployment of the sensor system.

Topfer et al. [78] explore the historical and contemporary use of microwave and millimeter wave technologies for medical diagnostics, particularly focusing on the dielectric properties of tissues and their frequency-dependent behaviors as potential diagnostic tools. Initially observed in the 19th century, and more extensively studied post-World War II, these properties facilitate the differentiation between types of tissues, including the identification of cancerous tissues, which exhibit notably different permittivities from healthy tissues at specific frequencies. Their review emphasizes the significant potential of millimeter wave sensing in medical applications such as non-invasive cancer detection and blood glucose monitoring, despite existing challenges like limited penetration depth and the need for high-resolution tissue imaging. They advocate for continued research and development to refine these technologies, enhance sensor systems from proof-of-concept to clinical implementation, and establish comprehensive datasets on tissue characteristics at millimeter wave frequencies, aiming to integrate these advancements into routine healthcare practices for improved diagnostic accuracy and patient outcomes.

Chan et al. [13] explore the development of a millimeter-wave (mm-wave) device for detecting cancerous tissue, specifically basal cell carcinoma (BCC), using a cost-effective silicon-based probe suitable for skin cancer detection. Operating within the 90–300 GHz spectrum to achieve millimeter precision and detect tumors up to 0.5 mm deep, the study employs electromagnetic (EM) modeling and high-frequency simulations (ANSYS HFSS) to assess the probe's effectiveness. The simulation involved two scenarios: one with BCC on the surface and another with BCC embedded at various depths in normal skin, also considering a $\pm 10\%$ variability in dielectric properties. The results demonstrate the probe's capability to differentiate between normal and cancerous tissue by analyzing both the magnitude and phase of the reflection coefficients. A dual-band approach (100–150 GHz and 200–250 GHz) was identified as most effective for highlighting deviations from normal skin properties. The findings suggest that this mm-wave technology could lead to a compact, handheld device for early skin cancer detection, pending further validation through *in vivo* preclinical studies and *in vitro* tests with human skin samples.

In research, Arab et al. [4] investigate the potential of millimeter-wave (mm-wave) radar technology to identify melanoma in its early stages by differentiating the dielectric characteristics of melanomas from those of normal skin tissues. Utilizing a 77 GHz multitone continuous-wave radar system, the team developed a low-power sensor with a Miniature Hybrid Microwave Integrated Circuit (MHMIC) on a thin ceramic substrate, incorporating dual 16-element patch antenna arrays and a sophisticated six-port homodyne quadrature

down-converter. The sensor's design allows for high-resolution detection of skin anomalies with precision up to tens of microns, crucial for early-stage cancer diagnosis. This integrated sensor system is encapsulated in a compact metallic fixture, enhancing its practical application in clinical settings. The research detailed the sensor's capability to produce and analyze reflection and transmission coefficients, employing a wide frequency range to assess various sample materials including melanoma, wet skin, dry skin, and water, demonstrating high phase resolution and measurement accuracy.

Further experimental validations contrast the dielectric properties of different sample types—wet hand, dry hand, and water—illustrating distinct transmission coefficients that are pivotal for accurate skin condition diagnostics. The experimental results were corroborated by simulations, indicating consistent performance across diverse test scenarios. Despite the compact and cost-effective design of the radar system, which promotes its use in in-vivo settings for direct application on the skin, the study suggests the necessity for clearer definitions in experimental setups, such as the specification of skin sample sizes and the conditions of water used. Moreover, it is proposed to extend testing to actual melanoma cases and possibly refine the system's design to enhance its application scope. This research highlights the promising role of millimeter-wave (mm-wave) radar technology in bio-medical fields, particularly in the non-invasive detection of skin cancer. This paves the path for the development of cost-effective, highly accurate diagnostic instruments.

In Owda et al. [61] utilize millimeter-wave (mmWave) radiometry to assess human skin reflectance, exploring its potential for diagnosing skin conditions non-invasively. In this study, a non-contact sensor operating within the 80-100 GHz spectrum was used to assess skin reflectance in 50 healthy volunteers under varying conditions (dry and wet) and across different body sites, including the palm, the back of the hand, and the forearm. They found significant reflectance differences attributable to factors like skin thickness, hydration level, and biological sex, with thinner skin areas showing higher reflectance than thicker ones. The outcomes of this research are pivotal. They demonstrate the capability of mmWave sensors to distinguish between various skin attributes associated with diseases such as eczema and psoriasis, among other skin disorders. Highlighting the value of mmWave technology in medical diagnostic contexts, this study lays a foundation for future research aimed at establishing standard reflectance values for diverse populations and skin conditions. This could significantly enhance the early detection and ongoing monitoring of skin-related health issues.

In the study by Kao, Jen-Li, and Chao, Yi-Ping [36], an innovative approach to breast cancer imaging utilizing millimeter-wave (mmWave) radar technology is explored. This method stands out due to its employment of high-frequency mmWave radar, specifically

operating within the 62 to 69 GHz range, which enables enhanced resolution in the imaging process. The core of their data representation involves the transformation of raw radar signals into 3D images that accurately depict the internal structure of a breast phantom embedded with a tumor phantom.

The data representation technique in their research harnesses the high-resolution capabilities of mmWave radar facilitated by an antenna array with 20 transmit (Tx) and 20 receive (Rx) antennas. This setup is designed to capture precise radar measurements, which are essential for differentiating the dielectric characteristics of cancerous and non-cancerous tissues. The collected data undergoes a noise reduction process using an averaging method, which improves the clarity and usability of the signals for image reconstruction.

For imaging reconstruction, the researchers utilize a delay-and-sum algorithm, which calculates the signal's time delay and amplitude at each spatial pixel, reconstructing a detailed 3D image of the target area. Using this approach, each pixel in the three-dimensional image accurately reflects the distribution of dielectric properties across the space. This high-resolution detailing is crucial for the detection and precise localization of tumors within a breast phantom model.

The relevance of this data representation method to this thesis lies in its application of mmWave radar for medical imaging, specifically in cancer detection. By adopting similar high-frequency radar techniques, skin cancer classification could achieve higher accuracy and resolution. The detailed data representation methods developed for breast cancer imaging could be adapted to enhance the detection and characterization of skin lesions, potentially leading to improved diagnostic processes in dermatology. This crossover suggests a promising avenue for research into the use of mmWave technology in different types of cancer diagnosis, offering a non-invasive, safe, and cost-effective alternative to traditional imaging techniques.

Recent advancements in artificial intelligence (AI) have notably enhanced the capabilities of healthcare professionals, allowing for improved accuracy in diagnostics and more informed decision-making. AI models, particularly effective in non-invasive medical imaging techniques such as dermoscopy, enhance the analysis of dermoscopic images, improving the detection and diagnosis of skin conditions.

Moreover, the integration of AI into dermatology, especially in skin cancer detection, represents a transformative development. Various AI algorithms, encompassing diverse neural network architectures, are now routinely utilized to process both image-based and non-image data, facilitating the early diagnosis of critical skin cancers like basal

cell carcinoma and melanoma. This early detection is crucial for administering effective treatment. Despite the benefits, challenges persist that can impact diagnostic accuracy, highlighting the growing relevance and application of image processing and machine vision within medical diagnostics, aiming to refine these technologies further for enhanced reliability and precision.

Purwins et al. [63] examine the application of state-of-the-art deep learning techniques across various audio signal processing domains. They discuss how deep learning, particularly through the use of convolutional neural networks (CNNs), long short-term memory networks (LSTMs), and audio-specific models, can enhance the interpretation of diagnostic images by learning task-optimized representations from raw data inputs, such as log-mel spectrogram or raw waveforms.

In their comprehensive analysis, Buettner et al. [10] discuss the significant role of deep learning in medical image analysis, emphasizing its revolutionary contributions to the detection and diagnosis of cancers in areas like the breast, lungs, and brain. The review underscores the profound efficacy of deep learning, notably convolutional neural networks (CNNs), in enhancing the identification, classification, and segmentation of cancer tissues. Specific advancements include volume segmentation of CT scans, which significantly speeds up and increases the accuracy of cancer tissue analysis by automating the segmentation process through a CNN-based system. This system divides the image volume into sub-volumes for detailed analysis, thereby mitigating the manual workload traditionally required in histopathological examination, which is both time-consuming and error-prone. Particularly in breast cancer detection, the integration of CNNs with Support Vector Machines (SVMs) has achieved up to 92% accuracy in identifying cancerous tissues, demonstrating deep learning's potential to substantially reduce the need for specialist pathologists and expedite early treatment initiation. This synthesis of deep learning techniques in medical imaging paves the way for more efficient, reliable, and faster diagnostic processes, which is crucial for conditions requiring rapid intervention, such as skin cancer.

In the study by Rathi et al. [66], delve into the utilization of machine learning for classifying various radar targets at multiple levels, including air, sea, and land movements. Their study offers insights that could potentially be applied to medical fields, such as detecting skin cancer, by adapting these machine learning approaches. The research demonstrates the use of supervised learning to enhance classification at various levels: signal level based on spectral content, detection level to optimize resource use, and post-tracking level to integrate and refine the classification results. This tiered approach to classification, particularly the emphasis on varying accuracy levels and the effective use of a confusion matrix

to evaluate classifier performance, mirrors potential strategies in skin cancer diagnosis, where AI could similarly stratify skin lesion images based on morphological features at different stages of detection and analysis. The effectiveness of their system in reducing resource expenditure and improving operational efficiency by filtering out non-relevant targets also underscores the potential for AI in dermatology to enhance diagnostic precision and reduce the workload by focusing on clinically significant anomalies.

Carrera et al. [11] present their work on the integration of machine learning into radar target detection systems. They showcase the effectiveness of using random decision forests alongside recurrent neural networks, notably those based on long short-term memory (LSTM) models, to enhance the accuracy in differentiating between actual targets and mere clutter. Their study showcases that machine learning algorithms, especially deep learning, significantly enhance detection capabilities at low signal-to-clutter ratios (SCR) while maintaining controlled false alarm rates. These advancements suggest potential parallels in medical imaging, such as skin cancer diagnosis, where AI could similarly improve the identification of cancerous lesions from benign conditions in dermoscopic images. The results underscore the evolving role of AI in refining diagnostic accuracy, making a case for incorporating deep learning to modernize and enhance skin cancer detection tools.

In their research, D. Lee et al. [43] explore the potential of machine learning to improve radar-based object classification. They use a convolutional neural network (CNN) model influenced by AlexNet to address these challenges. Their work focuses on enhancing the interpretation of radar echo signals, which are complex and vary widely depending on the object's features. They utilize spectrogram analysis to capture unique micro-Doppler signatures from radar reflections, which their CNN model is trained to recognize and classify. This method allows for the differentiation of subtle movements and characteristics of various objects, demonstrating a significant improvement over traditional deterministic analysis methods. Implemented on an Intel® Xeon CPU and Intel® Arria 10 FPGA with the Intel® OpenVINO toolkit, their system achieves real-time automated object classification with an accuracy exceeding 90% on the tested radar dataset. This research exemplifies the potential of advanced neural networks in processing non-visual sensor data, particularly in applications like automotive technologies, where accurate and rapid classification of environmental objects is critical.

F. Mammadova et al. [46] explore the innovative integration of radar and image data for enhanced melanoma detection, addressing the limitations of traditional dermoscopy which lacks crucial depth information. Their study proposes the use of artificial intelligence (AI) to analyze fused data from both radar, capable of penetrating to depth information,

and conventional 2D imaging, to improve the classification accuracy of skin cancer. By employing an early fusion technique, they amalgamate features from radar and image data, demonstrating a substantial 36% increase in diagnostic accuracy in preliminary tests involving skin with nevi, birthmarks, and other areas like the palms and arms.

A. Mirbeik et al. [54] exemplifies the integration of artificial intelligence in medical data analysis for cancer diagnosis, specifically in electromagnetic imaging for tumor margin identification. Their novel deep learning technique, termed DeepTMI, significantly advances the segmentation of tumor versus normal tissues within electromagnetic images. This method leverages a recurrent auto-encoder network architecture to analyze and segment medical images with high accuracy. By formulating deep learning with electromagnetic image reconstruction, they demonstrate a sophisticated approach to medical imaging that aligns with the characteristics of AI-driven analysis.

In their study [54], DeepTMI achieved impressive performance metrics, with a structure similarity measure (SSIM) and mean-square-error (MSE) far surpassing those of traditional backpropagation methods. Accurately distinguishing between cancerous and non-cancerous tissues is vital for meticulous surgical preparation and enhancing patient outcomes. The method's effectiveness in processing complex image data makes it relevant to mmWave data representation analysis in skin cancer classification, where similar AI techniques could enhance the accuracy and resolution of diagnostic imaging tools.

Despite challenges such as small sample size and the nascent stage of radar data application in dermatology, their findings underscore the potential of combining these technologies to significantly refine the early detection and classification of melanoma, paving the way for future research to expand on these methodologies with larger datasets and advanced machine learning models.

Chapter 4

Methodology

This chapter presents the research methodology utilized to investigate how effective millimeter wave data is in identifying skin cancer types. The millimetre wave technology has indicated new opportunities for non-invasive diagnostic tools in medical imaging and disease identification. Specifically, the study leverages sophisticated signal processing methods and machine learning techniques to differentiate benign from malignant skin abnormalities using millimeter wave data.

As shown in figure 4.1 and in flowchart 4.6, this chapter provides a comprehensive overview of the data acquisition process, detailing the characteristics of the millimetre wave radar system used, the nature of the collected data, and the conditions under which the data were captured. Following this, we delve into the data preprocessing methods employed to prepare the raw millimetre wave scans for analysis. This includes techniques for noise reduction, normalization, and feature extraction, which are critical for enhancing the accuracy and efficiency of the subsequent classification models.

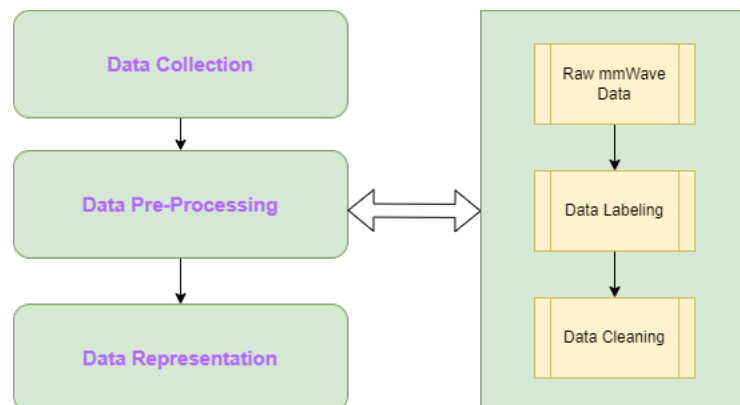


Figure 4.1: Overview of the methodology

4.1 Data Collection

This thesis focuses on creating a model that utilizes millimeter-wave (mm-Wave) radar data to identify and categorize skin lesions. However, there currently exists no public dataset featuring mm-Wave data specific to skin lesions. Consequently, it is essential to construct a dedicated dataset for this research.

In this section on Data Collection, the process of gathering and preparing the data necessary for the study of classifying skin lesions using mm-Wave radar technology is detailed. The section is divided into three parts to ensure clarity and thoroughness. Firstly, the mm-wave radar device used in this research is described, including its technical specifications and operational parameters. Secondly, the nature of the data collected by the mm-wave radar is discussed. Lastly, the meta-data associated with each person from whom data were collected.

4.1.1 IMAGEVK-74 - The mmWave Radar

In modern radar technology, mmWave (millimeter wave) radar systems represent a significant advancement, especially with the introduction of devices such as the IMAGEVK-74. This innovative tool harnesses Vayyar's advanced RFIC (Radio Frequency Integrated Circuit) technology and comprehensive radar intellectual property, consolidated into a compact evaluation kit. The IMAGEVK-74 is designed to facilitate the exploration and realization of mmWave imaging and sensing applications at a fraction of the cost and complexity traditionally associated with such developmental endeavors.

Key Features:

- **Antenna Configuration:** The unit's 40 antennas (20 Tx and 20 Rx) enhance the radar's ability to form a dense data acquisition network, crucial for creating detailed spatial imaging maps.
- **Frequency Range:** Operating within the 62 to 69 GHz range, the IMAGEVK-74 can penetrate various materials, making it ideal for applications in security, structural monitoring, and autonomous vehicle systems.
- **Transmit Profiles:** The device offers three performance-optimized transmit profiles, allowing researchers and developers to tailor the radar's output to suit specific experimental needs or application requirements.
- **Direct Access to Phasors:** A distinctive capability of the IMAGEVK-74 system is its direct provision of the Tx/Rx pair phasors at each frequency point it sweeps.

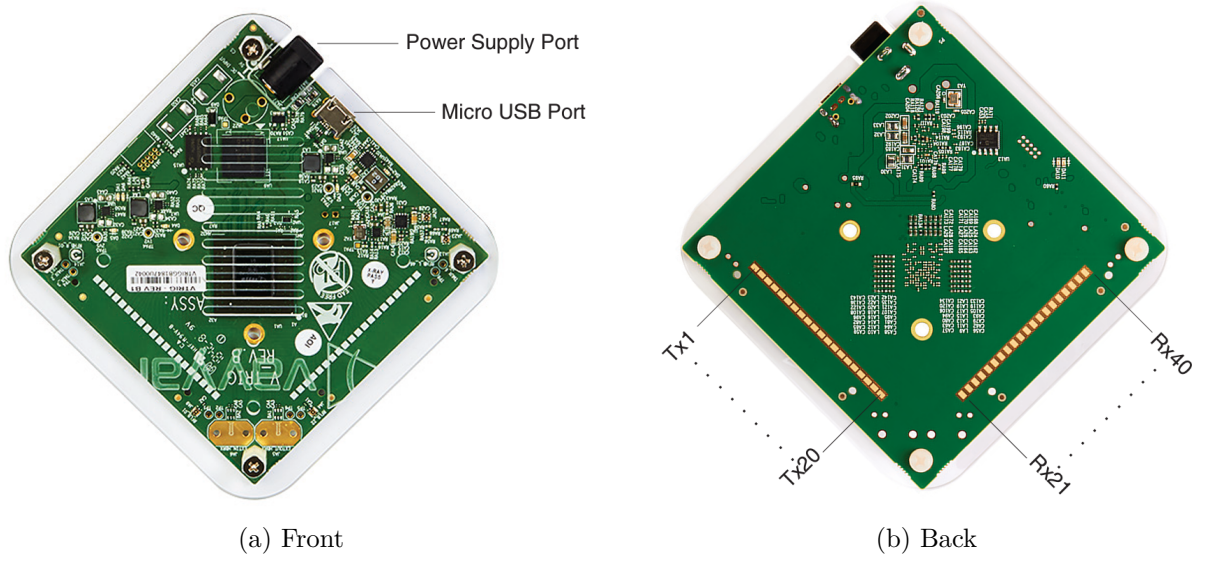


Figure 4.2: IMAGEVK-74 - The mmWave Radar [52]

This feature is critical for researchers who require detailed phase information to accurately interpret signal patterns and properties.

Such flexibility is pivotal for detailed mmWave imaging and sensing, as it provides the capability to penetrate different materials and capture high-resolution data.

4.1.2 Data Set

Data collection involved volunteers from the embedded systems department who consented to participate. Before commencing the data gathering, participants received a briefing about the objectives of the study and gave their verbal consent. Only those who agreed were included in the research. To protect privacy, all personally identifiable information was omitted to maintain participant confidentiality and anonymity.

The IMAGEVK-74 mmWave radar system is equipped with an advanced design featuring 20 transmitting (Tx) and 20 receiving (Rx) antennas, enabling the generation of a comprehensive data matrix. Each of the 400 unique Tx/Rx pairs captures measurements across 150 frequency points, resulting in a data matrix of 400x150 frequency points.

This high-resolution data collection is essential for capturing detailed characteristics in the mmWave frequency domain, which is vital for precise spatial and material characterization. The data is initially collected in the Frequency Domain directly from the radar outputs, providing a detailed matrix of 400 transmitter and receiver pairs spanning a wide frequency range.

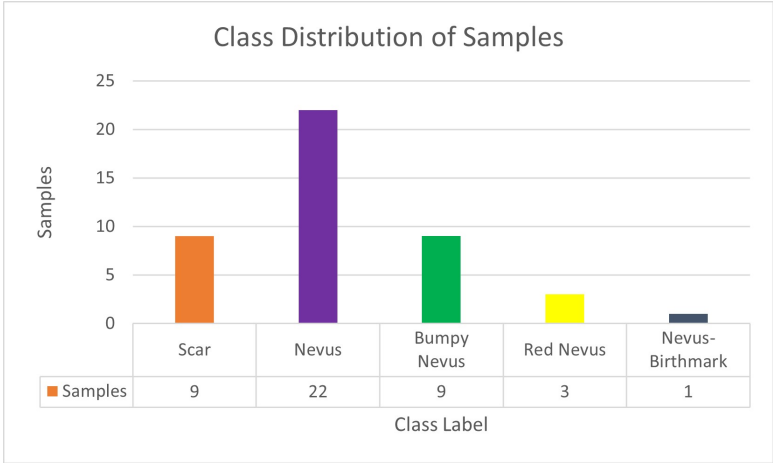


Figure 4.3: Class Distribution of Samples

In total, 44 samples consisting of images and their corresponding mm-wave data were collected. For each sample, 50 instances were gathered and stored in .txt format. The data gathered were organized into a .csv file using Microsoft Excel, which was then prepared for subsequent analysis [64].

4.1.3 Meta-Data

The meta-data accompanying the collected mm-wave radar measurements plays a pivotal role in the contextual analysis and interpretability of the data. Each data point is associated with a unique identifier (ID), ensuring the traceability and organization of the dataset. Demographic variables include gender and age, which are critical factors in the manifestation of skin pathologies and their responses to treatment. The specific body part location of the skin lesion is documented, providing essential insights into the distribution patterns of skin anomalies. Additionally, the meta-data captures each individual’s skin cancer history, an imperative attribute that informs the propensity for various skin lesion types. Sun exposure is quantified to assess its contributory effect on skin lesion development. Lastly, the label assigned to each data point classifies the lesion as the nevus, textured or raised nevus, and scar.

4.2 Data Pre-Processing

In this section, outlines the critical steps taken to validate and optimize the dataset, ensuring it is suitable for comprehensive analysis. Initially, data labeling is addressed. Following the labeling, data cleaning is described.

Both labeling and cleaning are critical to the development of a reliable dataset that is representative of the population under study and conducive to effective machine learning analysis.

4.2.1 Data Labeling

The Data Labeling explains the steps undertaken to organize and assign definitive labels to the dataset. Unique codes were designated to each sample for identification purposes. Subsequently, the samples were systematically grouped into one of five categories: red nevus, birthmark nevus, bumpy nevus, nevus, or scar, a classification reliant on their physical characteristics as determined by clinical experts. Following this classification, an association of each sample with its detailed meta-data was conducted through the unique ID numbers, ensuring a seamless and accurate merger of sample information with individual data attributes. This concatenation process was vital for the creation of a structured and navigable dataset, which is crucial for future data retrieval and analysis, thereby upholding the dataset's organizational integrity and analytical value.

4.2.2 Data Cleaning

The Data Cleaning explains the rigorous procedures performed to enhance the dataset's quality and ensure its suitability for analysis. The dataset was meticulously scrutinized to identify and correct any discrepancies or omissions, which are essential for preserving data quality. Subsequently, the CSV files containing the dataset were carefully processed; an iteration over the columns was conducted to convert the data values. For each column, except those containing categorical data, the stored string-type values were transformed into complex numbers. This transformation was essential for enabling precise computational analyses, facilitating correct mathematical operations, and preparing the dataset for effective integration into machine learning models.

4.3 Data Representation

The Data Representation explores the transformation and interpretation of mm-wave radar data within various domains. Three primary domains are investigated as illustrated in figure 4.4: the frequency domain, the time domain, and the spectrogram representation, each about a different domain of data analysis and subsequent visualization techniques, which will be further elaborated in subsequent sections.

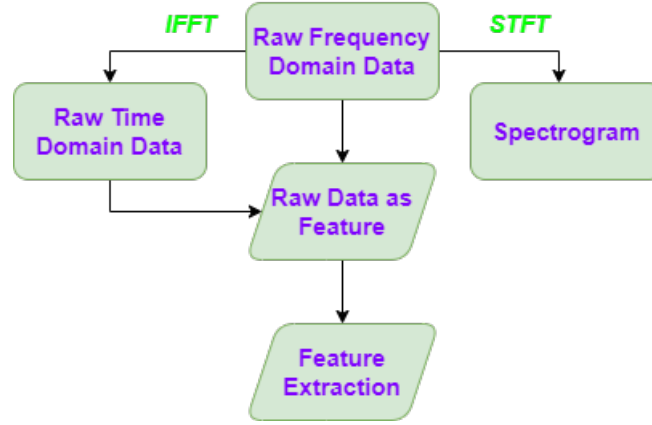


Figure 4.4: Data Representation

4.4 Raw Feature

In this section addressing Raw Features, two distinct classifications are presented, emphasizing the differentiation between processing methodologies in radar data analysis. These classifications are categorized into Raw Frequency Domain Features and Raw Time Domain Features, each on the initial datasets derived from radar outputs.

1. **Raw Frequency Domain Features:** This category encompasses the initial dataset, wherein all columns representing frequency points are converted into their absolute values. The transformation exclusively applies to non-categorical data columns within the data matrix. The conversion preserves the integrity of categorical columns by not subjecting them to the absolute value transformation.
2. **Raw Time Domain Features:** In this segment, we delve into the transformation of spectral data points back into the time domain. This reverse process is facilitated using the inverse fast Fourier transformation (IFFT), which is applied across all the non-categorical columns in the dataset matrix. Similar to the Raw Frequency Domain Features, the categorical columns remain unaltered, ensuring that the categorical integrity of the data matrix is maintained.

4.5 Feature Extraction - Domain Specific

This section deals with the formulation of features extraction and calculation of both frequency and time domain. Both categories collectively facilitate a comprehensive analysis of radar signals, offering insights into different dimensions of the data derived from radar outputs.

4.5.1 Feature Extraction - Frequency Domain

In this subsection, the extraction frequency domain features from raw radar data, including both signal characteristics and categorical information explained. This extraction involves several steps, from initial data separation to the computation of various spectral and statistical features. Here's a breakdown of them:

1. **Separation of Data:** Extracts all columns except which are assumed to be categorical. This subset represents frequency data.
2. **Conversion to Complex Numbers:** Converts the frequency data into a numpy array of complex numbers to facilitate subsequent frequency domain analysis.
3. **Handling Insufficient Data:** Checks if the array of complex numbers has less than two points. If so, returns a list of NaNs for all frequency features, indicating insufficient data, while retaining categorical data.
4. **Computation of Frequency Domain Features:**
 - **Magnitude and Phase:** Computes the magnitude and phase of each frequency point using the absolute value and angle of the complex numbers, respectively.
 - **Spectral Centroid:** This feature represents the "center of mass" of the spectrum, indicating the balance point of the spectrum's shape.

$$\text{Centroid} = \frac{\sum(\text{fft_freq} \times \text{magnitudes})}{\sum(\text{magnitudes})} \quad (4.1)$$

- **Spectral Bandwidth:** Measures the spread of the spectrum around its centroid, describing the spectrum's width.

$$\text{Bandwidth} = \sqrt{\frac{\sum((\text{fft_freq} - \text{centroid})^2 \times \text{magnitudes})}{\sum(\text{magnitudes})}} \quad (4.2)$$

- **Spectral Flatness:** Provides a measure of how noise-like a signal is, with higher values indicating a flatter spectrum.

$$\text{Flatness} = \frac{\exp(\text{mean}(\log(\text{magnitudes} + 1e - 6)))}{\text{mean}(\text{magnitudes})} \quad (4.3)$$

- **Spectral Roll-off:** The frequency below which 85% of the magnitude distribution is concentrated, indicating the spectrum's skewness towards lower

frequencies.

$$\text{Roll-off} = \text{percentile}(\text{magnitudes}, 85) \quad (4.4)$$

5. Statistical Features from Magnitude:

- **Mean, Standard Deviation, Skewness, and Kurtosis:** These statistical measures provide basic descriptive statistics of the magnitudes, capturing the central tendency (mean), dispersion (standard deviation), asymmetry (skew), and tailedness (kurtosis) of the frequency distribution.
- **Average Magnitude and Phase:** Simple averages of the magnitude and phase values, offering a straightforward summary of the signal's overall strength and phase behavior.

This detailed extraction and calculation of frequency domain features enable nuanced analysis of radar data.

4.5.2 Feature Extraction - Time Domain

In this subsection, the extraction of various statistical and signal-processing features from time-domain data derived from radar outputs. This is particularly useful for applications like condition monitoring, anomaly detection, or other analysis where characteristics of a signal are critical. Here's a breakdown of them:

1. **Exclusion of Categorical Data and Function:** The time-domain data excludes the columns which assumed to be categorical, focusing purely on numerical data. A function 'calculate_features' is defined to compute multiple features for each row (i.e., each signal) in the DataFrame.

Formulation of Time-domain Features:

2. **Mean Value:** The average of all data points in the time-domain signal. It represents the central value of the data distribution.

$$\text{Mean Value} = \frac{1}{n} \sum_{i=1}^n x_i \quad (4.5)$$

3. **Standard Deviation:** Measures the amount of variation or dispersion in the signal. A low standard deviation indicates that the data points tend to be close to the mean.

$$\text{std_val} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \text{mean_val})^2} \quad (4.6)$$

4. **Skewness:** Assesses the asymmetry of the data distribution around its mean. Positive skew indicates a distribution with an asymmetric tail extending toward more positive values.

$$\text{skew_val} = \frac{\sum_{i=1}^n \left(\frac{x_i - \text{mean_val}}{\text{std_val}} \right)^3}{n} \quad (4.7)$$

5. **Kurtosis:** Measures the "tailedness" of the data distribution. High kurtosis means a distribution has heavy tails and a sharp peak near the mean, indicating outliers.

$$\text{kurt_val} = \frac{\sum_{i=1}^n \left(\frac{x_i - \text{mean_val}}{\text{std_val}} \right)^4}{n} - 3 \quad (4.8)$$

6. **Peak-to-Peak Amplitude:** The difference between the maximum and minimum values in the time-domain signal, indicating the signal's overall amplitude range.

$$\text{peak_to_peak_amp} = \max(x) - \min(x) \quad (4.9)$$

7. **Signal Magnitude Area (SMA):** The average of the absolute values of the signal data points. It provides an overall magnitude of the signal without considering its direction.

$$\text{sma} = \frac{\sum |x_i|}{n} \quad (4.10)$$

8. **Energy:** The sum of the squares of each point in the time-domain signal, indicating the total power or energy contained in the signal.

$$\text{energy} = \sum x_i^2 \quad (4.11)$$

9. **Signal Entropy:** Measures the regularity or unpredictability of the signal. Higher entropy suggests more randomness in the signal's data distribution.

$$\text{signal_entropy} = - \sum (|x_i| \log_2(|x_i| + \epsilon)) \quad (4.12)$$

where ϵ is a small constant (machine epsilon) to prevent the logarithm from blowing up.

10. **Autocorrelation (Lag 1):** Represents the correlation of the signal with a delayed version of itself, typically used to find repeating patterns or periodic signals.

$$\text{autocorr} = \sum_{i=1}^{n-1} x_i \cdot x_{i+1} \quad (4.13)$$

11. **Maximum Index:** The index at which the maximum value of the signal occurs, which can be useful in locating significant events within a time series.

$$\text{max_index} = \arg \max(x) \quad (4.14)$$

12. **Crest Factor:** The ratio of the peak amplitude of the signal to its root mean square (RMS) value, often used to assess the extremity of a peak.

$$\text{crest_factor} = \frac{\max(|x_i|)}{\sqrt{\frac{1}{n} \sum x_i^2}} \quad (4.15)$$

13. **Form Factor:** The ratio of the root mean square of the signal to the mean of the absolute values of the signal's points, indicating the shape of the signal relative to a sine wave.

$$\text{form_factor} = \frac{\sqrt{\frac{1}{n} \sum x_i^2}}{\frac{1}{n} \sum |x_i|} \quad (4.16)$$

14. **Impulse Factor:** The ratio of the peak amplitude to the mean of the absolute values, used to assess the impulsiveness of the signal.

$$\text{impulse_factor} = \frac{\max(|x_i|)}{\frac{1}{n} \sum |x_i|} \quad (4.17)$$

This process effectively extracts and encapsulates significant statistical and signal characteristics from time-domain data, enabling deeper insights into the underlying patterns or anomalies within the dataset.

4.5.3 Feature Extraction - Spectrogram

The spectrogram, a visual representation of the spectrum of frequencies in a signal as it varies with time, serves as a critical feature in the analysis of millimeter-wave (mmWave) radar data for skin cancer detection. This method involves converting frequency signals back into the time domain using a series of transformations and visualizations. The goal is to capture the temporal evolution of the frequency-time spectrum, which may indicate unique patterns associated with different skin conditions.

Spectrogram Computation Process

1. **Reading Data:** The process begins by reading the radar data from each CSV file. The data includes both radar frequency responses and corresponding labels indicating the category.
2. **Inverse Fast Fourier Transform (IFFT):** Each row of radar data, representing the frequency domain information, is converted into the time domain using the IFFT. This transformation is crucial as it reconstructs the time-domain signal from the frequency components, providing a basis for further time-frequency analysis.
3. **Spectrogram Generation:** A spectrogram is computed for each row of time-domain data. The MATLAB function `spectrogram` is utilized, which performs a Short-Time Fourier Transform (STFT). The key parameters for STFT include:
 - **Window Size (`windowSize`):** Determines the segment of data to be analyzed in each step. A window size of 32 is used, providing a balance between time and frequency resolution.
 - **Overlap (`overlap`):** Set at 24, this parameter defines how much successive windows overlap each other, ensuring continuity and reducing information loss between segments.
 - **Number of FFT Points (`nfft`):** The number of FFT points, set at 256, dictates the resolution of the resulting frequency analysis.
 - **Sampling Frequency (F_s):** Calculated as 7 GHz which is elaborated further in this section, this parameter is crucial for determining the time scale of the spectrogram.
4. **Visualization:** Each spectrogram is then visualized and saved as an image file, with labels incorporated into the file names for easy identification. The images are stored in a structured directory system based on the file IDs.

Calculation of Parameters from Frequency-Domain Data

Frequency Resolution (Δf)

Frequency resolution is the smallest measurable difference between two distinct frequencies in your dataset, calculated by dividing the total frequency span by the number of frequency points sampled.

Given:

- Frequency Span = 7 GHz (from 62 GHz to 69 GHz)
- Number of Frequency Points = 150

The formula for frequency resolution (Δf) is:

$$\Delta f = \frac{\text{Frequency Span}}{\text{Number of Points}} = \frac{7 \text{ GHz}}{150} \quad (4.18)$$

$$\Delta f = \frac{7,000,000,000 \text{ Hz}}{150} = 46,666,666.67 \text{ Hz} = 46.67 \text{ MHz} \quad (4.19)$$

Total Time Span (T)

The total time span represented by your frequency-domain data in the time domain can be inferred as the reciprocal of the frequency resolution. This corresponds to the maximum period sampled in the time domain.

The formula for total time span (T) is:

$$T = \frac{1}{\Delta f} \quad (4.20)$$

$$T = \frac{1}{46,666,666.67 \text{ Hz}} \approx 21.43 \text{ nanoseconds} \quad (4.21)$$

Sampling Frequency (F_s)

The sampling frequency for the time-domain signal is calculated. Given that the number of time-domain samples (N) is equal to the number of frequency-domain points, the formula for the sampling frequency (F_s) is:

$$F_s = \frac{N}{T} \quad (4.22)$$

$$F_s = \frac{150}{21.43 \times 10^{-9} \text{ seconds}} \approx 7,000,000,000 \text{ Hz} \approx 7 \text{ GHz} \quad (4.23)$$

This is relationship between the frequency-domain resolution and the time-domain sampling, providing key parameters for signal processing and analysis in application.

4.6 Model Evaluation Metrics

In a machine learning model, accurately evaluating a model is paramount, particularly during the early stages of research and ongoing model monitoring. Evaluation metrics not only gauge the performance of a machine learning model but also elucidate its strengths and weaknesses. These metrics are integral in understanding how well a model performs under various conditions and configurations.

4.6.1 Key Performance Metrics

Several metrics derived from the confusion matrix are commonly employed to assess the performance of classification models, especially in text categorization tasks [41]:

The Confusion Matrix

A crucial tool in the assessment of a classification model is the confusion matrix, which provides a visual representation of the model's performance in predicting different outcomes. Typically structured as a 2x2 table, the matrix helps categorize prediction results into true positives, false positives, true negatives, and false negatives. This format facilitates a straightforward evaluation of accuracy in a binary classification context, distinguishing between two classes—positive and negative.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Figure 4.5: Confusion Matrix [41]

Accuracy

This metric represents the ratio of correctly predicted instances to the total predictions made, providing a basic measure of the model's effectiveness.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \quad (4.24)$$

$$\text{Total number of predictions} = \text{number of correct predictions} + \text{number of incorrect predictions} \quad (4.25)$$

Precision

Precision calculates the proportion of positive identifications that were actually correct, crucial for scenarios where the cost of a false positive is high.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (4.26)$$

Recall

Also known as sensitivity, recall measures the model's ability to identify all relevant instances correctly, essential in medical classifications like skin cancer detection.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (4.27)$$

F1-Score

The F1 score is the harmonic mean of precision and recall, providing a balanced measure of a model's accuracy, particularly useful in cases of uneven class distribution.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4.28)$$

4.6.2 Advanced Metrics for Model Performance

Beyond basic metrics, it's often necessary to incorporate additional evaluation criteria depending on the specific requirements and challenges of the task.

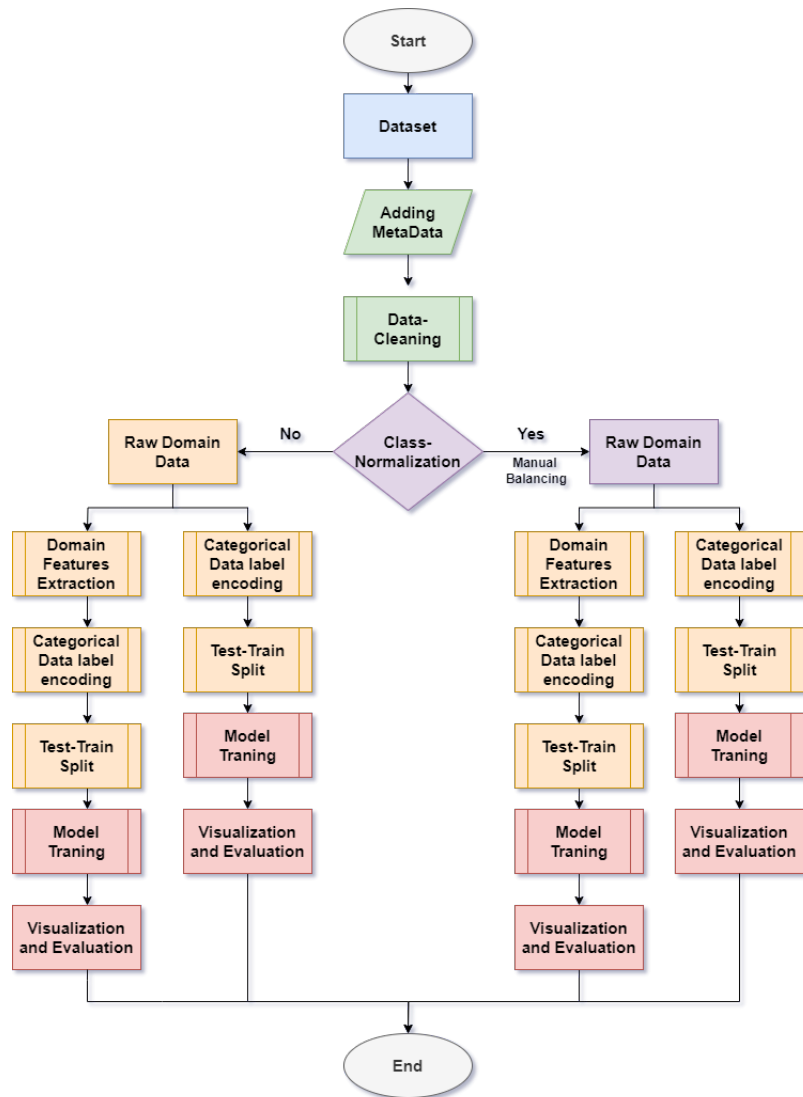


Figure 4.6: Flowchart to visualize the structure of algorithm

For example, **Area Under the Curve (AUC) Score**, representing the area under the ROC curve, is a comprehensive measure that assesses the model's ability to discriminate between the classes across various threshold settings. It is particularly valuable in scenarios with imbalanced datasets.

While accuracy can be a straightforward indicator of model performance, it does not always convey the true effectiveness of a model, especially in imbalanced datasets where it might misleadingly favor the majority class. Incorporating a variety of evaluation metrics such as the confusion matrix, precision, recall, F1 score, and AUC (Area Under the Curve) score is essential for a comprehensive analysis of model performance, particularly in the development and testing of models for classifying skin cancer using millimeter wave data. This multifaceted approach ensures a more robust assessment of the model's effectiveness.

Chapter 5

Experiments and Results

This chapter details the experimental framework utilized in this dissertation, which examines effective classification techniques. It explores various data types and their combinations for classification purposes. It gives deep insight into the experimental procedures carried out through this thesis.

5.1 Experimental Setup

The experiments were performed on the OMNI cluster. The OMNI cluster consists of approximately 450 compute nodes dedicated to processing tasks, 4 login nodes that users can access directly, and several specialized nodes. Each standard compute node is equipped with 64 CPUs and 256 GB of RAM [58].

The computing infrastructure is enhanced with the inclusion of 10 dedicated nodes, each equipped with NVIDIA Tesla V100 GPUs in configurations of either one, two, or four per node. These GPUs support vectorization of double-precision floating point numbers. Apart from the GPUs, the specifications of these nodes, including CPUs and RAM, are the same as those of the regular compute nodes [58].

Python was the chosen language for implementing the experimental scripts and the following frameworks and libraries have been used:

1. **Pandas**: Used for data manipulation and analysis.
2. **NumPy**: Provides support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

3. **Scikit-learn**: Used for machine learning and statistical modeling including classification, regression, clustering, and dimensionality reduction.
4. **SciPy**: Used primarily for signal processing.
5. **TensorFlow**: An end-to-end open source platform for machine learning to easily build and deploy ML powered applications.
6. **Matplotlib**: A plotting library for creating static, interactive, and animated visualizations in Python.
7. **Seaborn**: A Python data visualization library based on matplotlib that provides a high-level interface for drawing attractive and informative statistical graphics.
8. **Intel Extension for Scikit-learn**: Used to accelerate scikit-learn by patching it with Intel's optimizations.

These libraries cover a wide range of uses from basic data handling (Pandas, NumPy) to advanced machine learning and deep learning (Scikit-learn, TensorFlow). The visualizations are managed by Matplotlib, and Seaborn enhancing the ability to present data insights effectively.

the datasets were divided into two subsets: one for training the model and the other for testing it. The data should be segmented such that 20% is allocated for model evaluation, while the remaining 80% is utilized for the model's training phase.

In the following table 5.1 for mmWave data, the hyper-parameter settings for each machine learning model are summarized:

Model Name	Hyperparameters
Logistic Regression	max_iter=10000
Ridge Classifier	Default parameters (no modifications)
K-Nearest Neighbors	Default parameters (no modifications)
Support Vector Classifier	Default parameters (no modifications)
Decision Tree Classifier	Default parameters (no modifications)
Random Forest Classifier	n_estimators=100
Gradient Boosting Classifier	n_estimators=100
AdaBoost Classifier	n_estimators=100
Extra Trees Classifier	n_estimators=100
Gaussian Naive Bayes	Default parameters (no modifications)
Multilayer Perceptron	max_iter=1000

Table 5.1: Hyper-tuning of ML Models

In the following table 5.2 for mmWave data, summarizing the hyperparameter settings for the deep learning model is given:

Component	Hyperparameters
Sequential Model	
Dense Layer 1	128 neurons, activation='relu', input shape from X_train_preprocessed
Dropout Layer 1	rate=0.5
Dense Layer 2	64 neurons, activation='relu'
Dropout Layer 2	rate=0.5
Output Dense Layer	Neurons based on class count, activation depending on class count (binary: 'sigmoid', multi-class: 'softmax')
Model Compilation	
Optimizer	'adam'
Loss	Conditional on class count (binary: 'binary_crossentropy', multi-class: 'sparse_categorical_crossentropy')
Metrics	'accuracy'
Training the Model	
Epochs	50
Batch size	32
Validation Split	0.2
Verbosity Level	2

Table 5.2: Hyperparameters of the Deep Learning model

In Table 5.3 the settings for hyperparameters utilized in the construction of the deep learning model for Spectrograms are summarized:

Component	Hyperparameters or Settings
Model Architecture	
Conv2D (1st layer)	32 filters, kernel size (3, 3), activation='relu', input shape=(1167, 875, 3)
MaxPooling2D (1st layer)	Pool size (2, 2)
Conv2D (2nd layer)	64 filters, kernel size (3, 3), activation='relu'
MaxPooling2D (2nd layer)	Pool size (2, 2)
Conv2D (3rd layer)	128 filters, kernel size (3, 3), activation='relu'
MaxPooling2D (3rd layer)	Pool size (2, 2)
Dense	512 neurons, activation='relu'
Dropout	rate=0.5
Output Dense Layer	3 neurons (corresponding to the number of classes), activation='softmax'
Model Compilation	
Optimizer	'adam'
Loss	'categorical_crossentropy'
Metrics	'accuracy'
Data Generators	
ImageDataGenerator	Rescaling factor 1./255, Validation split 20%
Training Data Generator	Target size: (1167, 875), Batch size: 32, Class mode: 'categorical', Subset: 'training'
Validation Data Generator	Target size: (1167, 875), Batch size: 32, Class mode: 'categorical', Subset: 'validation'
Training the Model	
Epochs	10, 50
Callbacks	EarlyStopping: monitor='val_loss', patience=3, restore_best_weights=True

Table 5.3: Hyperparameters of Spectrogram Deep Learning model

5.2 Raw mmWave Data Classification

This classification is the analysis of raw mmWave data with all 5 classes. This classification further divided into two subsection for frequency and time domain datasets.

5.2.1 Frequency Domain Data Classification

A detailed analysis and model building process have been performed for the classification of frequency domain mmWave radar data. The dataset consists of frequency points from samples taken from individuals, labelled with 'bumpy nevus', 'nevus', 'scar', 'nevus/birth-mark', 'red nevus'. It is further divided into two parts. Firstly, the raw frequency data have been considered as a feature and further processed. In the second part features like Magnitude, Phase, and Spectral Centroid etc. were extracted and further analysis was performed on the extracted dataset.

5.2.1.1 Raw Frequency as a Feature

mmWave radar data from 44 individuals, each providing 50 samples of 400x150 frequency points, were used for skin cancer classification. This led to a dataset of 880,000 instances, each labeled with one of five skin conditions.

In evaluating the classification of raw mmWave radar data, the experiments yielded varying results across multiple machine learning models. The results of the analysis are given in table 5.4.

K-Nearest Neighbors and Extra Trees Classifier showed performance with accuracies of 89.76% and 90.53%, respectively. Their high precision, recall, and F1-scores close to 90% suggest excellent model fit and robustness in dealing with imbalanced classes. Their AUC-ROC values approaching 1.00 further show exceptional ability to differentiate between the classes.

Random Forest Classifier also showed robust performance with an accuracy of 88.11%, supporting the effectiveness of ensemble methods in handling complex datasets generated from mmWave radar frequencies.

Support Vector Classifier and Multilayer Perceptron offered moderate accuracy levels at 55.37% and 55.00%, respectively. Although not as high as the top performers, their precision and F1-scores above 45%.

Decision Tree Classifier achieved a relatively better accuracy of 70.22%, making it a viable option for scenarios where interpretability is crucial, despite lower performance compared

Model	Accuracy	Precision (Weighted Avg.)	Recall (Weighted Avg.)	F1-Score (Weighted Avg.)	AUC -ROC
Logistic Regression	0.5018	0.42	0.50	0.34	0.59- 0.69
Ridge Classifier	0.5032	0.42	0.50	0.34	0.59- 0.69
K-Nearest Neighbors	0.8976	0.90	0.90	0.90	0.96- 0.99
Support Vector Classifier	0.5537	0.66	0.55	0.45	0.69- 0.83
Decision Tree Classifier	0.7022	0.70	0.70	0.70	0.75- 0.81
Random Forest Classifier	0.8811	0.90	0.88	0.88	0.97- 1.00
Gradient Boosting Classifier	0.5034	0.60	0.50	0.34	0.61- 0.71
AdaBoost Classifier	0.5016	0.25	0.50	0.34	0.52- 0.65
Extra Trees Classifier	0.9053	0.92	0.91	0.90	0.98- 1.00
Gaussian Naive Bayes	0.1832	0.34	0.18	0.12	0.50- 0.54
Multilayer Perceptron	0.55	0.52	0.55	0.49	0.68- 0.84

Table 5.4: Performance of ML Models on Raw Frequency Data

to ensemble methods.

Logistic Regression, Ridge Classifier, Gradient Boosting Classifier, and AdaBoost Classifier hovered around 50% accuracy, only slightly better than random guessing. These models appeared to struggle with the dataset’s complexity or distribution, suggesting that linear models might be insufficient for this type of data.

Gaussian Naive Bayes had the lowest accuracy at 18.32%, reflecting the challenges this model faces with data featuring interdependent features, typical of frequency-based datasets.

Area Under the Curve (AUC-ROC):

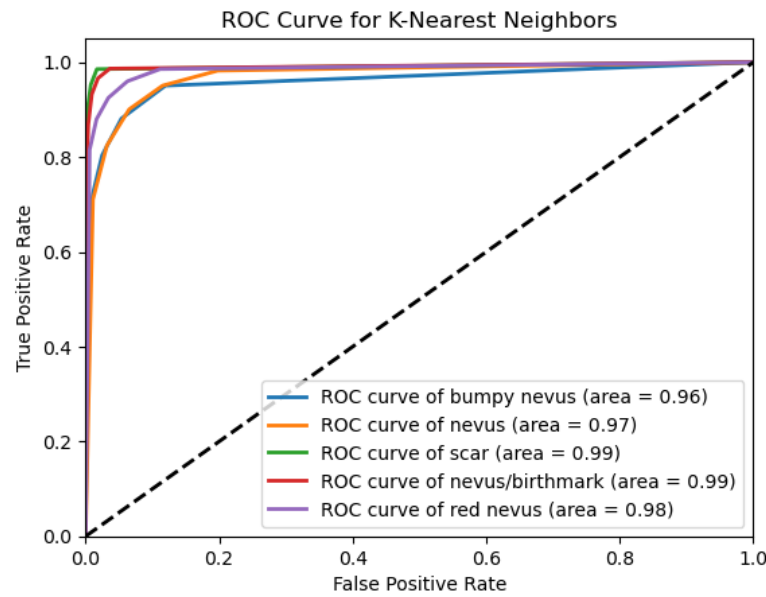


Figure 5.1: ROC curve for K-Nearest Neighbours

The AUC-ROC values provide additional insights into the model's capability to distinguish between the different classes. Models like K-Nearest Neighbors as can be seen in figure 5.1, Random Forest, and Extra Trees achieved the highest AUC-ROC scores, signifying their strong discriminatory power. In contrast, models like Gaussian Naive Bayes exhibited much lower AUC-ROC values, highlighting their limitations.

Confusion Matrix:

The confusion matrices across various models reveal significant disparities in performance for mmWave radar data. Models like AdaBoost, Logistic Regression, and Gradient Boosting Classifier predominantly misclassified different conditions as 'nevus', indicating a strong bias. The Decision Tree and Multilayer Perceptron showed moderate accuracy with a better spread but notable misclassifications.

In contrast, the Extra Trees as figure 5.2 and Random Forest classifiers showed robust performance with high accuracy and true positive rates, especially for 'nevus' and 'red nevus'. The Support Vector Classifier and Ridge Classifier, however, struggled with accurate predictions across various classes, particularly 'scar' and 'nevus/birthmark'.

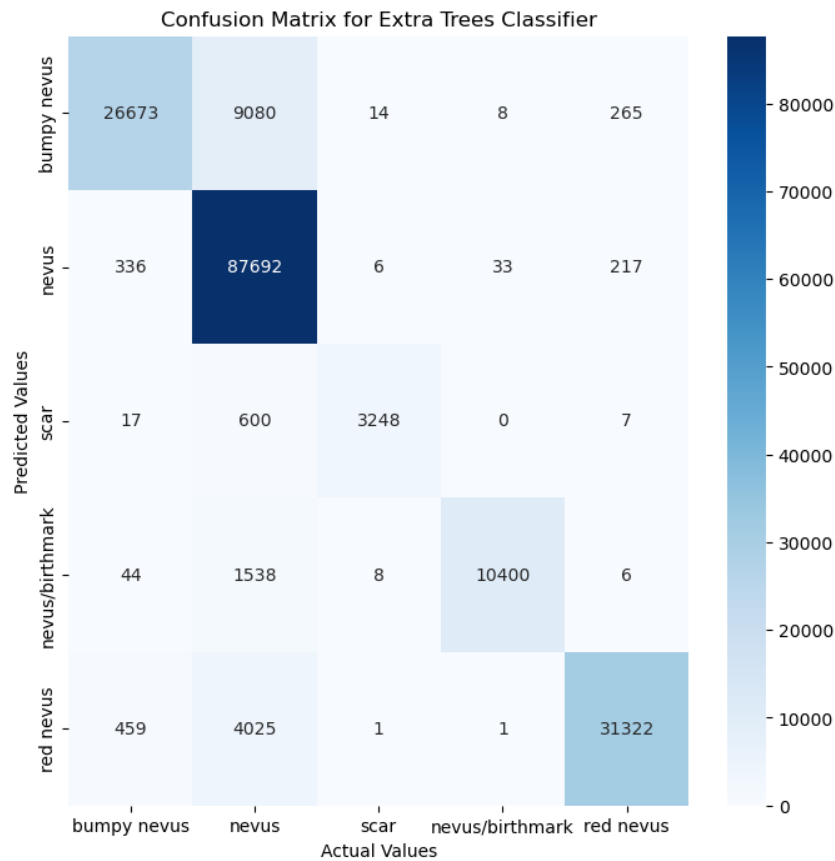


Figure 5.2: Confusion Matrix for Extra Tree Classifier

Deep learning Model:

The deep learning model’s training and validation results reveal a consistent challenge in achieving significant predictive accuracy, stabilizing around 50% across 50 epochs. The consistent loss values and the stagnant accuracy figures across training and validation phases might imply that the data itself does not contain enough discriminative information for more accurate classification using the provided features and model configuration.

5.2.1.2 Extracted Features: Frequency Domain

This data set has been generated by extracting features like Magnitude, Phase, and Spectral Centroid, etc. from the raw frequency data making it to a dataset of 880000 instances each with 15 frequency features labeled with one of five skin conditions.

Model	Accuracy	Precision (Weighted Avg.)	Recall (Weighted Avg.)	F1-Score (Weighted Avg.)	AUC -ROC
Logistic Regression	0.5020	0.25	0.50	0.34	0.51- 0.54
Ridge Classifier	0.5020	0.25	0.50	0.34	0.59- 0.69
K-Nearest Neighbors	0.7207	0.72	0.72	0.72	0.84- 0.92
Support Vector Classifier	0.5052	0.62	0.50	0.34	0.56- 0.70
Decision Tree Classifier	0.5873	0.59	0.59	0.59	0.67- 0.72
Random Forest Classifier	0.7499	0.79	0.75	0.74	0.88- 0.96
Gradient Boosting Classifier	0.5029	0.58	0.50	0.34	0.56- 0.63
AdaBoost Classifier	0.5020	0.32	0.50	0.34	0.53- 0.59
Extra Trees Classifier	0.7579	0.79	0.76	0.75	0.88- 0.96
Gaussian Naive Bayes	0.5002	0.26	0.50	0.34	0.51- 0.56
Multilayer Perceptron	0.5025	0.45	0.50	0.34	0.57- 0.66

Table 5.5: Performance of ML Models on Extracted Frequency Domain Data

In evaluating the classification of extracted frequency features, the experiments yielded varying results across multiple machine learning models. The results of the analysis are given in table 5.5.

High-performing models like the Random Forest and Extra Trees classifiers showcased robust accuracy, precision, recall, and F1-scores, all exceeding 74%. These tree-based models effectively leverage the structured frequency features, enabling them to discern patterns that predict skin conditions with higher reliability. Their high AUC-ROC values, nearing 0.96, further emphasize their proficiency in distinguishing between the classes effectively.

Conversely, models such as Logistic Regression, AdaBoost Classifier, and Gaussian Naive Bayes performed poorly, barely surpassing a 50% accuracy mark, which suggests challenges in capturing the complexities within the frequency features. These models, along with the Ridge and Gradient Boosting classifiers, exhibit lower precision and F1-scores, indicating a potential mismatch in model assumptions and the underlying data distri-

bution. The consistently low performance across several models highlights the need for more complex or differently tailored approaches to improve prediction accuracy for skin condition diagnosis using extracted frequency features.

Area Under the Curve (AUC-ROC):

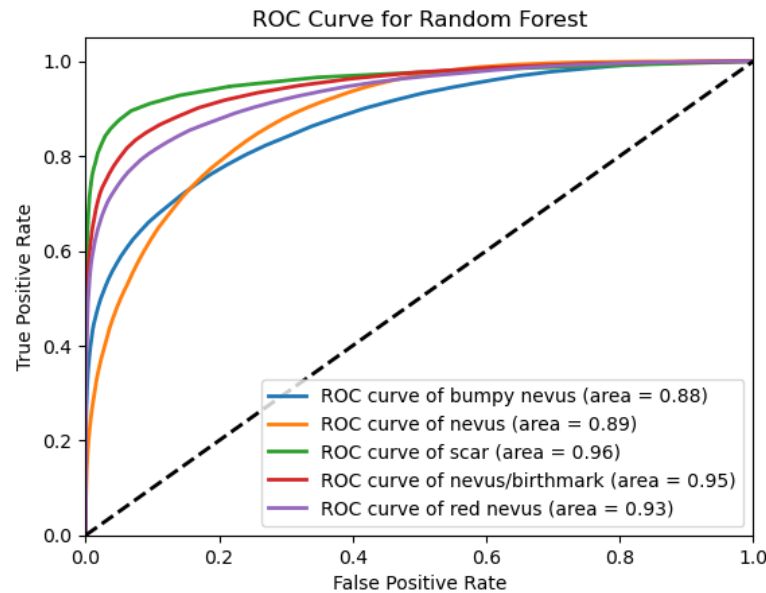


Figure 5.3: ROC curve for Random Forest Classifier

High AUC-ROC values between 0.88 to 0.96 for Random Forest in figure 5.3 and Extra Trees classifiers suggest excellent model sensitivity and specificity. In contrast, Logistic Regression, AdaBoost, and Gaussian Naive Bayes have AUC-ROC scores around 0.51 to 0.59, indicating poor discriminative ability. Models with mid-range AUC-ROC values, like the K-Nearest Neighbors and Decision Tree, show moderate classification capabilities.

Confusion Matrix:

Logistic Regression, Support Vector Classifier, and AdaBoost show a strong bias towards predicting the most common class, 'nevus', but fail almost entirely to identify other conditions. This indicates that the models may not be adequately complex or trained thoroughly enough to distinguish effectively between classes, which could result in under-performance.

In contrast, models like K-Nearest Neighbors, Decision Tree, Random Forest, and Extra Trees display better distributed predictions across different conditions. Particularly, Random Forest in figure 5.4 and Extra Trees classifiers demonstrate superior performance in not only capturing the majority class but also reasonably identifying less frequent conditions, indicating a more balanced and effective learning from the feature set.

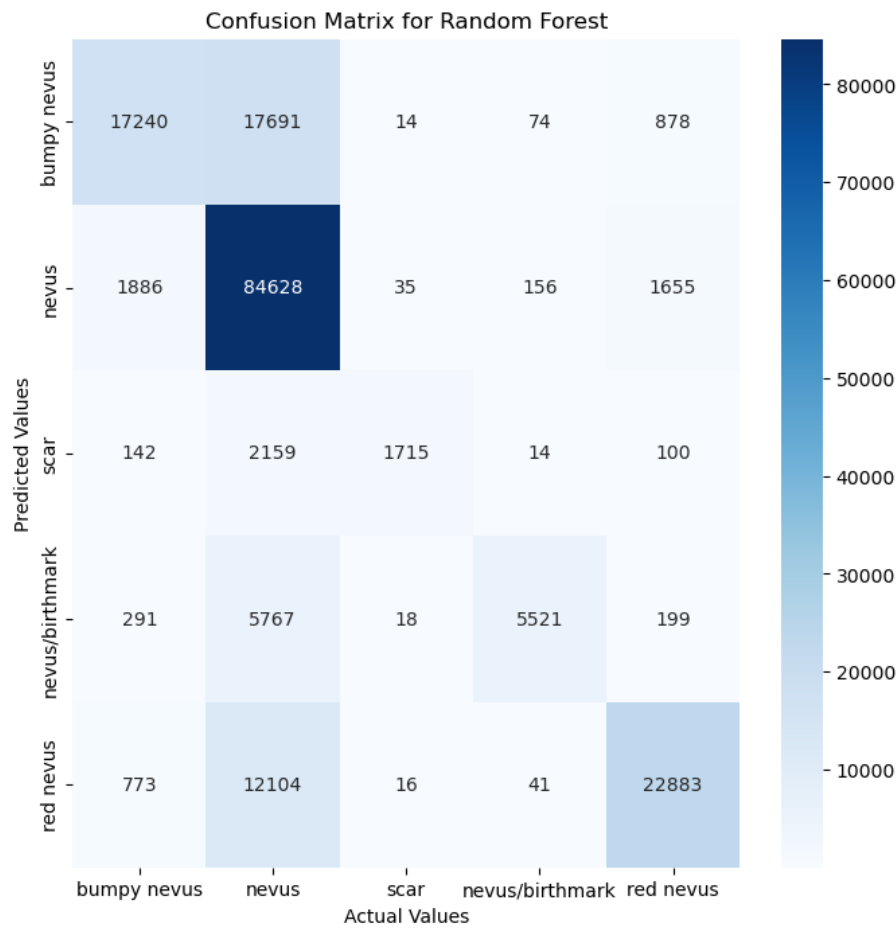


Figure 5.4: Confusion Matrix for Random Forest Classifier

Deep learning Model:

The neural network model’s training and validation results demonstrate a consistent under-performance, with both accuracy and loss metrics plateauing at levels indicative of non-optimal learning. The accuracy hovers around 49.97% across both training and validation phases, showing minimal improvement and failing to substantially differentiate between the classes. The model’s loss decreases only slightly over 50 epochs, further signifying the lack of effective learning and adaptation to the dataset’s features.

5.2.2 Time Domain Data Classification

Time domain data generated using Inverse Fast Fourier Transformation(IFFT) of Raw frequency mmWave radar data. The dataset labelled with 'bumpy nevus', 'nevus', 'scar', 'nevus/birth-mark', 'red nevus'. It is further divided into two parts.

5.2.2.1 Raw Time Domain as a Feature

This dataset consists of 880,000 instances with 150 sample points, each labeled with one of five skin conditions.

In evaluating the classification of raw time domain data, the experiments yielded varying results across multiple machine learning models. The results of the analysis are given in table 5.6.

Model	Accuracy	Precision (Weighted Avg.)	Recall (Weighted Avg.)	F1-Score (Weighted Avg.)	AUC -ROC
Logistic Regression	0.5020	0.75	0.50	0.34	0.50- 0.51
Ridge Classifier	0.5020	0.75	0.50	0.34	0.51- 0.52
K-Nearest Neighbors	0.8673	0.87	0.87	0.87	0.94- 0.98
Support Vector Classifier	0.5367	0.66	0.54	0.41	0.67- 0.79
Decision Tree Classifier	0.6642	0.66	0.66	0.66	0.73- 0.77
Random Forest Classifier	0.7691	0.83	0.77	0.75	0.94- 0.99
Gradient Boosting Classifier	0.5264	0.68	0.53	0.39	0.62- 0.78
AdaBoost Classifier	0.5043	0.49	0.50	0.34	0.53- 0.71
Extra Trees Classifier	0.7486	0.82	0.75	0.73	0.94- 0.99
Gaussian Naive Bayes	0.2244	0.37	0.22	0.20	0.51- 0.58
Multilayer Perceptron	0.5784	0.58	0.58	0.51	0.70- 0.89

Table 5.6: Performance of ML Models on Raw Time Domain Data

The results show a considerable variation in model performance, with Logistic Regression and Ridge Classifier both yielding a mere 50.20% accuracy, indicating potentially high

bias in these simpler models or insufficient model complexity to capture the nuances in the data.

On the other hand, models such as K-Nearest Neighbors and Random Forest demonstrate superior performance with accuracies of 86.73% and 76.91% respectively. These models benefit from their ability to handle non-linear relationships and complex pattern recognition in the time-domain features, which are likely abundant in the dataset. Furthermore, the satisfactory performance of the Decision Tree and Extra Trees classifiers implies that ensemble and non-parametric methods are well-suited for analyzing this kind of data.

Area Under the Curve (AUC-ROC):

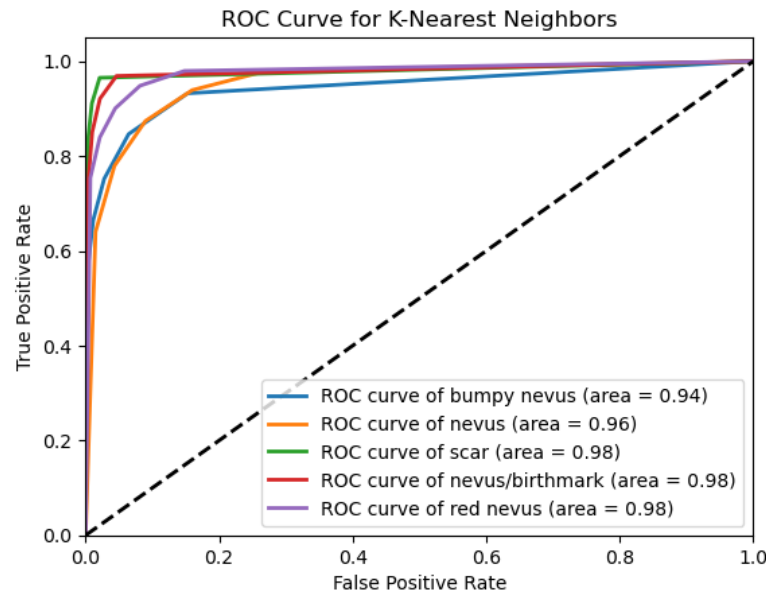


Figure 5.5: ROC curve for K-Nearest Neighbors

Analyzing the Area Under the Curve (AUC-ROC) provides further insights into model efficacy. The AUC-ROC values span from as low as 0.51 for Logistic Regression to as high as 0.94-0.98 for K-Nearest Neighbors and Random Forest, indicating excellent performance in distinguishing between different skin conditions. This metric highlights the robustness of tree-based ensemble methods in handling complex patterns within time-domain features.

Confusion Matrix:

Logistic Regression, showing high diagonal counts for "nevus" and "red nevus" but failing to identify other classes, reflects high specificity but extremely low sensitivity for other classes. K-Nearest Neighbors and Support Vector Classifier show improved recognition across multiple classes, particularly "nevus," but struggle with less frequent classes like

"bumpy nevus." Decision Tree and Random Forest exhibit better overall distribution in correct predictions across all classes, illustrating their capability to manage class imbalance more effectively. Gradient Boosting, Gaussian Naive Bayes and AdaBoost also display significant improvements in multi-class identification, though some classes still show lower detection rates. Extra Trees model as in figure 5.6 demonstrate a robust performance for "nevus" but vary in detecting other categories, with Naive Bayes showing substantial misclassifications across less frequent conditions. Multilayer Perceptron shows balanced performance, though it struggles with the scar and nevus/birthmark categories, suggesting potential overfitting to dominant classes.

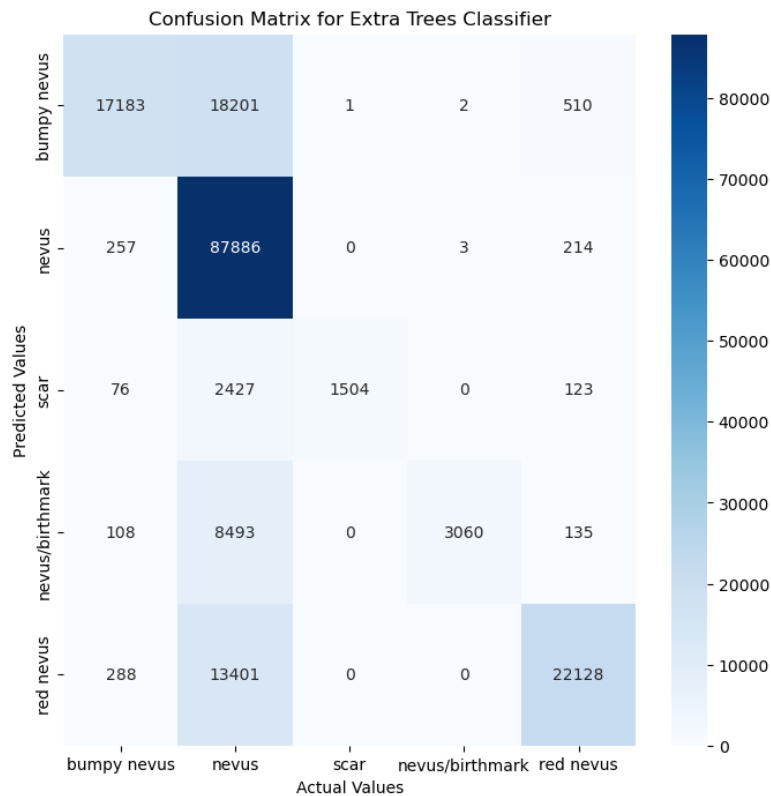


Figure 5.6: Confusion Matrix for Extra Tree Classifier

Deep learning Model:

The output of the deep learning model indicates a stagnation in performance improvements across the 50 training epochs, as evidenced by the minimal changes in accuracy, which hovered just above the 50% mark, and the consistent loss values. The consistently low validation accuracy near 0.50 suggests that the model struggled to generalize beyond the training data, performing only slightly better than random chance.

5.2.2.2 Extracted Features: Time Domain

This data set has been generated by extracting features like Signal Magnitude Area (SMA), Energy, and Peak-to-Peak Amplitude, etc. from the raw time domain data making it to a dataset of 880000 instances each with 14 time domain features including label with one of five skin conditions.

In evaluating the classification of extracted time domain data, the experiments yielded varying results across multiple machine learning models. The results of the analysis are given in table 5.7.

Model	Accuracy	Precision (Weighted Avg.)	Recall (Weighted Avg.)	F1-Score (Weighted Avg.)	AUC -ROC
Logistic Regression	0.5021	0.41	0.50	0.34	0.52- 0.56
Ridge Classifier	0.5020	0.25	0.50	0.34	0.52- 0.56
K-Nearest Neighbors	0.6999	0.70	0.70	0.70	0.83- 0.91
Support Vector Classifier	0.5031	0.54	0.50	0.34	0.56- 0.69
Decision Tree Classifier	0.6102	0.61	0.61	0.61	0.69- 0.74
Random Forest Classifier	0.7556	0.78	0.76	0.74	0.88- 0.96
Gradient Boosting Classifier	0.5038	0.58	0.50	0.34	0.56- 0.64
AdaBoost Classifier	0.5021	0.41	0.50	0.34	0.52- 0.60
Extra Trees Classifier	0.7644	0.79	0.76	0.75	0.89- 0.96
Gaussian Naive Bayes	0.4367	0.30	0.44	0.35	0.51- 0.56
Multilayer Perceptron	0.5031	0.47	0.50	0.34	0.56- 0.66

Table 5.7: Performance of ML Models on Extracted Features of Time Domain Data

Random Forest and Extra Trees Classifier significantly outperform simpler linear models such as Logistic Regression and Ridge Classifier. These ensemble methods, with accuracy scores of 0.7556 and 0.7644 respectively, leverage the strength of multiple decision trees to handle the non-linearity and complexity inherent in the time-domain features. In contrast, linear models struggle to capture these complexities, reflected in their near-random accuracy scores of around 0.50.

Furthermore, the Precision, Recall, and F1-Score metrics reinforce the superiority of ensemble methods in this context, with both Random Forest and Extra Trees achieving scores above 0.74 across these metrics, indicating robustness in both prediction relevance and result retrieval across class labels.

Area Under the Curve (AUC-ROC):

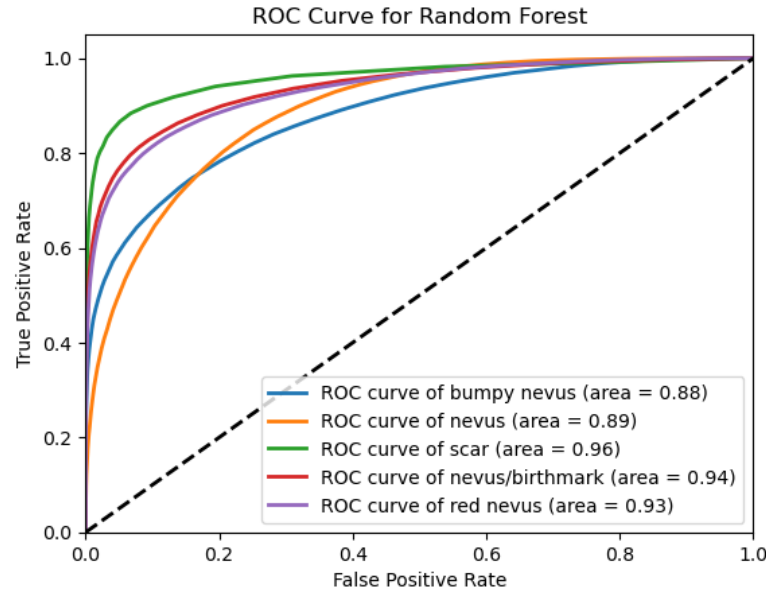


Figure 5.7: ROC curve for Random Forest Classifier

Logistic regression and similar basic models scored just above 0.50, indicating poor discriminative ability, ensemble methods such as Random Forest as in figure 5.7 and Extra Trees Classifier reached AUC-ROC values close to 0.96, demonstrating high true positive rates with fewer false positives. This enhanced performance underscores the effectiveness of ensemble approaches in handling complex patterns within time-domain feature datasets, ultimately leading to better generalization and robust prediction.

Confusion Matrix:

Logistic Regression and Support Vector Classifier show limited ability to differentiate between classes, mostly predicting the majority class, with high false negative rates for most conditions except 'nevus.' K-Nearest Neighbors and Decision Tree classifiers exhibit better performance, with more balanced true positive rates, but still significant confusion among less frequent classes. Random Forest show in figure 5.8 improved detection across most classes yet struggles with specific categories like 'nevus/birthmark' and 'red nevus'. The AdaBoost, Gradient Boosting, and Gaussian Naive Bayes classifier demonstrate moderate effectiveness leading to higher misclassification rates.

In contrast, Extra Trees classifiers show a better generalization ability across multiple classes, although their performance varies widely with some skin conditions still being predominantly misclassified as 'nevus.' Multilayer Perceptron achieves decent classification accuracies but shows a tendency to misclassify the 'scar' and 'nevus/birthmark' classes.

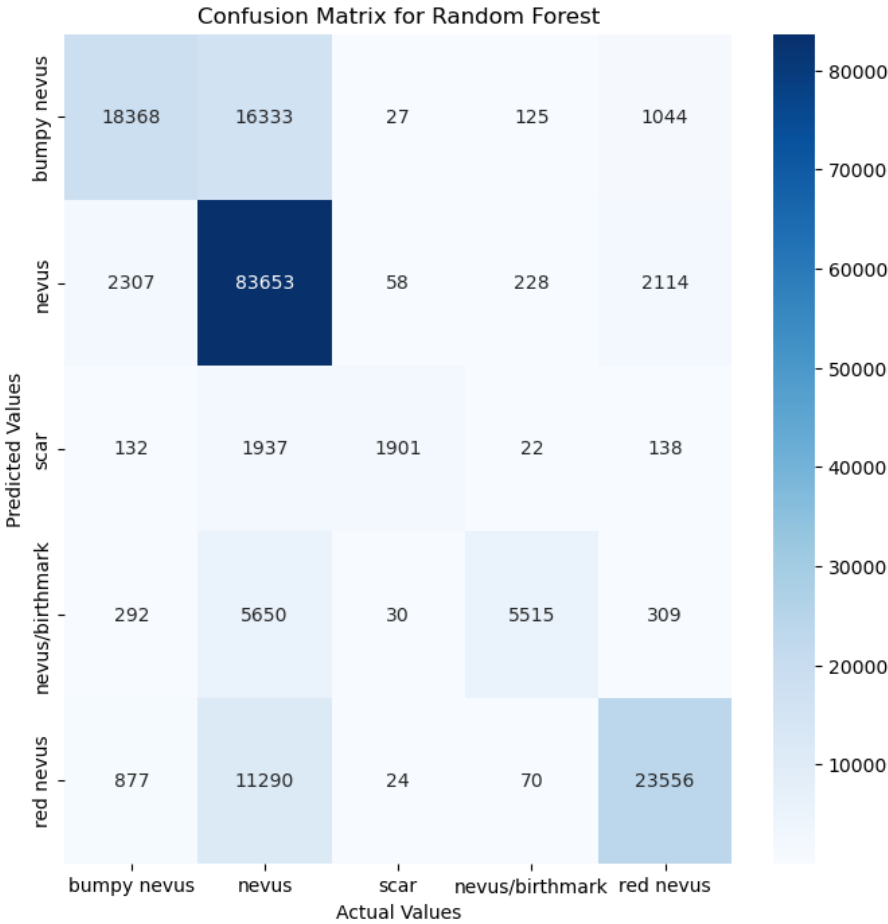


Figure 5.8: Confusion Matrix for Random Forest Classifier

Deep learning Model:

The output of the deep learning model indicates a stagnation in performance improvements across the 50 training epochs, as evidenced by the minimal changes in accuracy, which hovered just above the 50% mark, and the consistent loss values. The consistently low validation accuracy near 0.50 suggests that the model struggled to generalize beyond the training data, performing only slightly better than random chance.

5.3 Balanced mmWave Data Classification

In earlier experiments, 22 samples were classified as 'Nevus' which makes data imbalanced. To prevent data imbalance, 9 people were selected randomly from meta-data for each class 'Nevus', 'Bumpy Nevus', and 'Scar'. 'Red Nevus' and 'Nevus/Birthmark' classes were dropped due to significantly low samples which makes this classification the analysis of balanced raw mmWave data with 3 classes. This classification further divided into two subsection for frequency and time domain datasets.

5.3.1 Frequency Domain Data Classification

A detailed analysis and model building process have been performed for the classification of frequency domain mmWave radar data. The dataset consists of frequency points from samples taken from individuals, labelled with 'bumpy nevus', 'nevus', 'scar'. It is further divided into two parts. Firstly, the raw frequency data have been considered as a feature and further processed. In the second part features like Magnitude, Phase, and Spectral Centroid etc. were extracted and further analysis was performed on the extracted dataset.

5.3.1.1 Raw Frequency as a Feature

This is mmWave radar data from 27 individuals, each providing 50 samples of 400x150 frequency points, were used for skin cancer classification. This led to a dataset of 540000 instances, each labeled with one of three skin conditions.

In evaluating the classification of raw mmWave radar data, the experiments yielded varying results across multiple machine learning models. The results of the analysis are given in table 5.8.

K-Nearest Neighbors, Random Forest, and Extra Trees classifiers exhibit superior performance, achieving accuracies above 90%, with equally high precision and recall metrics, indicating their effectiveness in handling balanced datasets. Conversely, simpler models such as Logistic Regression, AdaBoost, and Gaussian Naive Bayes show notably lower effectiveness, with accuracies not exceeding 40%.

Decision Trees and Support Vector Classifiers demonstrate moderate success, with accuracies around 57% to 74%. These results suggest that while these models are capable of capturing more complex patterns than the simpler models, they still fall short of the performance achieved by ensemble methods and KNN.

Model	Accuracy	Precision (Weighted Avg.)	Recall (Weighted Avg.)	F1-Score (Weighted Avg.)	AUC -ROC
Logistic Regression	0.4003	0.40	0.40	0.39	0.56- 0.62
Ridge Classifier	0.4126	0.41	0.41	0.41	0.57- 0.63
K-Nearest Neighbors	0.9149	0.92	0.91	0.91	0.97- 0.98
Support Vector Classifier	0.5768	0.58	0.58	0.57	0.72- 0.77
Decision Tree Classifier	0.7408	0.74	0.74	0.74	0.79- 0.82
Random Forest Classifier	0.9238	0.92	0.92	0.92	0.98- 0.99
Gradient Boosting Classifier	0.4326	0.43	0.43	0.43	0.61- 0.62
AdaBoost Classifier	0.3801	0.38	0.38	0.38	0.55- 0.56
Extra Trees Classifier	0.9364	0.94	0.94	0.94	0.99
Gaussian Naive Bayes	0.3417	0.34	0.34	0.30	0.50- 0.54
Multilayer Perceptron	0.5404	0.54	0.54	0.54	0.69- 0.77

Table 5.8: Performance of ML Models on Balanced Raw Frequency Data

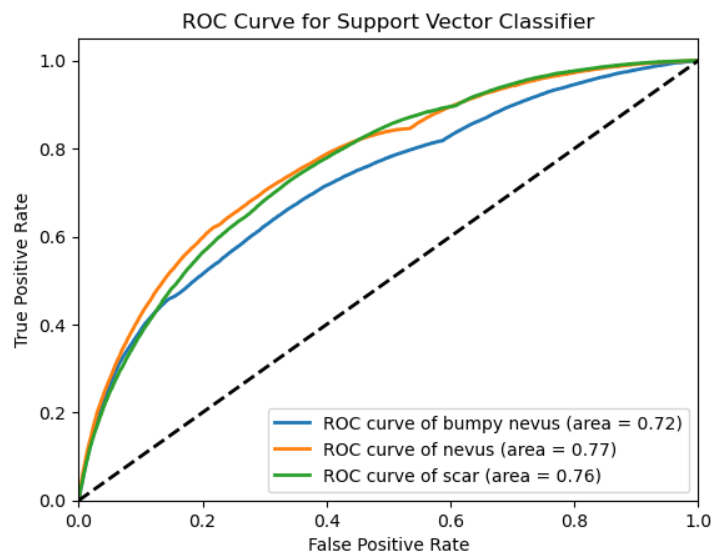
Area Under the Curve (AUC-ROC):

Figure 5.9: ROC curve for Support Vector Classifier

The Area Under the Curve (AUC-ROC) metrics further highlight the discriminatory power of the evaluated models. High AUC-ROC values close to 1.0, observed in Random Forest and Extra Trees classifiers, improved in Support vector classifier 0.72-0.77 as in figure 5.9 indicate an excellent ability to distinguish between the three skin conditions with minimal false positive rates. These values affirm the robustness of these models in maintaining performance consistency across varying threshold settings, making them highly reliable.

In contrast, models with AUC-ROC scores closer to 0.5, such as Gaussian Naive Bayes, reflect a performance barely better than random guessing, emphasizing their limited utility in medical diagnostics where precision is critical.

Confusion Matrix:

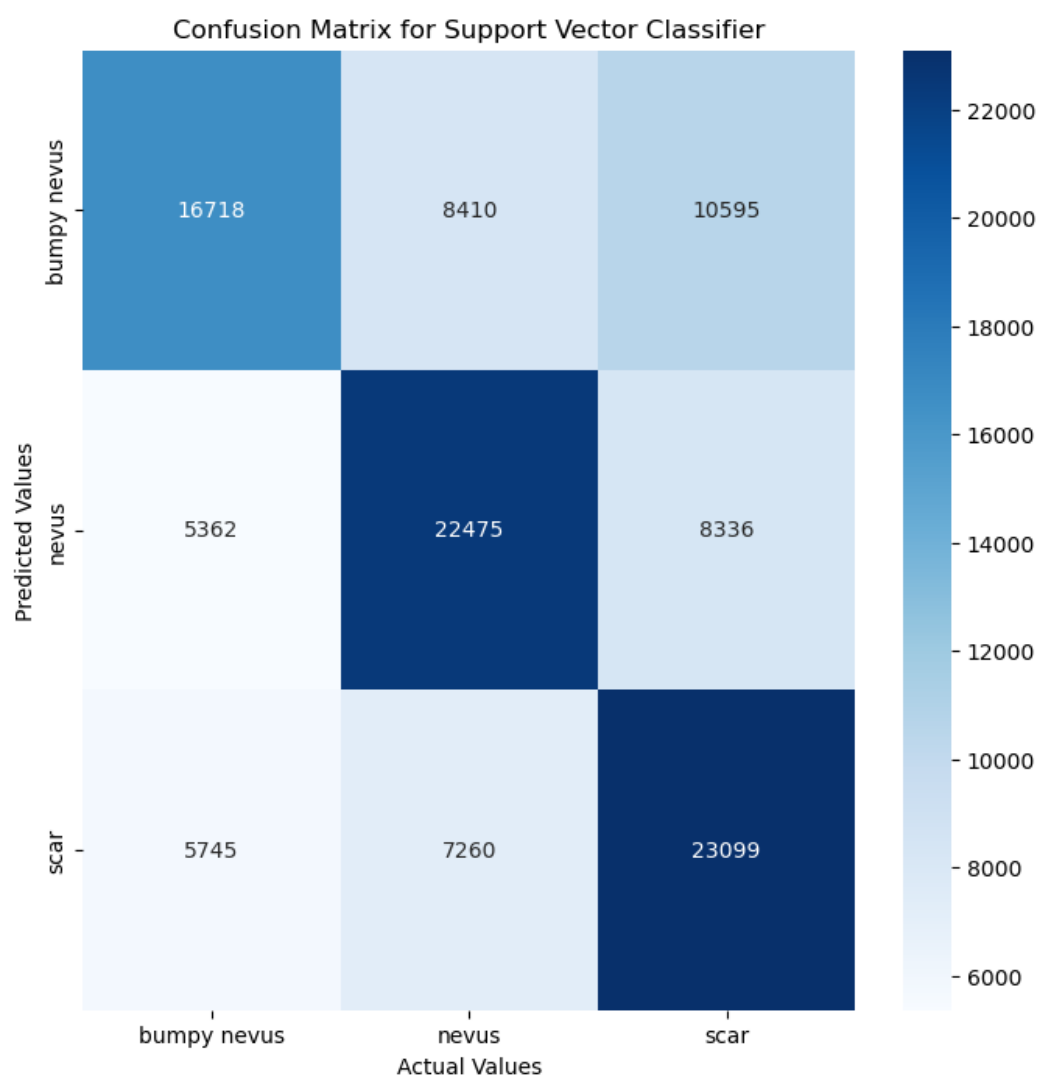


Figure 5.10: Confusion Matrix for Support Vector Classifier

The Logistic Regression and Ridge Classifier both display substantial confusion between classes, indicating difficulty in distinguishing between the three classes effectively. In contrast, K-Nearest Neighbors and Random Forest show significantly better performance, with high correct classifications, especially for the Nevus class, reflecting their robustness in handling this dataset.

Support Vector Classifier as in figure 5.10, Decision Tree, and Gradient Boosting Classifier present intermediate results with notable errors, especially in misclassifying Nevus as Scar. AdaBoost and Extra Trees also show improved performance but with some misclassifications. This analysis reveals the varying abilities of classifiers to handle complex patterns within skin condition data, suggesting that ensemble methods like Random Forest and Extra Trees tend to outperform simpler linear models in this specific context.

Deep learning Model:

The output of the deep learning model indicates a stagnation in performance improvements across the 50 training epochs, as evidenced by the minimal changes in accuracy, which hovered just above the 36% mark, and the consistent loss values. The consistently low validation accuracy near 0.36 suggests that the model struggled to generalize beyond the training data, performing only slightly better than random chance.

5.3.1.2 Extracted Features: Frequency Domain

This dataset has been generated by extracting features like Magnitude, Phase, and Spectral Centroid, etc. from the balanced raw frequency data making it to a dataset of 540000 instances each with 15 frequency features labeled with one of three skin conditions.

The analysis of balanced extracted frequency domain data classification using various machine learning models reveals a significant disparity in performance metrics as shown in table 5.9. Logistic Regression, Ridge Classifier, and Gaussian Naive Bayes exhibit notably low accuracy, precision, recall, and F1-scores, barely surpassing the 0.34 mark, indicating poor model fit or inadequate handling of the frequency domain features for these methods.

In contrast, machine learning models like K-Nearest Neighbors, Random Forest, and Extra Trees Classifier have shown superior performance, with accuracies exceeding 0.76, and also high precision and recall metrics. This indicates that these models effectively leverage patterns in frequency-based features for classification purposes.

Decision Tree, Support Vector Classifier, and Multilayer Perceptron show intermediate outcomes with accuracies ranging from 0.41 to 0.63. These results indicate moderate effectiveness in utilizing extracted features, with the Decision Tree performing relatively

Model	Accuracy	Precision (Weighted Avg.)	Recall (Weighted Avg.)	F1-Score (Weighted Avg.)	AUC -ROC
Logistic Regression	0.3439	0.34	0.34	0.33	0.51- 0.52
Ridge Classifier	0.3434	0.34	0.34	0.33	0.51- 0.52
K-Nearest Neighbors	0.7604	0.76	0.76	0.76	0.88- 0.91
Support Vector Classifier	0.4119	0.41	0.41	0.41	0.59- 0.60
Decision Tree Classifier	0.6310	0.63	0.63	0.63	0.71- 0.74
Random Forest Classifier	0.7913	0.79	0.79	0.79	0.92- 0.94
Gradient Boosting Classifier	0.3899	0.39	0.39	0.39	0.56- 0.58
AdaBoost Classifier	0.3666	0.37	0.37	0.36	0.53- 0.55
Extra Trees Classifier	0.8006	0.80	0.80	0.80	0.92- 0.95
Gaussian Naive Bayes	0.3429	0.35	0.34	0.31	0.51- 0.52
Multilayer Perceptron	0.4071	0.41	0.41	0.41	0.58- 0.60

Table 5.9: Performance of ML Models on Balanced Extracted Frequency Domain Data

better among them. Gradient Boosting and AdaBoost Classifier, while generally robust learning algorithms display limited effectiveness in this setting, as evidenced by their lower performance metrics around the 0.37 to 0.39 mark for accuracy, precision, recall, and F1-score.

Area Under the Curve (AUC-ROC):

The Area Under the Curve (AUC-ROC) metric across the models provides insights into their ability to discriminate between the classes effectively. The highest AUC-ROC values observed for K-Nearest Neighbors as in figure 5.11, Random Forest, and Extra Trees Classifier, ranging from 0.88 to 0.95, affirm their robustness in classification, particularly in distinguishing between the three skin conditions accurately. Models with lower AUC values, such as Logistic Regression and Gaussian Naive Bayes, which hover around 0.51 to 0.52, demonstrate inadequate classification ability, often misclassifying instances or failing to differentiate between classes effectively.

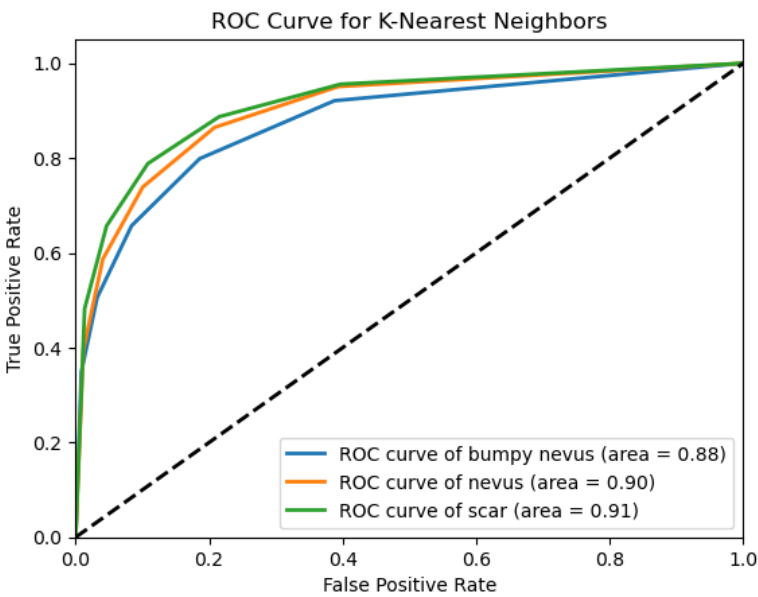


Figure 5.11: ROC curve for K-Nearest Neighbors

Confusion Matrix:

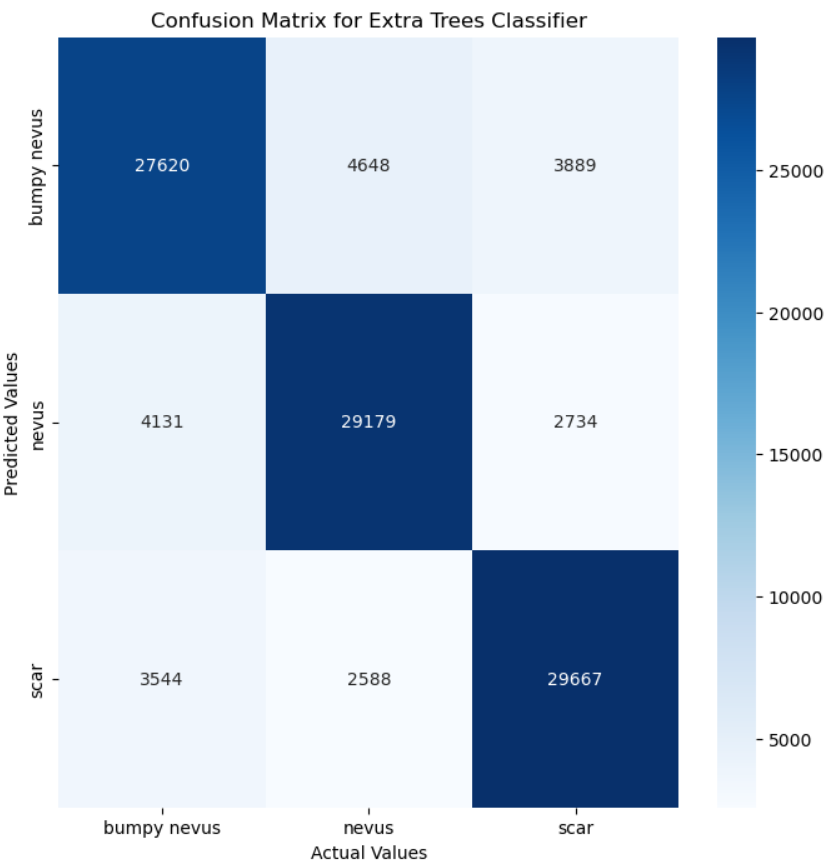


Figure 5.12: Confusion Matrix for Extra Tree Classifier

The K-Nearest Neighbors, Random Forest, and Extra tree classifiers exhibit notably strong performances, especially in correctly predicting the Scar class, suggesting that these models effectively capture the distinctive features of each class.

Conversely, Logistic Regression and Ridge Classifier show significant misclassifications between Bumpy Nevus and Scar, indicating challenges in distinguishing between these classes due to potentially overlapping feature spaces.

Deep learning Model:

The output of the deep learning model indicates a stagnation in performance improvements across the 50 training epochs, as evidenced by the minimal changes in accuracy, which hovered just above the 36% mark, and the consistent loss values. The consistently low validation accuracy near 0.36 suggests that the model struggled to generalize beyond the training data, performing only slightly better than random chance.

5.3.2 Time Domain Data Classification

Time domain data generated using Inverse Fast Fourier Transformation(IFFT) of Balanced Raw frequency mmWave radar data. The dataset labelled with 'bumpy nevus', 'nevus', 'scar'. It is further divided into two parts.

5.3.2.1 Raw Time Domain as a Feature

This dataset consists of 540,000 instances with 150 sample points, each labeled with one of three skin conditions.

The side-by-side evaluation of different machine learning algorithms shows considerable differences in how they perform when analyzing three skin conditions using a balanced dataset derived from raw time domain data, as detailed in table 5.10. K-Nearest Neighbors and Random Forest classifiers outperform others with accuracies nearing 90%, reflecting high precision, recall, and F1-scores, which indicates their superior ability in handling non-linear relationships and intricate patterns in time-domain data. Conversely, Logistic Regression, Ridge Classifier, and Gaussian Naive Bayes exhibit the lowest performance metrics, with accuracies around 34%, suggesting that linear models may struggle with the complexity or non-linear nature of raw time-domain data.

Model	Accuracy	Precision (Weighted Avg.)	Recall (Weighted Avg.)	F1-Score (Weighted Avg.)	AUC -ROC
Logistic Regression	0.3393	0.34	0.34	0.34	0.50- 0.51
Ridge Classifier	0.3409	0.34	0.34	0.34	0.51
K-Nearest Neighbors	0.8956	0.90	0.90	0.90	0.96- 0.98
Support Vector Classifier	0.5354	0.55	0.54	0.53	0.68- 0.74
Decision Tree Classifier	0.7557	0.76	0.76	0.76	0.80- 0.83
Random Forest Classifier	0.8945	0.89	0.89	0.89	0.97- 0.98
Gradient Boosting Classifier	0.4430	0.46	0.44	0.43	0.62- 0.66
AdaBoost Classifier	0.3967	0.40	0.40	0.39	0.56- 0.60
Extra Trees Classifier	0.8758	0.88	0.88	0.88	0.96- 0.98
Gaussian Naive Bayes	0.3429	0.36	0.35	0.31	0.52
Multilayer Perceptron	0.5545	0.56	0.55	0.54	0.70- 0.78

Table 5.10: Performance of ML Models on Balanced Raw Time Domain Data

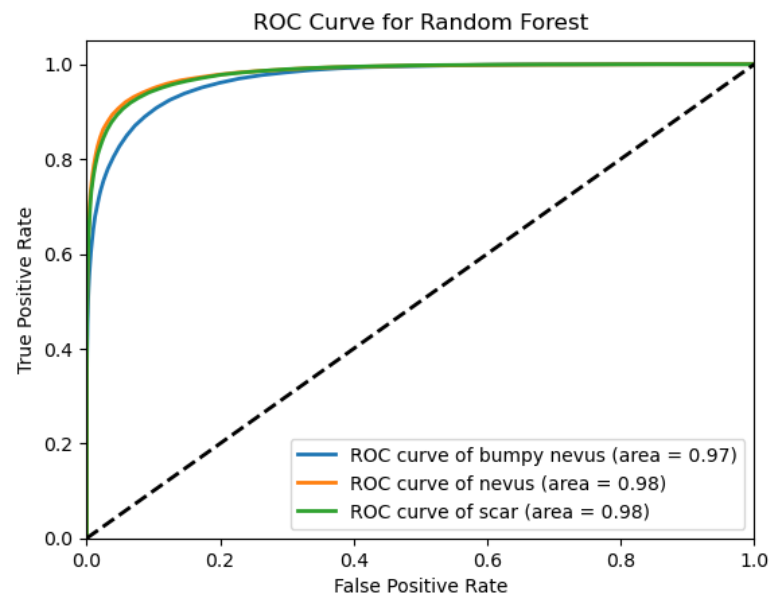
Area Under the Curve (AUC-ROC):

Figure 5.13: ROC curve for Random Forest Classifier

High AUC-ROC values close to 1.0 for models like K-Nearest Neighbors and Random Forest indicates excellent class separation capacity. In contrast, models like Logistic Regression and Gaussian Naive Bayes show AUC-ROC values around 0.50, barely different from random guessing, highlighting their limitations in this specific classification task.

Confusion Matrix:

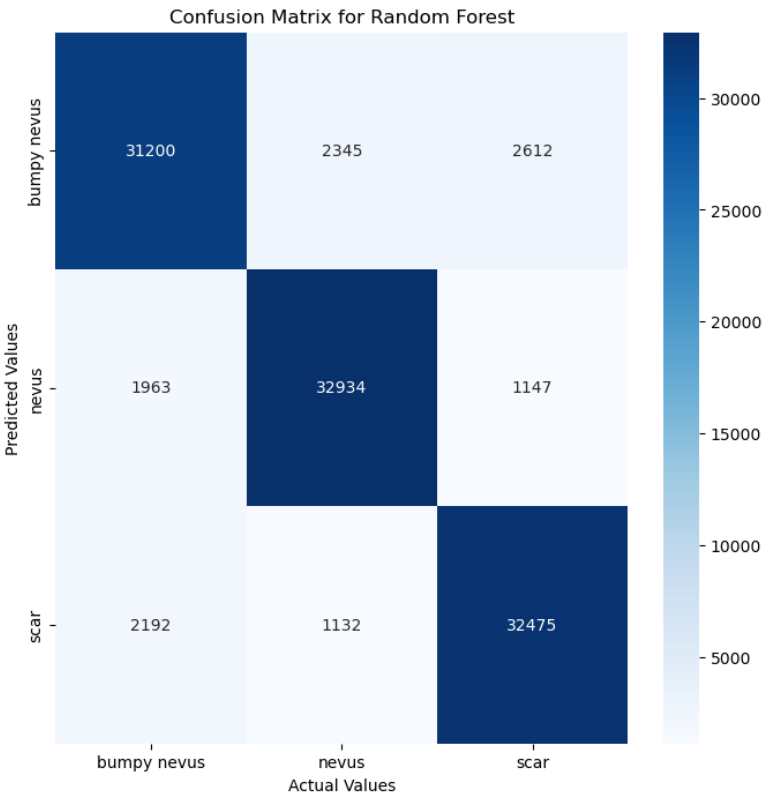


Figure 5.14: Confusion Matrix for Random Forest Classifier

The K-Nearest Neighbors, Random Forest, and Extra tree classifiers exhibit notably strong performances, especially in correctly predicting the Scar class, suggesting that these models effectively capture the distinctive features of each class. On the other part, Logistic Regression and Ridge Classifier show significant misclassifications between Bumpy Nevus and Scar, indicating challenges in distinguishing between these classes due to potentially overlapping feature spaces.

Deep learning Model:

The output of the deep learning model indicates a stagnation in performance improvements across the 50 training epochs, as evidenced by the minimal changes in accuracy, which hovered just above the 35% mark, and the consistent loss values. The consistently low validation accuracy near 0.35 suggests that the model struggled to generalize beyond the training data, performing only slightly better than random chance.

5.3.2.2 Extracted Features: Time Domain

This data set has been generated by extracting features like Signal Magnitude Area (SMA), Energy, and Peak-to-Peak Amplitude, etc. from the raw time domain data making it to a dataset of 880000 instances each with 14 time domain features including label with one of three skin conditions.

Model	Accuracy	Precision (Weighted Avg.)	Recall (Weighted Avg.)	F1-Score (Weighted Avg.)	AUC -ROC
Logistic Regression	0.3483	0.35	0.35	0.34	0.51- 0.52
Ridge Classifier	0.3483	0.35	0.35	0.34	0.51- 0.52
K-Nearest Neighbors	0.7360	0.74	0.74	0.74	0.87- 0.90
Support Vector Classifier	0.4030	0.41	0.40	0.40	0.58- 0.59
Decision Tree Classifier	0.6558	0.66	0.66	0.66	0.72- 0.76
Random Forest Classifier	0.7912	0.79	0.79	0.79	0.92- 0.94
Gradient Boosting Classifier	0.3934	0.39	0.39	0.39	0.56- 0.58
AdaBoost Classifier	0.3722	0.37	0.37	0.37	0.54- 0.55
Extra Trees Classifier	0.7961	0.80	0.80	0.80	0.92- 0.95
Gaussian Naive Bayes	0.3444	0.35	0.34	0.30	0.51- 0.52
Multilayer Perceptron	0.4151	0.42	0.42	0.41	0.59- 0.61

Table 5.11: Performance of ML Models on Extracted Features of Balanced Time Domain Data

In evaluating the classification of extracted time domain data, the experiments yielded varying results across multiple machine learning models. The results of the analysis are given in table 5.11.

Random Forest and Extra Trees Classifier significantly outperform simpler linear models such as Logistic Regression and Ridge Classifier. These ensemble methods, with accuracy scores of 0.7912 and 0.7961 respectively, leverage the strength of multiple decision trees to handle the non-linearity and complexity inherent in the time-domain features. In

contrast, linear models struggle to capture these complexities, reflected in their near-random accuracy scores of around 0.40.

Furthermore, the Precision, Recall, and F1-Score metrics reinforce the superiority of ensemble methods in this context, with both Random Forest and Extra Trees achieving scores around 0.80 across these metrics, indicating robustness in both prediction relevance and result retrieval across class labels.

Area Under the Curve (AUC-ROC):

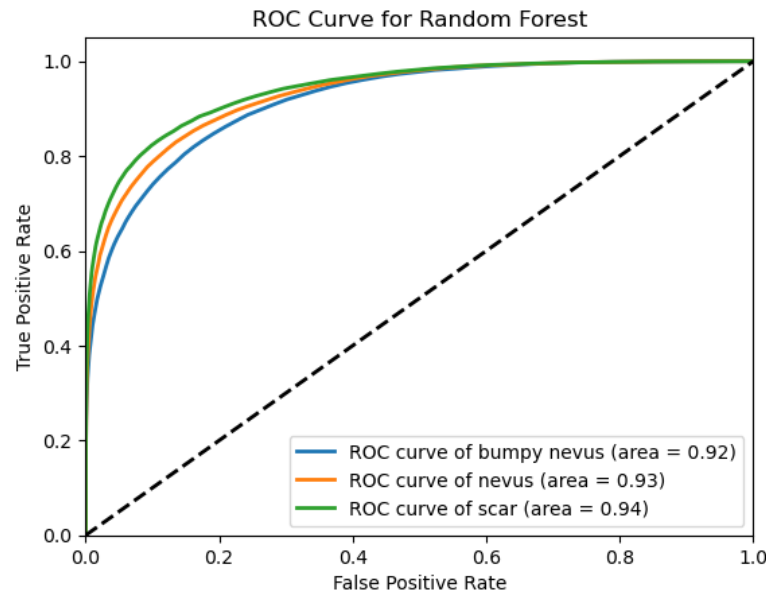


Figure 5.15: ROC curve for Random Forest Classifier

Logistic regression and similar basic models scored just above 0.50, indicating poor discriminative ability, ensemble methods such as Random Forest as in figure 5.15 and Extra Trees Classifier reached AUC-ROC values close to 0.96, demonstrating high true positive rates with fewer false positives. This enhanced performance underscores the effectiveness of ensemble approaches in handling complex patterns within time-domain feature datasets, ultimately leading to better generalization and robust prediction.

Confusion Matrix:

Logistic Regression and Support Vector Classifier show limited ability to differentiate between classes, mostly predicting the majority class, with high false negative rates for most conditions except 'nevus.' K-Nearest Neighbors and Decision Tree classifiers exhibit better performance, with more balanced true positive rates, but still significant confusion among less frequent classes. Random Forest show in figure 5.16 improved detection.

The AdaBoost, Gradient Boosting, and Gaussian Naive Bayes classifier demonstrate moderate effectiveness leading to higher misclassification rates.

In contrast, Extra Trees classifiers show a better generalization ability across multiple classes, although their performance varies widely with skin conditions still being predominantly misclassified as 'nevus.' Multilayer Perceptron achieves decent classification accuracies but shows a tendency to misclassify the 'scar' class.

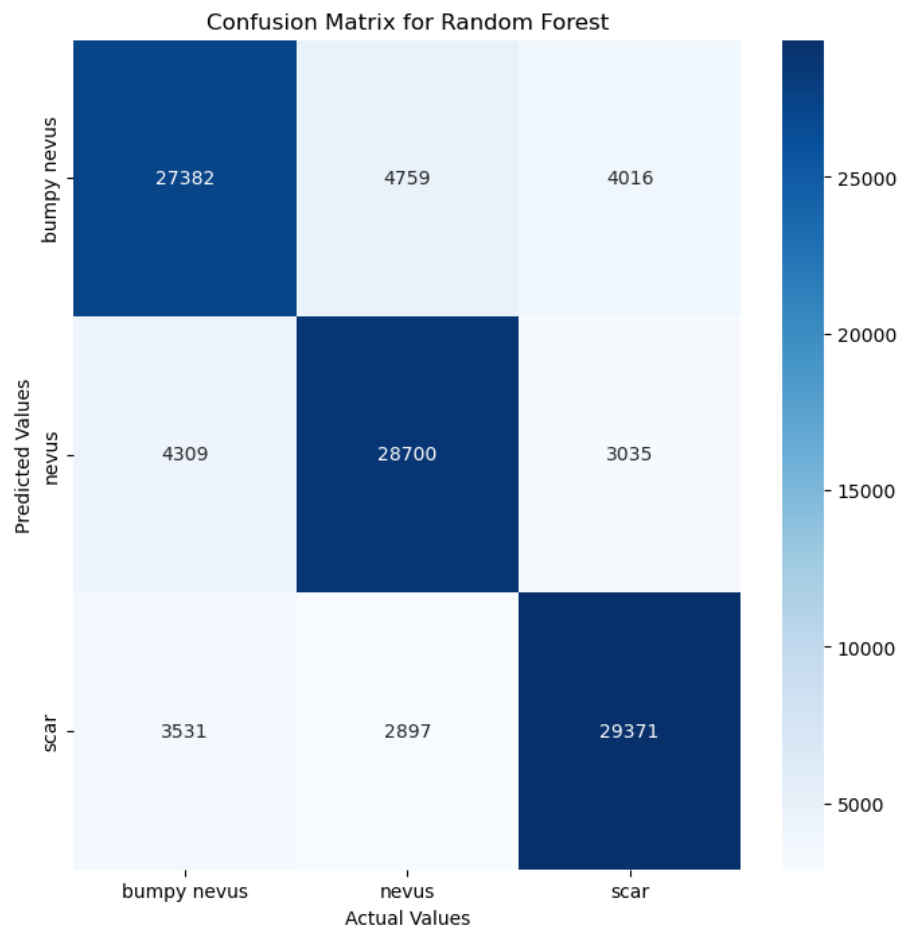


Figure 5.16: Confusion Matrix for Random Forest Classifier

Deep learning Model:

The output of the deep learning model indicates a stagnation in performance improvements across the 50 training epochs, as evidenced by the minimal changes in accuracy, which hovered just above the 35% mark, and the consistent loss values. The consistently low validation accuracy near 0.35 suggests that the model struggled to generalize beyond the training data, performing only slightly better than random chance.

5.4 Spectrogram Image Classification

The spectrogram classification is performed in three experimental parts based on the nature of the spectrogram, recorded in Table 5.12.

A convolutional neural network (CNN) model were employed for spectrogram classification. The architecture includes multiple convolutional layers, max pooling layers to reduce spatial dimensions and control overfitting, and dropout layers to further mitigate overfitting by randomly setting a fraction of input units to zero during training. The resulting metrics were recorded in Table 5.12:

Experiment	Nature of Spectrogram	Val	Test
Experiment 1	Spectrogram of Each Row	Loss: 0.00	Loss: 0.20
	Spectrogram Count: 880000	Acc: 1.00	Acc: 0.99
Experiment 2	Spectrogram of Each Sample File	Loss: 1.2e-06	Loss: 8.2e-06
	Spectrogram Count: 2200	Acc: 1.00	Acc: 1.00
Experiment 3	Spectrogram of Each Sample File	Loss: 5.4e-05	Loss: 1.6e-05
	Balanced Dataset	Acc: 1.00	Acc: 1.00
	Spectrogram Count: 1350		

Table 5.12: Experiment Results for Spectrogram Classification

Analysis of Experiments on Spectrogram Classification:

Experiment 1:

In Experiment 1, the model improved slowly, reaching near-perfect validation accuracy by the end of the first epoch. This trend continued across the subsequent epochs, achieving an accuracy of 1.0000 consistently, which indicates the model could be overly fitting to the training data. The continual perfect scores on the validation set could also indicate that the validation set may not be challenging enough to the training set.

Experiment 2:

In Experiment 2, the model rapidly improved, reaching near-perfect validation accuracy by the end of the first epoch. This trend continued across the subsequent epochs, achieving an accuracy of 1.0000 consistently, which indicates the model could be overly fitting to the training data.

Experiment 3:

In Experiment 3, the setup was adjusted to classify three classes of spectrogram images. Similar to Experiment 2, the initial loss was high, but the model quickly adapted, achieving a validation accuracy of 1.0000 by the second epoch as the plot in figure 5.17. The model continued to perform with perfect accuracy through ten epochs. The rapid convergence to perfect accuracy suggests that the model could be overly fitting to the training data. Consistent perfect scores on the validation set might suggest that the validation data closely resembles the training set.

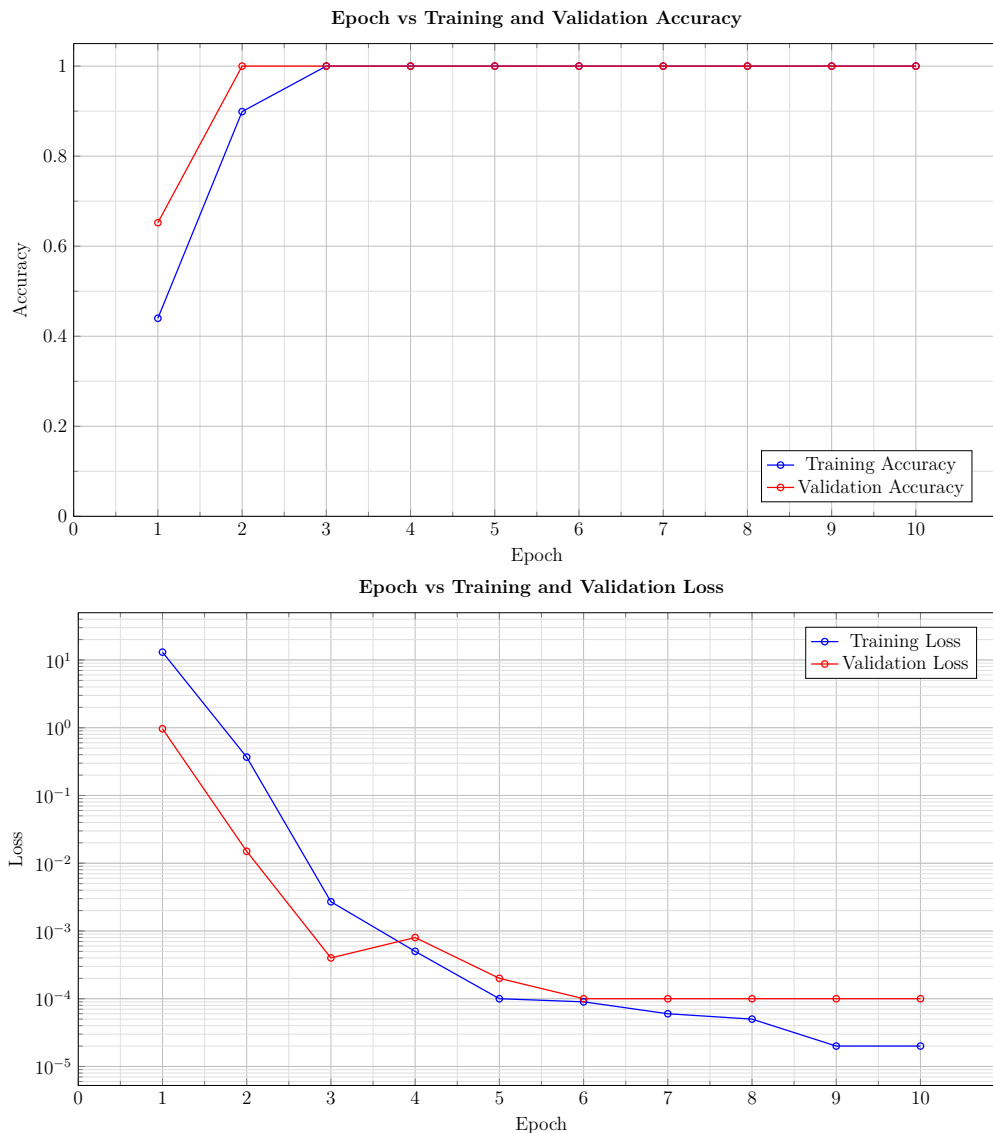


Figure 5.17: Epoch vs Training and Validation Metrics for Experiment 3

Chapter 6

Discussions and Conclusion

In this chapter, an extensive examination is provided regarding the results obtained from the experimental analyses conducted throughout this thesis, emphasizing the categorization of mmWave radar and spectrogram data based on the outcomes of the experiments discussed in Chapter 5.

The experiments conducted in this thesis have demonstrated a range of outcomes across different classification models and data types. In the context of mmWave radar data, both frequency and time domain analyses were explored, employing a variety of machine learning and deep learning models.

Time and Frequency domain representations of mmWave data:

Models trained on raw data generally outperformed those trained on extracted features data, because the raw data highlights the feature of the data that are most relevant to the classification task. The results indicated that ensemble methods based on trees, such as Random Forest and Extra Trees, consistently demonstrated superior performance compared to alternative models. These classifiers achieved high accuracy, precision, recall, and F1-score, especially when dealing with balanced datasets and extracted features. In contrast, simpler models like Logistic Regression and Gaussian Naive Bayes exhibited lower performance, struggling with the complexity and non-linearity inherent in mmWave data. The effectiveness of these ensemble models highlights their capability to handle large, complex datasets by capturing intricate patterns that simpler models often miss.

The deep learning model indicates a stagnation in performance suggesting that the model struggled to generalize beyond the training data, performing only slightly better than random chance. There are various reasons behind this result, such as an unequal distribution within the dataset, insufficient model intricacy to grasp the inherent patterns, or

the necessity for enhanced data preprocessing and feature engineering techniques. The negligible improvement over epochs particularly points to potential issues with the model's ability to learn from the dataset, which might benefit from examining the learning rate, introducing more layers, or employing different architectures such as convolutional neural networks for potentially better feature extraction and performance on this task.

Spectrogram Image Classification:

The experiments with spectrogram images utilized convolutional neural networks, yielding high validation accuracies. However, the persistent flawless performance observed during both validation and testing stages implies a potential issue of overfitting, highlighting the model's adeptness at learning from the training data but raising concerns about its capability to generalize effectively to new, unseen data. This outcome necessitates a re-evaluation of model complexity and training procedures to ensure broader applicability and robustness.

6.1 Implications

The results highlighted in this thesis emphasize the capabilities of sophisticated machine learning methodologies. The high performance of ensemble methods on mmWave data suggests that these approaches can be effectively applied in classification, where accurate and reliable diagnostics are critical. However, the risk of overfitting, as observed in the spectrogram classification tasks, points to the need for careful model selection, validation strategies, and data handling to prevent models from learning to merely memorize training data.

6.2 Limitations

While the experiments provided valuable insights, they also highlighted several limitations:

Overfitting in Deep Learning Models: The deep learning models, despite their high accuracy on training and validation sets, showed signs of overfitting. This issue could limit their practical deployment unless addressed through techniques such as dropout, cross-validation, and the introduction of more generalized training data.

Data Imbalance: Although efforts were made to balance the datasets, the initial unbalanced nature of the data could have influenced the early stages of model training, potentially skewing the learning process and affecting model generalization.

Model Complexity: The complexity of models, particularly in deep learning, makes them challenging to interpret. This complexity could hinder their acceptance in medical practice, where understanding model decisions can be as important as the decisions themselves.

Small Data Set: One major limitation of this study stems from the restricted size of the dataset, which presents difficulties when attempting to generalize the findings. Despite efforts to mitigate this concern through data balancing, the dataset remains constrained, potentially compromising the trustworthiness and resilience of the developed models. To elaborate, the thesis relies on a small 44 data samples, a notably small pool that may hinder the models' robustness and applicability to broader contexts.

6.3 Future Scope

According to the findings of this study, the subsequent suggestions are put forward for future investigation:

Enhanced Data Augmentation: To combat overfitting and improve model robustness, future studies should incorporate more sophisticated data augmentation techniques. This approach would help models learn generalized features and perform better on unseen data.

Hybrid Models: Combining the strengths of different models, such as integrating ensemble methods with deep learning architectures, could yield better performance and reliability. Hybrid models might leverage the robustness of tree-based methods and the learning capability of neural networks.

Cross-Domain Validation: Utilizing models trained on one dataset, such as mm-Wave data, for assessing their efficacy across varying domains, like optical images, may enhance their validation and broaden their applicability within diverse medical contexts.

GPU compatible libraries: Applying GPU compatible libraries to Machine Learning Models could save a lot of time in the model training. As in this some datasets consumed more than 48 hours for training.

Data Collection: Partnering with institutions or organizations to gain access to extra data samples or conducting focused data collection initiatives could yield valuable insights for future research endeavors.

Bibliography

- [1] Chapter 4 of the ntia manual of regulations & procedures for federal radio frequency management (redbook). Online, 2024. <http://www.ntia.doc.gov/osmhome/redbook/redbook.html>.
- [2] F. J. Abdu, Y. Zhang, M. Fu, Y. Li, and Z. Deng. Application of deep learning on millimeter-wave radar signals: A review. *Sensors*, 21(6):1951, 2021.
- [3] S. a. Ahmed. Millimeter-wave radars: Revolutionizing next-generation driver assistance systems. *Radar, Sonar & Navigation*, 2016.
- [4] H. Arab, L. Chioukh, M. Dashti Ardakani, S. Dufour, and S. O. Tatu. Early-stage detection of melanoma skin cancer using contactless millimeter-wave sensors. *IEEE Sensors Journal*, 20(13):7310–7317, 2020.
- [5] C. A. Balanis. *Antenna theory: analysis and design*. John wiley & sons, 2016.
- [6] D. Barton, C. Cook, and P. Hamilton. *Radar Evaluation Handbook*. Artech House, Dedham, MA, 1991.
- [7] B. Bekker. Privacy preserving detection and classification of playground users. Master’s thesis, Delft University of Technology, Delft, Netherlands, 2024.
- [8] L. Blake. *Radar Range-Performance Analysis*. Artech House, Dedham, MA, 1986.
- [9] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
- [10] R. Buettner, M. Bilo, N. Bay, and T. Zubac. A systematic literature review of medical image analysis using deep learning. In *2020 IEEE Symposium on Industrial Electronics Applications (ISIEA)*, pages 1–4, 2020.
- [11] E. V. Carrera, F. Lara, M. Ortiz, A. Tinoco, and R. León. Target detection using radar processors based on machine learning. In *2020 IEEE ANDESCON*, pages 1–5, 2020.
- [12] R. Cesar, Iovescu;Sandeep. The fundamentals of millimeter wave sensors. 2017.

- [13] K. Y. Chan and R. Ramer. Novel concept of detecting basal cell carcinoma in skin tissue using a continuous-wave millimeter-wave rectangular glass filled probe. *Medical Devices: Evidence and Research*, pages 275–285, 2018.
- [14] Q. Chen et al. Millimeter wave radar for industrial applications. *IEEE Transactions on Industrial Electronics*, 2015.
- [15] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia. Multi-view 3d object detection network for autonomous driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1907–1915, 2017.
- [16] F. Chollet et al. Keras. <https://github.com/fchollet/keras>, 2015. Accessed: [16.04.2024].
- [17] T. M. Cover and P. E. Hart. *Nearest Neighbor Pattern Classification*, volume 13. IEEE, 1967.
- [18] D. J. Daniels. *Ground Penetrating Radar*. IET, 2005.
- [19] F. de Ponte Müller. Survey on ranging sensors and cooperative techniques for relative positioning of vehicles. *Sensors*, 17:271, 2017.
- [20] J. Difrancio and W. Rubin. *Radar Detection*. Artech House, Dedham, MA, 1980.
- [21] N. A. W. C. W. Division. Ew and radar systems engineering handbook. <https://apps.dtic.mil/sti/tr/pdf/ADA617071.pdf>, 2013.
- [22] J. Eaves and E. Reedy. *Principles of Modern Radar*. Van Nostrand Reinhold, New York, 1987. Equation (1-16).
- [23] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.
- [24] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, pages 1189–1232, 2001.
- [25] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.
- [26] W. Guo, J. Wang, and S. Wang. Deep multimodal representation learning: A survey. *IEEE Access*, 7:63373–63394, 2019.
- [27] J. Hasch. Millimeter-wave technology for automotive radar sensors in the 77 ghz frequency band. *IEEE Transactions on Microwave Theory and Techniques*, 2012.

- [28] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.
- [29] W. Hong. Multibeam antenna technologies for 5g wireless communications. *IEEE Transactions on Antennas and Propagation*, 2014.
- [30] IEEE. *IEEE Std 149-1979 (Revision of IEEE Std 149-1965): IEEE Standard Test Procedures for Antennas*, 1979. Reaffirmed December 10, 2008; Approved December 15, 1977, IEEE-SA Standards Board; Approved October 9, 2003, American National Standards Institute.
- [31] Institute of Electrical and Electronics Engineers. *IEEE Std 521-2002: Standard for Letter Designations for Radar-Frequency Bands*, 2002. [Online]. Available: <https://ieeexplore.ieee.org/servlet/opac?punumber=7951>.
- [32] International Telecommunication Union. Recommendation ITU-R V.431-8: Nomenclature of the frequency and wavelength bands used in telecommunications. Technical report, International Telecommunication Union, Aug. 2015. [Online; accessed 03-04-2024].
- [33] G. H. John and P. Langley. Estimating continuous distributions in bayesian classifiers. *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345, 1995.
- [34] R. Johnson. *Antenna Engineering Handbook*. McGraw Hill, New York, 3 edition, 1993.
- [35] D. H. Jr, S. Lemeshow, and R. Sturdivant. Applied logistic regression. *Technometrics*, 2013.
- [36] J.-L. Kao and Y.-P. Chao. Breast tumor detection using multi-channel 62-69 ghz millimeter-wave 3d imaging technology. In *2022 IEEE 4th Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability (ECBIOS)*, pages 16–19, 2022.
- [37] R. N. Khushaba and A. J. Hill. Radar-based materials classification using deep wavelet scattering transform: A comparison of centimeter vs. millimeter wave units. *IEEE Robotics and Automation Letters*, 7(2):2016–2022, Apr. 2022.
- [38] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv: Computing Research Repository*, abs/1412.6980, 2014.
- [39] S. P. Kingsley and S. Quegan. Understanding radar systems. 1992.
- [40] J. Krause. *Antennas*. McGraw-Hill, New York, 1988.

- [41] R. Kundu. Confusion matrix: How to use it & interpret results [examples], May 2023. Accessed: 2023-04-27.
- [42] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–444, 2015.
- [43] D. Lee, C. Cheung, and D. Pritsker. Radar-based object classification using an artificial neural network. In *2019 IEEE National Aerospace and Electronics Conference (NAECON)*, pages 305–310, 2019.
- [44] C. Li. Real-time vital signs monitoring and interpretation system using wireless mmwave radar. *IEEE Access*, 2016.
- [45] H.-J. Li. 10 – radar and inverse scattering. 2005.
- [46] F. Mammadova, D. Onwuchekwa, and R. Obermaisser. Towards melanoma detection using radar and image data. In *2023 12th Mediterranean Conference on Embedded Computing (MECO)*, pages 1–10, 2023.
- [47] C. Martins. Dropout layer explained in the context of cnn. <https://cdanielaam.medium.com/dropout-layer-explained-in-the-context-of-cnn-7401114c2c>, 2023. Accessed: 2024-05-09.
- [48] MathWorks. Modeling the propagation of radar signals. <https://de.mathworks.com/help/radar/ug/modeling-the-propagation-of-rf-signals.html>, 2024.
- [49] A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification. In *AAAI Conference on Artificial Intelligence*, 1998.
- [50] J. McCarthy. What is ai? <http://jmc.stanford.edu/artificial-intelligence/what-is-ai/index.html>, 1999.
- [51] W. Menzel and A. Moebius. *Advanced Millimeter-Wave Technologies: Antennas, Packaging, and Circuits*. Wiley, 2013.
- [52] Mini-Circuits. IMAGEVK-74 mmWave Radar Datasheet. <https://www.minicircuits.com/pdfs/IMAGEVK-74.pdf>, 2024. Accessed: 2024-04-25.
- [53] M. Minsky. *The Society of Mind*. Simon & Schuster, 1986.
- [54] A. Mirbeik and N. Ebadi. Deep learning for tumor margin identification in electromagnetic imaging. *Scientific Reports*, 13(1):15925, 2023.
- [55] R. Nabati and H. Qi. Radar-camera sensor fusion for joint object detection and distance estimation in autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 22:1341–1360, 2021.

- [56] F. Nathanson. *Radar Design Principles*. McGraw-Hill, Inc., New York, 2 edition, 1991.
- [57] W. S. Noble. What is a support vector machine? *Nature Biotechnology*, 24(12):1565–1567, 2006.
- [58] U. of Siegen. Getting started – omni cluster, 2023. Accessed: 2024-05-03.
- [59] A. V. Oppenheim and R. W. Schaffer. *Discrete-Time Signal Processing*. Prentice Hall, 2nd edition, 1999.
- [60] E. Osuna, R. Freund, and F. Girosi. An improved training algorithm for support vector machines. In J. Principe, L. Gile, N. Morgan, and E. Wilson, editors, *Neural Networks for Signal Processing VII. Proceedings of the 1997 IEEE Signal Processing Society Workshop*, pages 276–285, New York, 1997. IEEE.
- [61] A. Y. Owda, N. Salmon, A. J. Casson, and M. Owda. The reflectance of human skin in the millimeter-wave band. *Sensors*, 20(5):1480, 2020.
- [62] S. M. Patole. Automotive radars: A review of signal processing techniques. *IEEE Signal Processing Magazine*, 2017.
- [63] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, and T. Sainath. Deep learning for audio signal processing. *IEEE Journal of Selected Topics in Signal Processing*, 13(2):206–219, 2019.
- [64] K. Rajbanshi. Multimodal analysis of millimetre wave and image data for skin cancer detection. Master’s thesis, University of Siegen, Siegen, Germany, 4 2024. Chair for Embedded Systems.
- [65] T. S. Rappaport. Wireless communications and applications above 100 ghz: Opportunities and challenges for 6g and beyond. *IEEE Access*, 2017.
- [66] A. Rathi, D. Deb, N. Sarath Babu, and R. Mamgain. Two-level classification of radar targets using machine learning. In *Smart Trends in Computing and Communications: Proceedings of SmartCom 2019*, pages 231–242. Springer, 2020.
- [67] M. A. Richards. *Fundamentals of Radar Signal Processing*. McGraw-Hill Education, Atlanta, GA, USA, 2014. Georgia Institute of Technology.
- [68] M. A. Richards, J. Scheer, W. A. Holm, and W. L. Melvin. Principles of modern radar. 2010.

- [69] P. N. Robinson and M. R. Yin. Modified spherical coordinates for radar. *American Institute of Aeronautics and Astronautics*, 1994. Copyright c 1994 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.
- [70] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.
- [71] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.
- [72] M. Skolnik. An introduction and overview of radar. *Radar Handbook*, 3:1–1, 2008.
- [73] M. I. Skolnik. Introduction to radar. *Radar handbook*, 2:21, 1962.
- [74] A. Stove. Linear fmcw radar techniques. *Proceedings of the IEEE*, 1992.
- [75] A. G. Stove. Linear fmcw radar techniques. *IEE Proceedings F - Radar and Signal Processing*, 1992.
- [76] W. Stutzman and G. Thiele. *Antenna Theory and Design*. Wiley, New York, 2 edition, 1997.
- [77] R. Sullivan. *Radar Foundations for Imaging and Advanced Concepts*. SciTech Publishing, Inc., Raleigh, NC, 2004.
- [78] F. Topfer and J. Oberhammer. Millimeter-wave tissue diagnosis: The most promising fields for medical applications. *IEEE Microwave Magazine*, 16(4):97–113, 2015.
- [79] A. M. Turing. Computing machinery and intelligence. *Machine Intelligence: Perspectives on the Computational Model*, pages 1–28, 1950.
- [80] J. Weiß and A. Santra. Material classification using 60-ghz radar and deep convolutional neural network. In *2019 International Radar Conference (RADAR)*, pages 1–6, 2019.
- [81] F. Yanovsky. *Millimeter Wave Radar: Principles and Applications*, pages 305–376. 01 2008.
- [82] H.-S. Yeo, G. Flamich, P. Schrempf, D. Harris-Birtill, and A. Quigley. Radarcats: Radar categorization for input & interaction. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, page 833–841, New York, NY, USA, 2016. Association for Computing Machinery.